Contents

Introduction 9

I Fundamentals 11

1 Introduction to the Satellite communications 13
   1.1 History 13
   1.2 Ground segment 13
   1.3 Satellite channel 15

2 Autonomous Receiver: a short introduction 17
   2.1 Definition of an Autonomous Receiver 17
      2.1.1 Basic Architecture of an Autonomous Receiver 18
   2.2 Motivation: need for an Autonomous Receiver 20
   2.3 State of the art 21

II Modulation classification 23

3 Overview of the different Modulation Classification (MC) methods 25
   3.1 Introduction 25
   3.2 Pattern recognition (PR) 26
      3.2.1 Tree structure decision subsystem 27
      3.2.2 Tree structure vs. Artificial Neuronal Network 27
      3.2.3 Auto Regressive spectrum estimate feature extraction 29
      3.2.4 Statistical moments 29
      3.2.5 Artificial Neuronal Network decision system 29
      3.2.6 Maximum Likelihood decision system 30
      3.2.7 Conclusion 31
   3.3 Maximum likelihood classification (ML) 32
      3.3.1 Two hypothesis MPSK method 33
      3.3.2 Three hypothesis ML method 34
      3.3.3 Theoretical Results 34
      3.3.4 Suboptimal classifier 34
      3.3.5 Various publications 34
3.3.6 Conclusion .................................................. 34
3.4 Comparison of the different methods ............................................ 35

4 Hybrid Modulation Classifier for satellite telemetry signals 37
  4.1 Description of the satellite scenario and state of the art .................. 37
  4.1.1 State of the art ........................................... 37
  4.1.2 Satellite scenario general requirements .................................. 38
  4.1.3 Design of the algorithm: challenging design parameters .............. 38
  4.2 Hybrid modulation classifier (MC) for satellite telemetry signals ....... 39
  4.2.1 Design of the algorithm ........................................ 39
  4.2.2 Implementation of the Analog vs. Digital classifier ...................... 49
  4.2.3 Full Analog vs. Digital classifier .................................. 56
  4.3 Experimental results: analog vs. digital classifier ......................... 57
  4.3.1 Simulation parameters ........................................... 57
  4.3.2 Multiple estimation ............................................ 57
  4.3.3 Simulation parameters ........................................... 57
  4.3.4 Experimental results ............................................ 58
  4.3.5 Performance dependence of different parameters ......................... 62
  4.3.6 Gaia and Bepicolombo missions performance ................................ 67
  4.3.7 Conclusions .................................................. 68
  4.4 Further work and new proposals ............................................. 70

III Symbol-rate Estimation 73

5 Overview of the different Symbol-rate estimation methods 75
  5.1 Introduction .................................................. 75
  5.2 Maximum Likelihood symbol-rate estimate .................................. 76
  5.3 Cyclostationarity based symbol-rate estimation technique ................ 79
    5.3.1 Filter bank ................................................. 81
    5.4 Inverse Fourier transform ......................................... 82
    5.5 Symbol-rate and SNR joint estimation based on the Split Symbol Moment Estimation (SSME) ........................................ 83
  5.6 Other methods .................................................. 85
  5.7 Conclusion .................................................. 86

6 Broad Range Symbol-rate Estimation for Satellite Scenarios 89
  6.1 Description of the satellite scenario and state of the art ................. 89
    6.1.1 State of the art ........................................... 89
    6.1.2 Satellite scenario: general requirements ................................ 89
    6.1.3 Design of the algorithm: challenging design parameters ............. 90
  6.2 Broad range symbol rate estimate for satellite scenario .................. 91
    6.2.1 Design of the algorithm ........................................ 91
  6.3 Experimental results ............................................. 105
    6.3.1 Simulation parameters ........................................... 105
    6.3.2 Experimental results ............................................. 107
    6.3.3 Conclusions .................................................. 111
6.4 Further work and new proposals

6.4.1 Time and complexity constrains

6.4.2 Coarse estimate and decimate module

6.4.3 Fine estimate

Conclusions

IV Appendixes

A Analysis of the Telecommand and Telemetry Modulations

A.1 Carrier Modulations: SP-L/PM

A.2 Sub Carrier Modulations

A.2.1 PCM/PSK(sine)/PM modulation

A.2.2 PCM/PSK(square)/PM(modulation)

A.3 Suppressed carrier modulation

A.3.1 OQPSK

A.3.2 GMSK

A.4 MATLAB code

A.4.1 GMSK

A.4.2 OQPSK

B Review of the application scenarios

B.1 ESA missions

B.2 Overview of the scenarios specifics

C TT&C: Telemetry, Tracking, and Commanding

D Modulation index estimation

Bibliography
Després d'aquests meravellosos anys d'estudi, vull agrair a totes les persones que m'han donat suport per arribar al final del túnel. Als meus pares, germans i família pel suport moral i econòmic i per motivar-me constantment. Als companys de la UPC per les interminables tardes al despatx, a les Aigües o a econòmiques. Als companys de la TUD, per haver fet durat dos anys de pares, amics, companys de classe i tot el que fes falta. A la gent d’ESOC per estar sempre disposats a respondre preguntes i per fer-me-les entendre. Als meus amics: gràcies pel suport moral durant tota la carrera, tant en forma de cerveses pel barri, com en excursions al càmping o visites durant el dur hivern alemany. Als d’hoquei, per haver-nos divertit tant. A tots per aguantar-me i entendre'm.
Introduction

Nowadays the number of spacecraft in the solar system is huge. Since the start of the space race, the launch of spacecraft has shown a growing rate. Many kinds of spacecraft coexist in space: there are satellites that have been in the space for the last 20 years and spacecraft that have been recently launched, apart from spacecraft designed for specific missions and designed by different space agencies. For this reason, at the present time there is a diversity of communication technologies working in different missions.

From the Ground Stations it is needed to provide support to all these kinds of missions. Specifically, stations have to receive the Telemetry signal from the spacecraft and transmit Telecommand in order to change the spacecraft configuration. Communication parameters are different depending on the mission. In addition, they may also change during a specific mission due to varying conditions. The diversity and variability of the communication parameters force the receivers in the Ground Stations to continuously switch their configuration. This process is done manually, with the consequences of time expense, possibility of human failure and overloaded technicians schedule.

In scenarios like these, where the receiver has to be continuously reconfigurated, is where the concept of Autonomous Receiver is introduced. An Autonomous Receiver is a receiver that is capable of estimating the incoming signal parameters. To be more specific, it is able to demodulate and decode an unknown received signal without any external help. This automatic configuration process consists of an iterative algorithm where the communication parameters are estimated. ESA is planning to incorporate Autonomous Receivers in its communication systems standards in the near future. For this reason, preliminary research is being done in the specifications and system architecture of such receivers. This Diplomarbeit is part of these first stages of work on Autonomous Receivers applicable to ESA missions.
The main focus is on the research and design of symbol-rate estimation and modulation classification algorithms to be applied in an Autonomous Receiver as well as the relation between these two estimation modules inside an Autonomous Receiver. Most of the published work on parameter estimation is not directly applicable to ESA requirements. So, an initial study on current methods has been performed, and then new ones have been developed to suit these requirements.

The document is organized in 5 parts. Part I makes an introduction to the satellite scenario, and a motivation for the need of an AR and also a short description of the architecture of an AR. Part II and III center on the research and design of applicable algorithms for the AR. Part II deals with the research and design of a Modulation classifier method: in Chapter 3 a summary of the performance and applicable scenarios of the different published method for the Modulation Classification is shown. The different trade-offs between these methods are compared and commented. In Chapter 4 a Modulation Classifier algorithm is designed. The design focuses on how to coarsely classify the different modulations in 2 groups. In addition, more refined algorithms to classify the modulations are designed. Part III is centered on the research and design of a symbol-rate estimation method: in Chapter 5 a summary of the performance and applicable scenarios of the different published method for the Symbol-rate estimation is shown. The different trade-offs between these methods are shown and commented. In Chapter 6 a symbol-rate estimation algorithm is presented. Our design efforts are concentrated in designing an estimator that should be able to estimate the symbol-rate within a broad range of different symbol-rates. The last part (Part IV) consists of a discussion about how the two designed modules should fit inside an AR architecture. The conclusion about the performance, and the advantages of the designed modules with respect to the previous existing methods are also commented. The Appendixes at the end of this document contain additional information about ESA missions.
Part I

Fundamentals
Chapter 1

Introduction to the Satellite communications

1.1 History

The artificial satellite history starts after World War II, with the launch of Sputnik 1 by the Soviet Union in October 1957. This launch started the Space Race along which there was a fast development of satellite technologies. On November 3 of 1957, Sputnik 2 was launched, and carried the first living passenger into orbit, a dog named Laika. Explorer 1 became the United States first satellite on January 31, 1958. On July 20, 1969 the United States achieved the first manned landing on Earth’s Moon, when the lunar module Eagle landed on the surface of the Moon as part of the Apollo 11 mission commanded by Neil Armstrong.

Nowadays exist many different kinds of satellites, orbiting in different orbits around the Earth (like GAIA or GOCE) or in deep space missions like Mars Express or Venus Express. There are different uses for satellites, such as commercial (e.g. satellite TV), scientific missions (Ocean circulation), space observation (Hubble) and deep space missions (Rosetta, Bepicolombo). The largest artificial satellite currently orbiting the Earth is the International Space Station.

1.2 Ground segment

The support to all this missions is done in the Ground stations (station in the Earth). The Ground Stations of ESA are organized in the ESTRACK Network (see Figure 1.1) and coordinated from ESOC. Its mission is to provide support to the different ESA satellites. The large number of Ground stations provides ESA with flexibility, increased visibility of the satellites, robustness against climatic or logistics problems and allows more mission
allocation. There are also Ground Stations with specific configurations (large antennas etc.) to accomplish specific tasks, for example the Cebreros 35 meter antenna should support the deep-space Bepicolombo mission. The main tasks of the Ground Stations are:

- Receive the Telemetry data from the satellites.
- Store and distribute the Telemetry data.
- Transmit Telecommand to the satellites.
- Provide measures of Doppler and Ranging.

There is high number of missions and each mission/satellite has its own special communication characteristics. For this reason the Ground Stations need a configuration phase previous to every satellite pass. The receivers have to be configured in the specific operation mode of the particular satellite, including bit-rate, modulation scheme, estimated Doppler, frequency band, codification. Also, the antenna has to point to the estimated satellite position.

As an Example, the steps for a satellite pass could be:

- Ground Station configuration
  - LNA\(^1\), HPA\(^2\), converters, switches, IFMS\(^3\), etc.
- Ranging calibration
  - Station configured in Long Loop, the ranging signal is reflected at the antenna phase center back to the baseband unit.
- Beginning of the pass
- Spacecraft becomes visible for the ground station once is above the station horizon mask.
- Reception of TM\(^4\)
- Once the spacecraft goes above 5 degrees for non-deep space missions (transmitted power 2kW) or above 10 degrees for deep space missions (20kW), Ground Station start uplink sweep.
- Once locked onto the uplink carrier, the spacecraft goes into coherent mode.
- Configure IFMS to coherent mode.
- Ranging and Doppler are started by the station.
- Start Ranging and Doppler measurement.

---

\(^1\)Low Noise Amplifiers
\(^2\)High Power Amplifiers
\(^3\)Intermediate Frequency Modem System the ESA standard receiver
\(^4\)Telemetry
• Spacecraft TC\textsuperscript{1} is made available for the SPACON\textsuperscript{2} to begin.

• Decide to change symbol-rate of the TM
  – Uplink TC: change spacecraft symbol-rate
  – IFMS reconfiguration
  – Spacecraft transmits with the new symbol-rate

• Decide to change modulation scheme of the TM
  – Uplink TC: change spacecraft modulation scheme
  – IFMS reconfiguration
  – Spacecraft transmits with the new modulation scheme

• At elevation of 10 or 5 degrees descending ranging measurements are stopped and uplink carrier is brought down.

• Having lost uplink carrier, the spacecraft goes into non-coherent mode again.

• Configure IFMS to non-coherent mode.

• Once the spacecraft goes below ground station horizon mask, TM signal is lost.

• End of tracking

• Ground station antenna pointing is brought to the STOW position (safety)

Notice that even inside a single satellite pass the modem has to be manually reconfigured a lot of times. The technician in the Ground Station had to look up at the reconfiguration schedule and configure the modem manually every time any change on the modulation scheme, coding scheme, Doppler etc. occurs.

1.3 Satellite channel

Due to the specific conditions of the different satellite scenarios each satellite channel could have specific characteristics. In addition, the availability of energy sources in the space is very restricted. The satellites just have two power sources: photovoltaic panels and a limited amount of fuel. Due to this limitation, fuel is only used to correct/change the satellite position. All the power needed for the modems, amplifiers, antennas is extracted from the photovoltaic panels. For this reason the transmitted power (downlink) is always pretty small, however the available power in the uplink channel is high. Because of this the SNR is better in the uplink channel than in the downlink channel. The visibility (duration of a satellite pass) depends on the orbit of the mission. Some missions have visibility spans of around 10 hours (like Rosetta), while other satellites are visible for less than 5 straight minutes (like Goce). For this later case (small satellite pass length) reducing the configuration time of the equipment is very important. In addition, the satellites and the Ground Stations are in constant motion, this relative motion between transmitter and

\textsuperscript{1}Telecommand
\textsuperscript{2}SPAcecraft CONtroller
receiver produces high Doppler. Here follows a list of the main characteristics of a satellite channel.

- Small available power in to the on board (satellite) transmitter.
- Small computational power in the on board (satellite) transmitter.
- Long distance (even in low orbits) high losses and delays.
- Atmospheric attenuation.
- Receiver signal in ground stations is weak, so there is need for low noise receivers.
- Time of the satellite visibility (satellite pass) is limited.
- High Doppler effect.
Chapter 2

Autonomous Receiver: a short introduction

In this Chapter the Autonomous Receiver (AR) will be briefly introduced. We will introduce its main parts, basic architecture, applicable scenarios and definition, along with a small review of the state of the art in AR research.

2.1 Definition of an Autonomous Receiver

An AR is an autonomous radio that can receive (demodulate and decode) a signal without much a priori knowledge about its defining characteristics. In [17] is shown that this may be accomplished by estimating, classifying and extracting the relevant properties from the observed signal, making the autonomous configuration of the AR possible. The AR consists of different estimation blocks that interact to ultimately recover the transmitted message. Each step in the demodulating process uses its own methods and makes use of the results of the previous ones to refine its estimations, and vice-versa: the results of the next step are used by the previous ones to refine the search [46, 17].

The fundamental difference between a conventional radio and an AR is that an AR has the ability to recognize features of an incoming signal and to respond automatically. Therefore, no explicit pre-configuration or manual reprogramming to define the functions of the radio has to be done [16]. A block diagram of a receiver is shown in Figure 2.1. The tasks that the receiver has to perform are shown in rectangles, while the a priori knowledge about the signal characteristics is shown in ellipses. The knowledge of the transmitted signal parameters simplifies the design and implementation of the receiver. However a conventional radio usually does not have the capability to receive signal types different from the single type for which it was designed. On the other hand, from the AR point of view the parameters shown in ellipses in Figure 2.1 are assumed unknown a priori and they have to be estimated from the incoming signal. The performance of the estimators and classifiers of the AR is limited by its lack of knowledge of any of the other parameters. Owing to this fact, the order of the estimations/classifications and their internal relationships is critical.
2.1.1 Basic Architecture of an Autonomous Receiver

The basic procedure of an AR consists of the estimation of the incoming signal parameters. In order to classify the signal to ultimately demodulate it. The parameters to be estimated are shown in ellipses in Figure 2.1. However, the order in which these parameters are estimated and the methods used for the estimation are scenario dependent. Using conventional estimation and tracking designs, one quickly gets into a chicken and egg problem, with nearly every estimator needing the output of the other estimators before it can make an estimate.

As far as the author’s knowledge goes, the architectures proposed in [46, 17] are the only ones published. The main idea of the AR is to estimate the signal parameters in multiple stages and many iterations. In the first iterations of the algorithm the modules perform coarse estimations in blind mode. In the following iterations the modules enter an aided mode and make use of the results obtained by the other modules in previous iterations. As a consequence the accuracy of the estimations increases. While the output of the different estimation modules increases in accuracy, the modules in aided mode increases also its performance since they are using more accurate information to estimate parameters. Finally the system should enter into a stationary phase where the different estimations of the parameters remain constant.

This iterative estimation of the signal parameters, which starts with coarse estimates (blind mode), later on changes to fine estimation (aid mode), using the other modules estimations and its own previous results, is characteristic from the AR architecture.

As an Example of AR architecture we will briefly introduce the architecture proposed by JPL in [17]. In Figure 2.2 the different estimation modules of this architecture and their order is shown. Notice that the estimation of Data-rate, Pulse shape, SNR and coarse symbol timing is done jointly after the estimation of the modulation index and the frequency. The specific characteristics of the NASA missions and standards have shaped the current architectural design of their AR. However, ESA missions and standards differ from those of NASA, so a different architectural design should be expected for an ESA AR.
In Figure 2.3 a detailed scheme of the JPL AR is shown. It can be seen that each estimation module has different modes of operation (for example the Carrier synchronization module has 4 modes of operation, depending on the estimations of the other modules it uses one or another). It is important to notice that most of the estimation modules receive information from the other modules, nearly all of the modules are connected to each other.

At first glance, it may seem that some of these estimator modules are long-established, conventional designs. For example, phase tracking loops have been designed and analyzed for nearly every reasonable signal type. However, in [17] authors were unable to find any literature for the design of a phase tracking loop for suppressed-carrier signals in which the modulation type is unknown. As a consequence after designing the architecture of an AR, it will be also necessary to adapt conventional methods or design new methods for the different estimation blocks.

Figure 2.2: Basic scheme of the AR architecture proposed in [17]
Figure 2.3: Scheme of the AR architecture [17] with the iterative feedbacks, notice that the different estimation modules have different modes of operation concerning the aid (estimation from other modules) that they have.

2.2 Motivation: need for an Autonomous Receiver

As an Example here we will introduce some scenarios where an AR will be useful.
1. ESA ground stations: each ground station has to provide support to different missions, that implies different frequency bands, modulation schemes, data rates, coding schemes etc. The demodulator configuration will be different from each term. The AR will provide the demodulator to automatically configure it selfs, avoiding configuration schedules and human errors.

2. Mission allocation: some times there is a short time between passes, in this times the ground stations have to change its configuration. An automatic configuration procedure will decrease the configuration time and will increase the number of Mission that a ground station can give support.

3. Satellite pass duration: the pass of a satellite could have length from a few minutes to more ten hour. For short satellite passes is it important to configure the ground station rapidly in order to spend just a few seconds of the pass.

4. In addition to easing the scheduling, an AR also will handle unpredictable or anomalous events. Even when all scheduled events occur successfully, there may be uncertainty due to unpredictable factors (large Doppler during a landing or satellite entering in safe mode because a malfunction detection).

5. Communication signals from multiple deep space assets. For example different rovers in a planet that relay with a satellite in the planet orbit. The advantage of an AR in this emerging scenario is that it can communicate to each asset that comes into view, automatically, without having to be reconfigured from Earth for each pass to account for differences in the signal characteristics.

2.3 State of the art

In this section we will briefly explain the research state of the different agencies in the AR field.

**JPL and NASA:** The research in the AR field started in NASA about 15 years ago and many papers have been published about the general architecture and different algorithms for the estimation of various signal parameters. In [16] they show and AR software testbed, where the architecture described in [24] and the different estimation modules described in the different published papers are fitted all together. In addition in [17] all the different papers related with the AR design are compiled.

**Electra:** is the NASA’s first highly capable software-defined radio [12]. Electra is designed to operate in BPSK over a wide range of data rates from 1 kbps to 4Mbps, and can accommodate frequency uncertainties up to 20KHz. Electra is now the standard in situ receiver and will fly on the Mars Reconnaissance Orbiter, Mars Telecommunications Orbiter, and Mars Science Laboratory missions among others [24]. It is not an AR but it is the first flexible receiver that is designed with a philosophy similar to that of an AR.

**ESA:** ESA is starting a research line to define the architecture for an AR that should fit ESA mission parameters. Its studies are just in the architectural design part, but in the following stages it is planned to include the AR in its standards.
Published papers: Different Universities and Research centers have published different algorithms to estimate different signal parameters. In most of the cases these algorithm are ad-hoc algorithms for specific scenarios and have a lack on the general purpose that an AR should have. In addition, the techniques to estimate these parameters are not linked one to each other and as a consequence there are no proposed architectures for an AR. However, they can be the starting point of the design of an AR.
Part II

Modulation classification
Overview of the different Modulation Classification (MC) methods

In this Chapter we will introduce different methods for the automatic classification of modulations (MC). Since there is a big amount of publications just the most relevant methods will be explained in detail. The different methods will be compared in terms of probability of correct classification $P_{cc}$. In addition, an important parameter of interest is the degree of autonomy of the MC. The degree of autonomy is defined as amount of knowledge about the signal parameters that is need for the MC to make a classification of the modulation $^1$.

3.1 Introduction

Classically, one can consider two main approaches: the Pattern Recognition (PR) and the Maximum Likelihood (ML) approach. In the first one some statistical representations of the signal or some of its parameters are extracted from the observed signal and used as discriminating features. In the ML approach, quasi-optimal rules are derived from the development of the average log likelihood function of the signal. In Section 3.2 different methods for the PR are presented, in Section 3.3 different methods and approaches to the Maximum Likelihood function are presented, finally in Section 3.4 a comparison of the different methods of both main approaches is presented.

$^1$A estimation technique that doesn't need, additional signal information is call blind estimation (high autonomy), however when the estimation technique requires information about the incoming signal is call aid estimation (low autonomy).
3.2 Pattern recognition (PR)

The pattern recognition classifier is composed of two subsystems.

- **Feature extraction** subsystem which role is to extract useful information from the input signal. The features are parameters that can be used to differentiate between distinct modulations. As an example in [30] some of the used spectral features are
  
  - the Maximum Value of the Spectral Power Density of the normalized-centered instantaneous amplitude $\gamma_{\text{max}}$ defined in Equation 3.1.
  
  - Standard deviation of the absolute value of the centered, non-linear component of the instantaneous phase $\sigma_{\text{ap}}$ defined in the Equation 3.3.

- **Pattern recognizer** subsystem whose function is to, concerning the extracted features, classify the incoming modulation. An example on how to classify can be seen in Table 3.1 where the previous features ($\sigma_{\text{ap}}$ and $\gamma_{\text{max}}$) are used to distinguish between angle modulation, amplitude modulations or a combination of both.

\[
\gamma_{\text{max}} = \frac{\max |(\text{fft}(a_{\text{cn}}(i))^2)|}{N} \tag{3.1}
\]

where $a_{\text{cn}}$ is defined as:

\[
a_{\text{cn}} = \frac{a(i) - \frac{1}{N} \sum_{i=1}^{N} a(i)}{\frac{1}{N} \sum_{i=1}^{N} a(i)} \tag{3.2}
\]

where $a(i)$ is the instantaneous amplitude.

\[
\sigma_{\text{ap}} = \sqrt{\frac{1}{c} \left( \sum_{a_{\text{cn}}(i)>a_t} \phi_{\text{NL}}^2(i) \right) - \frac{1}{c} \left( \sum_{a_{\text{cn}}(i)>a_t} \phi_{\text{NL}}(i) \right)^2} \tag{3.3}
\]

where $c$ is the number of samples and $\phi_{\text{NL}}(i)$ is the value of the centered non-linear component of the instantaneous phase.

One of the problems of the PR lies in defining a good set of features to be measured and provide a rigorous mathematical proof or justification that these algorithms are optimum in a statistical sense. In most cases, pattern recognition algorithms rely on defining and extracting a set of features that, intuitively, look as an optimum set. The feature set should consist on a group of parameters that are sensible to the different characteristics of the different modulations to be classified. For example $\gamma_{\text{max}}$ provides an idea of the amplitude variability. As a consequence a modulation with constant envelope may have a small value of $\gamma_{\text{max}}$ in the other hand a amplitude modulation (high amplitude variability) should present a high $\gamma_{\text{max}}$ value. In the Pattern recognizer subsystem (analyze the features and decide for a modulation) there are different approaches:

1. Tree structure (hard decision).
2. Artificial Neuronal Network decision system.
3. Maximum likelihood decision system \(^1\).

In the following sections we present some different implementations of the decision subsys-
tems, the parameter sets are also a bit different from one method to the other since the
modulations that have to be classify are also different.

### 3.2.1 Tree structure decision subsystem

Azzouz et. al. \[30\] proposed a PR recognition method to distinguish the following mod-
ulations: AM, DSB, MASK, band-limited PSK2, FM, MFSK, ideal MPSK, combined
AM-FM, SSB, band limited MPSK, CW and ideal PSK2. This method consist on a two
step pattern recognizer system. First, features are extracted in order to detect where the
information content of the incoming signal is (in the instantaneous amplitude, in the instan-
taneous phase or in a combination of them), later on a specific group-features are extracted
in order to classify the modulation. So the first classification is a rough classification, after-
wards more specific features are extracted in order to make a fine modulation classification.
Once the features are extracted the modulation recognition is done by comparing (hard
decision) the features values to certain threshold values. The values of that thresholds are
derived off-line and are chosen in order to minimize the misclassification. Table 3.1 shows
the hard decision table for the first step of the algorithm.

<table>
<thead>
<tr>
<th>(\sigma_{ap} \leq t_{\sigma_{ap}})</th>
<th>(\gamma_{max} \leq t_{\gamma_{max}})</th>
<th>(\gamma_{max} &gt; t_{\gamma_{max}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>CW signal</td>
<td>amplitude signal</td>
<td></td>
</tr>
<tr>
<td>angle modulated signal</td>
<td>combined signal</td>
<td></td>
</tr>
</tbody>
</table>

**Table 3.1:** Example of hard decision for modulation classification. Extracted from \[30\]. The
threshold values are usually decided experimentally.

Experimental results have been carried out at two different SNR levels 10dB and 20dB,
the experiments shows that for a SNR of 10 dB the probability of correct classification of
the first step is about the 90%. No simulations for the second step are comented.

### 3.2.2 Tree structure vs. Artificial Neuronal Network

Nandi and Azzouz in \[7\] compared two different decision algorithms for discriminating
the modulations after the feature extraction. The first one is based in a multi step tree
structure. The second method tries to avoid the hard decision by using an Artificial Neuron
Network (ANN). Simulations with 12 different analog modulations and 9 different digital
modulations show that: the \(P_{cc}\) for the tree structure is 94% and for the ANN the \(P_{cc}\) is
96% for a SNR of 15dB. Figures 3.1 and 3.2 plot the decision scheme for the tree decision
method and the ANN respectively.

---

\(^1\)It is important no to confuse with the Maximum Likelihood method. The method here described is a
PR method where the decision system is based on a Maximum Likelihood idea. The Maximum Likelihood
(ML) methods are systems that are completely based on the ML idea not just the decision subsystem.
Figure 3.1: Tree structure proposed by [7]
3.2.3 Auto Regressive spectrum estimate feature extraction

Mammone et. al. in [27] introduced a method to estimate the carrier frequency, modulation type and bit rate of a set of signals. The method used to classify the modulation type was based on parametric spectrum estimate technique’s. An auto regressive estimate of the spectrum is used to extract features of the signal such as instantaneous bandwidth or instantaneous frequency. The algorithm was designed to distinguish between CW, BPSK, QPSK, BFSK and QFSK. The performance of the algorithm was high $P_{cc} = 99.9\%$ for a CNR of 15dB. This is not the case of ESA, where besides of having just 5 different modulations, each modulation has different parameters that modifies the spectral shape, in addition there are ESA modulations with very similar spectral shape. This spectral similarities decrease the performance of this method. In addition the use of an auto regressive parametric spectrum estimate, a minimum knowledge of the spectrum shape is required.

3.2.4 Statistical moments

Soliman et. al. in [21] used the statistical moments of the signal phase to classify different MPSK modulations. They discovered that for a MPSK signal the $n$th moment ($n$ event) is a monotonic increasing function on M. They used this characteristic in order to classify the modulations. The $P_{cc}$ depends on the $n$th moment used to classify and also on the length of the signal frame used to compute the $n$th moment. On average the $P_{cc}$ was 95% for a CNR of 1dB. This method is useless in ESA missions because the only MPSK like modulation used is the OQPSK.

3.2.5 Artificial Neuronal Network decision system

In [26] a ANN is used to classify, depending on the extracted feature, the modulation of the incoming signals. The ANN was trained to distinguish between 2ASK, 4ASK, 2FSK, 4FSK, BPSK, QPSK, 16QAM and 64QAM. The $P_{cc}$ was on average 96% for a SNR of 8dB.
3.2.6 Maximum Likelihood decision system

In [46] the decision subsystem is based on the ML criteria. They claim that a ML decision system avoids the hard decision\(^1\) and then the performance increases. The basic operation of the ML decision system is to, concerning the extracted features, decide for the most likely modulation. In order to decide which values of the parameters are more likely for a specific modulation the system is trained.

**The training** The training of the system consist on a set of exhaustive simulations. Each modulation scheme is simulated and contaminated with AWGN, then the features are extracted. This process is done many times under different conditions\(^2\). Finally we have a big set of features values, extracted from the different modulations under different conditions. As a consequence is it possible to build a histogram\(^3\) for each feature under different modulations and conditions. The results are stored in a data base. For a particular modulation and/or particular conditions we have an estimation of the pdf for each one of the features.

\[
\text{pdf}_{\text{Pi,\overrightarrow{\text{cond}},m}}(x_i) = DB(\text{Pi,\overrightarrow{\text{cond}},m}) \quad (3.4)
\]

in the above Equation the DB data base is a 3D matrix\(^4\) where the first parameter Pi is the parameter index, the second parameter cond is the conditions and m is the modulation type. Then \(\text{pdf}_{\text{Pi,\overrightarrow{\text{cond}},m}}(x_i)\) is the pdf of the parameter Pi of the modulation m under the conditions cond. The Data base can be build for different conditions. As a consequence if some conditions of the signal or the scenario are known a more precise pdf can be used for the ML decision system.

**System Description** Once the system is trained\(^5\) and the features extracted. The system has to decide which of the candidate modulations is the most likely (ML). The individual probability of each parameter value is calculated for each modulation, afterward the overall probability for the parameter set for each modulation is compared.

The provability of for the parameter Pi of having the value \(x_i\) under the hypothesis of modulation m and conditions \(\overrightarrow{\text{cond}}\) is:

\[
P_{\text{Pi,\overrightarrow{\text{cond}},m}}(Pi = x_i) = \text{pdf}_{\text{Pi,\overrightarrow{\text{cond}},m}}(x_i) = DB(\text{Pi,\overrightarrow{\text{cond}},m}) \quad (3.5)
\]

Then the overall probability of the modulation m for a input parameters set \(PN\) with values \(\overrightarrow{x}\) where \(Pi = x_i\) is:

\[
P(m, \overrightarrow{\text{cond}}) = \prod_{i=1}^{N} P_{\text{Pi,\overrightarrow{\text{cond}},m}}(Pi = x_i) = \prod_{i=1}^{N} \text{pdf}_{\text{Pi,\overrightarrow{\text{cond}},m}}(x_i) \quad (3.6)
\]

\(^1\)The hard decision is one of the known problems of the three decision subsystems. The wrong estimation of a single parameter produces a wrong classification of the modulation.

\(^2\)The different conditions are still to be defined but they could be: different levels of SNR, modulation index, bit rate, etc.

\(^3\)here the histogram works as a probability density function (pdf).

\(^4\)concerning on the definition of \(\overrightarrow{\text{cond}}\) the DB will habe more dimensions

\(^5\)For each extracted feature, the system has (because it has been trained) a pdf function of the probability of the feature values for each modulation.
To decide for the most likely modulation the different $P(m,\text{cond})$ are compared and the modulation with maximum probability is selected.

$$\text{Modulation} = \arg\max_m (P(m,\text{cond}))$$  \hspace{1cm} (3.7)

In the figure 3.3 there is the scheme of the proposed system. The two known problems of the PR methods are the hard decision [46] and the difficulty to extract good feature estimates in conditions of low SNR [43]. The PR with ML decision solves the problem of the hard decision because the final decision is weighted between all the features. However the problem of feature extraction in low SNR conditions is not still solved. Further simulations are required in order to evaluate the performance of the system are needed.

### 3.2.7 Conclusion

Design a modulation classifier for a Satellite scenario is a hard task with high requirements. In one hand the system must have an acceptable performance in scenarios with a very low SNR (0 to 5 dB depending on the mission requirements [45]), the $P_{cc}$ of the system must be in an acceptable range. On the other hand, the most important feature that has to be taken into account during the implementation/design of the AR is degree of blindness of the MC. An other important part of the design is the decision for a suitable features set to be extracted to characterize the incoming signal, the choice of the features set depend on the
different possible incoming modulations. The main aspects that drive the PR performance are

- Its difficulty in extracting good features in situations of low SNR. As a consequence the performance of the overall systems decreases.

- The number of modulations that the system should be able of distinguish and the capacity on finding features capable to distinguish those modulations.

- The tree decision structure works in hard-decision manner, that is: when a branch is decided is it impossible to come back and go through the other branch. Which means that the bad estimation of one single feature will lead to a misclassification of the modulation, that’s the reason why most of the methods require a high SNR.

One of the characteristics to exploit is the reduced set of possible incoming signals in the ESA missions (about 5 different modulations). That small range of different modulations is the characteristic that might make our Satellite modulation classifier work under low SNR. Since the modulations that we want to classify are different from the ones introduced in the previous sections, new features must be decided, then the performance of the system will be also different.

### 3.3 Maximum likelihood classification (ML)

The ML approach is valid when the number of modulation candidates is finite and is based on hypotheses testing. The complexity of the ML based classifier depends on the number of unknown parameters that are associated with the received waveform and the classifier. Usually this method leads to a set of 1 by 1 maximum likelihood comparisons working in a cascade scheme. If we assume that $s(t)$ is the transmitted signal and that $r(t)$ is the received signal. Signals are related:

$$ r(t) = s(t) + n(t) $$  \hspace{1cm} (3.8)

where $n(t)$ is AWGN. Then the logarithmic likelihood function (LLF) of receiving $r(t)$ when $s(t)$ is sent is:

$$ LLF(r(t)|s(t)) = \ln \left( e^{\frac{2}{N_0} \int_0^{NT_0} r(t) \cdot s(t) \, dt} \right) $$  \hspace{1cm} (3.9)

The ML methods classify the incoming modulation to be the more likely (maximum likelihood function) modulation. For example: if we want to know how likely is the incoming signal to be a BPSK modulation we should calculate the $LLF(r(t)|s(t))$. Since the exact shape of $s(t)$ is not known\(^1\) it is not possible to get the exact value of $LLF(r(t)|s(t))$. Similarly parameters like the timing offset $\epsilon$ or the carrier phase $\theta_c$ are also not known, and it is impossible to calculate the exact value for $LLF(r(t)|s(t))$. One way to solve this problem is to take the expectation values of the LLF:

$$ l(r(t)) = E_r(E_{\theta_c}(E_{\theta_h}(LLF(r(t)|s(t))))) $$  \hspace{1cm} (3.10)

\(^1\)To know the exact shape of $s(t)$ it is need to know about all the characteristics of the signal, that is the exact data vector, the data rate, modulation parameters etc.
or to estimate the unknown parameters:

\[ l(r(t)) = \ln \left( e^{\frac{r(t)}{N_0}} \int_0^{N_T} r(t) \cdot s(t, \tilde{c}, \tilde{\theta}_c) \, dt \right) \]  

(3.11)

Both methods suppose that the statistical values of the data vector \( \theta_k \) are known. Once the probability of all the possible modulations is computed, one can compare them and decide which modulation is the more likely to have been transmitted. The following methods shown different approaches and simplifications in order to compute the LLF.

### 3.3.1 Two hypothesis MPSK method

Polydoros et al. in [23] developed an algorithm based on the likelihood function to classify different MPSK signals contaminated with AWGN. The main idea of ML is to compute the LLF of the different hypothesis \( H_i \) and then compare them in order to decide for the most likely modulation. For example to distinguish between a binary hypothesis (MPSK vs. MPSK') we use the following equation:

\[
LLF_{MPSK} - LLF_{MPSK'} \leq M_{PSK'} \text{ Threshold}
\]  

(3.12)

where \( \text{Threshold} \) is obtained previously in order to minimize the misclassification probability. Because [3.12] is a complicated equation, it is simplified using some assumptions.

Three different scenarios for the \( LLF \) are compared,

1. **coherent and synchronous environment** (CSE), wherein the phase \( \theta_c \) and the symbol timing offset \( \epsilon \) are known.

2. **noncoherent and asynchronous environment** (NCAE) both of them (\( \theta_c, \epsilon \)) are unknown and modeled as random variables

3. **noncoherent and synchronous environment** (NCSE) is an intermediate case where it is assumed that \( \epsilon = 0 \) and \( \theta_c \) is a random variable.

The three different models (CSE, NCAE and NCSE) where implemented in [17] and tested in order to distinguish between BPSK / QPSK and 8PSK / 16PSK. The simulation results shown in [17] that the performance depends on the two hypothesis of the comparison (see Equation 3.12). Table 3.2 shows the minimum SNR value to achieve a performance of 95%, it can be seen that since the order of the MPSK increases the minimum SNR needed to detect the modulation also increases.

<table>
<thead>
<tr>
<th></th>
<th>BPSK / QPSK</th>
<th>QPSK / 8PSK</th>
<th>8PSK / 16PSK</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSE</td>
<td>-4dB</td>
<td>4.5dB</td>
<td>12dB</td>
</tr>
<tr>
<td>NCSE</td>
<td>-3dB</td>
<td>5.5dB</td>
<td>13dB</td>
</tr>
<tr>
<td>NCAE</td>
<td>-1.5dB</td>
<td>7dB</td>
<td></td>
</tr>
</tbody>
</table>

**Table 3.2:** Minimum SNR to have a \( P_{cc} \) of 95%, extracted from [17]
3.3.2 Three hypothesis ML method

Polydoros et. al. proposed in [28] a multiple hypothesis maximum likelihood method. The different hypothesis are BPSK, QPSK and OQPSK. A new way to compute the comparison threshold is introduced and also an estimate for $E_c$ and $N_{0,c}$ (carrier power and noise power) is used to make an autonomous multiple hypothesis modulation classifier. The algorithm achieves a $P_{cc}$ of 95% at a SNR of -2dB.

3.3.3 Theoretical Results

In [37] the theoretical upper bound for the MLMC (Maximum Likelihood Modulation Classifier) performance is developed. Theoretical expressions for the $P_{cc}$ are also developed. Finally simulation results are compared with the theory. The simulation results match the theory for SNR higher than 0dB. The minimum SNR to obtain a 95% of $P_{cc}$ is between 2 and 6dB (concerning on the data length) for the 8PSK vs. V.29 comparison. For the comparison star-QAM vs. 8PSK the minimum SNR is to achieve a 95% of $P_{cc}$ is between 0 and 4 dB. Finally, when classifying between V.29 and star-QAM, to achieve a $P_{cc}$ of 95% a SNR of 4-10dB is needed.

3.3.4 Suboptimal classifier

Soliman et. al. presented in [43] presented a suboptimal algorithm for the classification of MPSK modulations. The suggested algorithm uses a MAP criteria by comparing the LF of the signal phase $\psi(t)$. The simulation results show that a $P_{cc}$ of 95% is achieved with a minimum SNR of -5dB, when classifying between a BPSK vs. QPSK.

3.3.5 Various publications

1. Boiteau and Martret in [36], presented the general maximum likelihood classifier (GMLC) that can be applied to all modulations without restriction of pulse function. No simulations are reported.

2. Kim and Weber in [49] introduced a method to classify modulations with different Spectral Correlation (SPCR). The classifier is not affected by the a priori probabilities, but modulations with similar spectral shapes are not correctly classified.

3. Polydoros et. al. in [41] present a similar paper to [23] they compare 16QAM, 16PSK and V.29 under different scenarios and with different simplifications to the LLF.

3.3.6 Conclusion

The performance for the ML methods are higher than the PR methods, that allows this methods to work under low SNR conditions. However, the performance of the ML is

\[ r(t) = s(t) + n(t) \]

Here the pdf of the signal phase $\psi(t) = \theta_M(i) + \zeta(i)$ is used to decide for the most likely modulation.
directly dependent on the pair of modulations of the hypothesis. For instance, in [23] the performance decreases 12dB depending on the pair of modulations to be detected (see Table 3.2).

The knowledge about the input signal is also important, since there are algorithms to detect signals without knowledge of the symbol timing ($\epsilon$) or the carrier phase ($\theta_c$), but their performance is lower when the algorithms have a more precise knowledge of the signal parameters. Due to the ML MC methods are modulation dependent, it is difficult to estimate the performance of ML methods under untested modulations. As a consequence to check if the ML methods can classify the modulations of ESA (see Appendix A) under the special conditions of a satellite channel (see Section 1.3). Simulations under these conditions are needed.

### 3.4 Comparison of the different methods

In the previous section we haven presented a set of different publications and methods to the automatic classification of modulation. Since we want to apply the MC methods to an AR working in a Satellite scenario special considerations must being taken in to account:

1. MC must be designed to work under low SNR conditions.
2. Dependence of the MC on unknown / estimate parameters of the signal such as symbol timing ($\epsilon$), the carrier phase ($\theta_c$) or the SNR is very important in an AR applications (see Section 2).
3. Small set of possible configurations and modulations of the incoming signal.
4. Big set of possible modulation parameters.
5. Very high computational power available in the ground stations.

Taking those conditions into account, we conclude that non of the current MC methods matches the ESA requirements. The ML methods have a good performance but they need additional information about the signal (i.e. $T, \theta_k$, etc.) before making an estimation of the signal. In addition, all this information could not be available in the first iterations of the autonomous receiver (AR). In the other hand, the PR methods have worse performance than the ML. However its lack of performance is compensated with the capability of making blind classifications (without needing additional parameters estimation). It may seem that there is no solution to our MC problem, because none of the methods accomplishes the requirements. Nevertheless in Chapter 4 we purpose an Hybrid modulation classifier that combines the PR and the ML methods in a multiple stage classification.
### Modulation Classifiers

<table>
<thead>
<tr>
<th>Pattern Recognition</th>
<th>Maximum Likelihood</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Feature extraction</strong></td>
<td><strong>Decision Sub system</strong></td>
</tr>
<tr>
<td>• Decide for a suitable parameter/feature set. ❌</td>
<td>• Tree structure</td>
</tr>
<tr>
<td>• Under low SNR parameter extraction becomes hard. ❌</td>
<td>• Low performance. ❌</td>
</tr>
<tr>
<td>• Blind classification: small or non additional knowledge of the signal is required. ❌</td>
<td>• Hard decision: wrong estimation of single parameter, bad estimation of the modulation ❌</td>
</tr>
<tr>
<td>• Artificial Neuronal N.</td>
<td>• Avoid hard decision. ❌</td>
</tr>
<tr>
<td>• Avoid hard decision. ❌</td>
<td>• Low performance ❌</td>
</tr>
<tr>
<td>• Maximum Likelihood</td>
<td>• No simulations are done. ❌</td>
</tr>
<tr>
<td>• Avoid hard decision. ❌</td>
<td>• system need to be trained. ❌</td>
</tr>
<tr>
<td>• No simulations are done. ❌</td>
<td></td>
</tr>
<tr>
<td>• system need to be trained. ❌</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.3: Overview of the different trade off for the MC methods
Hybrid Modulation Classifier for satellite telemetry signals

In this Chapter an algorithm to distinguish the modulation scheme of the different telemetry signals is introduced. The algorithm proposed is designed to distinguish between the different types of ESA telemetry modulation (see Appendix A). It uses a combination of the different techniques of modulation classification, in addition it has 2 classification steps. However, just the first module of the algorithm has been fully implemented and simulated. The first module consist on an Analog vs. Digital classifier. In the following a general description of the proposed modulation classification (MC) is shown.

4.1 Description of the satellite scenario and state of the art

4.1.1 State of the art

In Section 3 the different methods for the MC have been analyzed, in Table 3.3 an overview of the trade off between the different methods is shown. Summarizing there are two main approaches for the MC:

- In the pattern recognition (PR) approach the classifier is composed of two subsystem: the feature extractor sub system with the role of extracting features or characteristics of the signal and the pattern recognizer or the decision sub system which from the extracted features classifies the signal modulation.

- In the Maximum likelihood (ML) approaches the received signal is compared with the different possible modulations and the most likely is chosen.

This different methods have present different trade offs, there are some versions of the PR that are nearly blind but on the contrary they have low performance. In contrast ML methods have better performance but they need additional information about the signal before making a classification.

Our conclusion is that just the ML methods have an acceptable performance in terms of SNR nevertheless they are designed to distinguish between a small group of modulations (commonly 2) and to work properly they need to estimate additional parameters of
the signal, such as the symbol-rate \( br \), the SNR and the symbol-timing \( \epsilon \). However this parameters could not be available for the MC module (at least in the first iterations of the Autonomous Receiver).

In contrast, the PR approach presents a very low performance, however its performance is highly scenario dependent (depends on the type and number of modulations schemes to classify).

4.1.2 Satellite scenario general requirements

ESA is supporting a broad range of different mission types (see Appendix B). Mission in Saturn such as Cassini with very low signal to noise ratio, or satellites that are in different orbits around the Earth (with better signal to noise ratio), the modulation scheme used depends on every mission, but it also depends on the operation mode of the satellite. The requirements of the different mission are briefly explained in Appendix B but for more detailed information please refer to [45].

Due to special nature of the satellite scenarios (see Chapter 1), our requirements for a MC system are different from the requirements of the other published MC systems, presented in the Chapter 3.

In the following there is an overview of the general requirements:

- **Signal to noise ratio (SNR):** the algorithm should have an acceptable performance in scenarios with SNR above 0dB.

- **Frequency offset (\( \gamma \)):** has a very important role. We will require to have an acceptable performance in scenarios with Doppler effect up to \( \pm 400 kHz \).

- **Modulation scheme:** Since to there are different modes of operation, and different missions the different modulations schemes are: GMSK, OQPSK, PM on carrier, PM on square sub carrier and PM on sinusoidal sub carrier (see Appendix A for more details).

- **Convergence time:** because of the limited time of visibility of the satellites, the algorithm should be fast enough to acquire and to estimate the modulation scheme in a reasonable short time.

4.1.3 Design of the algorithm: challenging design parameters

From the PR approach point of view the most challenging design parameter is to achieve an acceptable performance, in terms of correct modulation classification probability, under low SNR values. For example in [30] the probability of correct classification achieves the 95% at a SNR of 10 dB, in [27] a SNR of 15 dB is need to achieve a performance of 96%, the other PR methods have similar performances. In our case we need an acceptable performance at SNR arround 0 dB which is 10dB less that the best pattern recognition algorithm.
From the ML algorithm point of view the most challenging parameter is to recognize 5 different modulations schemes without much a priori knowledge of the signal parameters. Designing estimation methods within an AR architecture the lack of autonomy of the ML methods, is a restrictive factor to take into account. Since it will be decisive for the final order of the different estimation modules of the AR architectural design. However, is it known (see Section 2) that a common architecture of the AR, consist on having different estimation methods for the same parameter using the blind estimation modules during the first algorithm iterations and later on use other methods that require more signal information. We have an interesting trade off where to achieve good performance we need to use ML algorithms but to make a blind estimation we need to use the PR methods.

4.2 Hybrid modulation classifier (MC) for satellite telemetry signals

In the Sections 4.1.1 we have seen a small overview of the different methods for the modulation classification, however none of them matches our requirements (see Section 4.1.2). For this reason we have decided to implement an Hybrid system that will combine both techniques, the PR and the ML modulation classification algorithm. The main idea is to make a coarse Modulation classification during the first iterations of the AR. Next, when the different AR modules have produced estimations of the signal parameters, the ML will be ready to be run. With this hybrid architecture we exploit the best characteristic of the PR (blind) and the best characteristic of the ML (performance).

4.2.1 Design of the algorithm

We can classify the telemetry modulations (see Appendix A) in two big groups, the modulations with carrier (or sub carrier) and the modulations with a suppressed carrier. Similary we can group the modulations in to Digital Modulations and Analog Modulations, even if all of them are generated digitally.

- Analog Modulations: with carrier or subcarrier
  - PM on carrier (SP-L/PM)
  - PM on sin subcarrier (PCM/PSK(sine)/PM)
  - PM on square subcarrier (PCM/PSK(square)/PM)

- Digital Modulations: supress carrier modulations
  - OQPSK (Offset Quadrature Phase Shift Keying)
  - GMSK (Gaussian Minimum Shift Keying)

Looking in to the spectra of the different signals (see Figures 4.1 and 4.2) we can see that the power of the Analog modulations is split between the carrier (sub carrier) power and the data power. The data produces two lobes around the peak of the carrier (sub carrier). On the contrary in the Digital modulations case the power is located altogether forming one main lobe around the 0 Hz (assuming the signal in baseband). This difference in the spectra shapes is one characteristic (feature) that will be exploited to make a first
(general) classification of the signal. For this reason the very first step of our modulation classification algorithm will be the classification of the signal in two groups: analog and digital signals. Owning to this evident spectral characteristic and the small number of modulations (two modulation types: analog and digital) we consider that this classifier can be implemented with a PR method and achieve an acceptable performance. In addition, because of the blind characteristic of a PR we can use the analog vs. digital classifier in the early iterations of the AR.

In the second step of the algorithm we have the advantage of having fewer different modulations 2 in the case of the Digital modulations and 3 in the case of Analog modulations. In addition this second step will be different for the digital and analog signals and we can assume that the other estimation modules of the AR have estimated some additional parameters that could be used by the ML algorithm. In Figure 4.3 there is the basic scheme of the algorithm proposed, consisting in 2 steps, a first classification between analog and digital signals and a second step to distinguish inside these two groups. In the following we enter in to the detail of the design of the different steps of the MC algorithm.

![Figure 4.1: Spectra of the different carrier modulations](image)

(a) PM on square sub carrier

(b) PM on carrier
Figure 4.2: Spectra of the different suppressed carrier modulations
4.2.1.1 Coarse classification: Analog vs. Digital (spectral PR)

This classifier will be designed to classify the incoming signal into Analog (carrier or subcarrier) and Digital (suppressed carrier) modulated. We want to design the classifier to be a blind classifier, in other words the classifier should make the estimation without using the estimates of the other AR modules. In order to classify the signal we will use a PR classifier that will work in the frequency domain. More precisely the PR will extract characteristics of the spectrum correlation of the different signals. In [49] and in [14] the correlation of the spectrum is used as a classification feature. Knowing that the spectra of our signals have differences (between analog signals and digital signals) and with the motivation of [49] and [14], we decide to work in the correlation spectra domain to make the classification of the signal. Working in the correlation spectra domain is currently used in the ESA modem IFMS\(^1\) to find the symmetries between signals [6]. Alternatively in [17] a modulation index $\beta$ estimator to distinguish between analog and digital modulations is used. This idea is similar to our idea, because we want to exploit the spectral differences (mainly the presence of data lobes in the analog modulations) and the modulation index is a ratio between the data power and carrier power. In other words both techniques exploit the presence of two data lobes (analog modulations) against the presence of a single data lobe.

Let’s define the received signal $y(n)$

$$y(n) = s(n) + n(n)$$

where $s(n)$ is the transmitted signal and $n(n)$ is the uncorrelated white Gaussian noise. If we assume that $s(n)$ is stationary with $E\{s(n)\} = 0$, $n(n)$ is stationary with $E\{n(n)\} = 0$,
variance \( E\{n(n)n(n)^*\} = \sigma^2 \delta(n) \). The correlation operand \( C_{x,x}(n) \) is defined as follows for stationary signals:

\[
C_{x,x}(n) = E\{x(k) \cdot x(k-n)^*\} \tag{4.2}
\]

then the cross correlation between \( s(n) \) and \( n(n) \) is \( C_{s,n}(n) = 0 \) and the correlation of \( y(n) \) is

\[
C_{y,y}(n) = C_{s,s}(n) + C_{n,n}(n) \tag{4.3}
\]

then the spectrum is

\[
S_{y,y}(f) = \mathcal{F}(C_{y,y}(n)) = S_{s,s}(f) + S_{n,n}(f) \tag{4.4}
\]

where \( S_{s,s}(f) \) is the spectrum of the transmitted signal \( s(n) \) and \( S_{n,n}(f) \) is the spectrum of the noise, notice that we have supposed that there is no cross term between the signal \( s(n) \) and the noise therm \( n(n) \). The noise therm \( n(n) \) is white and Gaussian distributed with a variance \( \sigma^2 \), then the noise spectrum becomes flat with amplitude \( N_0/2 \)

\[
S_{n,n}(f) = \mathcal{F}(C_{n,n}(n)) = \mathcal{F}(E\{n(k) \cdot n(k-n)^*\}) = \mathcal{F}(\sigma^2 \delta(n)) = N_0/2 \tag{4.5}
\]

If we make the correlation of the spectrum \( S_{yy}(f) \) we get,

\[
C_{S_{y,y},S_{y,y}}(n) = E\{S_{y,y}(f) \cdot S_{y,y}(f-n)^H\} \tag{4.6}
\]

however \( S_{y,y}(f) \) has a finite size since it has been computed with a FFT of length \( N \). As a consequence, the length of the correlation \( C_{S_{y,y},S_{y,y}}(n) \), is \( L_c = 2 \times N - 1 \). To compute the exact discrete correlation we have to window the spectrum and then correlate it, if we assume that the spectrum extends from \( f = -N/2 \) to \( f = N/2 \), and that the correlation \( C_{S_{y,y},S_{y,y}}(n) \) extends from \( n = -L_c/2 \) to \( n = L_c/2 \). Then the noise term of the spectrum becomes

\[
S_{n,n}(f) = \frac{N_0}{2} \Pi\left(\frac{f}{N}\right) \tag{4.7}
\]

being \( \Pi\left(\frac{f}{N}\right) \) a pulse function of length \( N \) (centered in 0).

If we assume that the signals are ergodic, that means that one realization has the same characteristic as the mean, the correlation function can be also defined as

\[
C_{S_{y,y},S_{y,y}}(n) = \sum_{k=-N}^{N} S_{y,y}(k) \cdot S_{y,y}(k-n)^* \tag{4.8}
\]
As a consequence we can write the correlation function

\[ C_{S_y,y,y}(n) = \sum_n \{S_{y,y}(f) \cdot S_{y,y}(f - n)^*\} \]

\[ = \sum_f \{(S_{s,s}(f) + S_{n,n}(f)) \cdot (S_{s,s}(f - n)^* + S_{n,n}(f - n)^*)\} \]

\[ = \sum_f \{S_{s,s}(f) \cdot S_{s,s}(f - n)^*\} + \sum_f \left\{\frac{N_0}{2} \Pi\left(\frac{f - n}{N}\right) \cdot S_{s,s}(f)\right\} \]

\[ + \sum_f \left\{\frac{N_0}{2} \Pi\left(\frac{f}{N}\right) \cdot S_{s,s}(f - n)^*\right\} + \sum_f \{S_{n,n}(f) \cdot S_{n,n}(f - n)^*\} \]

\[ = C_{S_{s,s},S_{s,s}}(n) \]

\[ + \sum_f \left\{\frac{N_0}{2} \Pi\left(\frac{f - n}{N}\right) \cdot S_{s,s}(f)\right\} \]

\[ + \sum_f \left\{\frac{N_0}{2} \Pi\left(\frac{f}{N}\right) \cdot S_{s,s}(f - n)^*\right\} \]

\[ + C_{S_{n,n},S_{n,n}}(n) \quad (4.9) \]

where we can expand the second and the third term

\[ C_{S_y,y,y}(n) = C_{S_{s,s},S_{s,s}}(n) + C_{S_{n,n},S_{n,n}}(n) \]

\[ + \frac{N_0}{2} \sum_{f=-\frac{N}{2}}^{\frac{N}{2}-n} S_{s,s}(f)^* \sum_{f=-\frac{N}{2}}^{\frac{N}{2}+n} S_{s,s}(f) \quad (4.10) \]

if the spectrum is real and has even symmetry, we can write

\[ C_{S_y,y,y}(n) = C_{S_{s,s},S_{s,s}}(n) + C_{S_{n,n},S_{n,n}}(n) + 2 \cdot \frac{N_0}{2} \sum_{f=-\frac{N}{2}}^{\frac{N}{2}-n} S_{s,s}(f) \quad (4.11) \]

In Equation 4.11 we have 3 different terms, the first one is the correlation of the data spectrum, the second one is the correlation of the noise spectrum and the third one a cross term between the noise spectrum and the data spectrum. We can still develop the second term,

\[ C_{S_{n,n},S_{n,n}}(k) = \sum_{f=-N}^{N} S_{n,n}(f) \cdot S_{n,n}(f - k)^* \]

\[ = \sum_{f=-N}^{N} \frac{N_0}{2} \Pi\left(\frac{f}{N}\right) \cdot \frac{N_0}{2} \Pi\left(\frac{f - k}{N}\right) \}

\[ = \frac{N_0}{2} \sum_{f=-N}^{N} \Pi\left(\frac{f}{N}\right) \cdot \Pi\left(\frac{f - k}{N}\right) \]

\[ = \frac{N_0}{2} N \Lambda\left(\frac{n}{2N-1}\right) \quad (4.12) \]

so we can see that the noise contribution to the correlation of the spectrum has a triangular shape. The first term of Equation 4.11 is the contribution of the correlation of the
data spectrum to the general correlation. Notice that in Equation 4.9 we have assumed ergodicity, as a consequence we can write this part as

$$C_{S_{s,s}S_{s,s}}(n) = \sum_{f=-N}^{N} S_{s,s}(f) \cdot S_{s,s}(f-n)^* \quad (4.13)$$

Finally, using Equations 4.13 and 4.12 the expression for Equation 4.11 is

$$C_{S_{y,y}S_{y,y}}(n) = \sum_{f=-N}^{N} S_{s,s}(f) \cdot S_{s,s}(f-n) + \frac{N_0}{2} N \Lambda \left(\frac{n}{2N-1}\right) + 2 \cdot \frac{N_0}{2} \sum_{f=-N}^{N-n} S_{s,s}(f) \quad (4.14)$$

From the above Equation 4.14 we can see that the correlation of the spectrum of the received signal is composed by three components, one is the correlation of the data spectrum, the second is the noise contribution and the third is the cross term between the data spectrum and noise spectrum.

Let’s focus on the shape of the correlation of the data spectrum. The correlation will be maximum at the origin (0 shift), but the secondary lobes (the lobes where the data is) will produce a second maximum in the correlation function.

In Figure 4.4 is shown the graphical correlation between the spectrum of two PM on square subcarrier modulations, where it can be seen that the presence of data lobes produce peaks in the correlation of the spectrum. The different plots of the Figure 4.4 show the correlation operation at different shifts between signals (different values of $n$), notice that when the shift value is 0 both signals are confronted, while $n$ is changing there is less part of the signals confronted, but for certain values of $n$ the data lobes of the spectrum are confronted, that confrontation produces peaks in the correlation of the spectrum.

In Figure 4.5 there are the plots of the different correlation functions of the analogical spectrum, one can see that all of them have similar shape: the presence of a strong peak (global maximum) in the center and two secondary peaks (local maximum) around the central peak. On the contrary the Digital modulations (see Figure 4.6) correlation will produce only a big peak. This difference in the correlation of the spectrum will be used to classify the signals into analog or digital.

In conclusion, to distinguish between analog and digital modulations, first we will compute the Spectrum estimation of the signal, next we will compute the correlation of the spectrum, finally a peak detector will check out if there exist secondary peaks (local maximums) in the correlation spectrum. If they exist the signal will be classified as analog and if not the signal will be classified as digital. In Figure 4.7 is shown the basic scheme for the analog vs. digital modulation classification.
Figure 4.4: Example of the spectral correlation of two PM on square sub carrier modulations, notice that the maximum occurs when both signals are totally overlapped, but the second maximums occurs when the data lobes of the signal are overlapped.
Figure 4.5: Correlation of the spectrum of the different analog signals. Parameters modulation index $\beta = 0.7$, frequency sub carrier $f_{sc} = 500\text{kbps}$, factor of bit rate 4, bit rate (on carrier modulation) $512\text{kbps}$

Figure 4.6: Correlation of the spectrum of the digital modulations. Parameters of the modulations: symbol-rate $512\text{kbps}$, $\alpha = 0.1$, $bt = 0.5$

Figure 4.7: Basic scheme of the analog vs. digital modulation classifier
4.2.1.2 Fine classification: Analog vs Analog

Once we know that the signal is analog, we have reduced the number of possible modulations to two:

- PM on carrier (SP-L/PM)
- PM on sin subcarrier (PCM/PSK(sine)/PM)
- PM on square subcarrier (PCM/PSK(square)/PM)

It is difficult to distinguish between the two subcarrier modulations with a feature extraction algorithm is difficult because they have very similar spectra shapes and also similar temporal characteristics (since the only difference is the shape of the sub carrier wave (see Appendix A)). However we propose the following parameters:

- In [3] is suggested to use time domain parameters such as the phase linearity to distinguish between the sharp changes in the phase due to a square sub carrier and the smooth changes produced by a sinusoidal subcarrier.
- The normalized instantaneous frequency is proposed in [7] and [8]
- The instantaneous frequency proposed in [39] can be also useful to distinguish between the two subcarrier modulations.

One of the commented problems of the features extraction algorithms is that in low SNR scenarios some times is hard to extract features, some other times they are corrupted by the noise, here if we combine the extraction of the different features, we can use a Maximum likelihood decision system (see Section 3.2.6 and [46]), unfortunately as far as the authors knowledge no ML decision system has been tested, and no features like the proposed has been tested with subcarrier modulations. For this reason we encourage the use of a ML Estimator introduced in [17] to distinguish between the different subcarrier modulations, the ML parameters (result of the ML comparison) can be also used as a parameter (feature) for the final decision.

Since of the analog vs. analog classifier is in the second step of the MC, we can assume that additional estimated parameters of the signal will be available from the other modules, then the use of a MC is possible. To detect the PM on carrier modulation we can use feature extraction methods to exploit the absence of subcarrier.

4.2.1.3 Fine classification: Digital vs. Digital

Once we know that the signal is digital, we have reduced the number of possible modulations to two:

- GMSK
- OQPSK

In [3] the propose to check envelop constancy (defined in [7]) in order to distinguish between filtered modulation schemes from unfiltered. Since the OQPSK modulation is filtered by a raised cosine filter it should present no constant amplitude\(^1\), on the contrary the GMSK

\(^{1}\)Besides of the OQPSK is supposed to have no constant amplitude, the offset effect smooths the envelope to almost constant.
has a constant amplitude\(^1\). Additionally in [39] propose the use of the instantaneous frequency as a characterization for the GMSK against a OQPSK. The problem of feature extraction algorithms is again its performance at low SNR values, in addition the OQPSK has almost constant amplitude. As a consequence we don’t believe that it will be possible to classify the digital modulations with a PR method, unless we find any other parameter which differs between both modulations. For this reason we propose to use a Maximum likelihood classifier to distinguish between the GMSK and the OQPSK. The use of a ML classifier assumes that the estimations of the other AR modules are available.

### 4.2.2 Implementation of the Analog vs. Digital classifier

The analog vs. Digital classifier consists on 3 steps (see Figure [4.7](#))

- Compute the spectrum of the signal
- Make the correlation of the spectrum
- Check if there exist local maximums around the global maximum

In this section we will enter in to detail in how this different steps have been implemented.

#### 4.2.2.1 Spectrum estimation of the incoming signal

We have used the Welch Spectrum estimate in order to estimate the spectrum of the signal, the Welch spectrum estimate is a well known spectrum estimate and is broadly used in different applications, it mainly consist on averaging different periodograms\(^2\). For more information about parametric spectrum estimate refer to Section [6.2.1.1](#) or to [40](#) and [55](#).

From now on the Welch spectrum estimate of the signal \(y(t)\) is denoted with the symbol \(\hat{C}_{Wy,y}(e^{jw})\).

#### 4.2.2.2 Spectrum correlation

The next step, before computing the spectrum estimation \(\hat{C}_{Wy,y}(e^{jw})\) of the incoming signal \(y(n)\) is to compute the autocorrelation of the signal, from Equation [4.13](#) we know that

\[
C_{\hat{C}_{Wy,y}, \hat{C}_{Wy,y}}(n) = \sum_{k=-N}^{N} \hat{C}_{Wy,y}^{*}(k) \cdot \hat{C}_{Wy,y}(k-n) \quad (4.15)
\]

One of the requirements for the modulation classifier is to be robust against frequency offset (Doppler). As a consequence, we propose to modify the way to compute the correlation in order to increase its robustness.

---

\(^1\) Notice here that the GMSK is filtered by a Gaussian filter, however is the data of a GMSK that is Gaussian filtered not the modulated data itself for this reason the GMSK has constant envelope [42].

\(^2\) The periodogram is basic non-parametric spectrum estimate, it is similar to estimate the spectrum as the square of the FFT transform of the windowed signal.
The Doppler effect shifts the baseband signal out of the center frequency, to avoid this shift effect we have to look in the spectrum for the exact position of the data (or the carrier) because we can not be sure that the data (or carrier) is at 0 frequency. For this reason we suggest to make the cross correlation of the spectrum \( \hat{C}_{\hat{Y},\hat{y}}(e^{jw}) \) with its flipped version.

With this method we ensure that the position of the maximum in the correlation will correspond to the position of the data (carrier) of the signal. The maximum of the spectrum correlation corresponds also with the point of symmetry of the spectrum. Mathematically

\[
Cf_{\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}}(n) = \sum_{k=-N}^{N} \hat{C}_{\hat{Y},\hat{y}}(n + k) \cdot \hat{C}_{\hat{Y},\hat{y}}(n - k)^* \tag{4.16}
\]

The next step should be to look for local maximum next to the global maximum, however in the presence of noise the spectrum becomes rough, and the local maximum search algorithm might not work properly. Here we propose the use of an algorithm that smooths the correlation function \( Cf_{\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}}(n) \) by averaging different correlation functions and using smooth filters. Each correlation is computed in the following way: first the cross correlation between the spectrum \( \hat{C}_{\hat{Y},\hat{y}}(e^{jw}) \) of \( y(t) \) and the flip version of the spectrum is computed (see Equation 4.16). After that the maximum position is detected, and the maximum peak is deleted and replaced by a smooth peak\(^1\), after that the correlation function is smoothed by a hamming filter \( h(n) \). If we call the delete-peak the function that finds the global maximum and replaces it by a smooth filter then the correlation function after the deleting the peak is

\[
Cfp_{\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}}(n) = \text{delete-peak}(Cf_{\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}}(n)) \tag{4.17}
\]

next \( Cfp_{\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}} \) is filtered by a hamming filter \( h(n) \)

\[
Cs_{\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}}(n) = h(n) \ast Cfp_{\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}}(n) = h(n) \ast \text{delete-peak}(Cf_{\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}}(n)) \tag{4.18}
\]

In Figure 4.8 there is the scheme of this correlation method. After that we average many correlations to smooth and reduce the variance of the final correlation estimation.

Finally we average over different correlation estimations \( Cs_{\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}}(n) \) in order to reduce the variance and smooth the correlation

\[
\widehat{Cav}_{\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}}(n) = \frac{1}{I} \sum_{i=1}^{I} Cs_{i\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}}(n) \tag{4.19}
\]

where \( I \) is the total number of averaged functions, and \( Cs_{i\hat{C}_{\hat{Y},\hat{y}}\hat{C}_{\hat{Y},\hat{y}}} \) is the \( i \)'th correlation estimation (see Equation 4.18) of a specific signal \( y(t) \) during the \( i \)'th time slot.

\(^1\)the objective of deleting the peak is prepare the correlation signal for the smoothing filter (if the central peak is very big, then the secondary peaks can be covered by the big one after smoothing)
4.2.2.3 Search local maximums algorithm

Once we have obtained a low variance \( \hat{C}_{y,y} \), \( \hat{C}_{Wy,y} \), and \( \hat{C}_{Wy,y} \) and smoothed version of the spectral correlation \( \hat{C}_{Wy,y} \) of the incoming signal \( y(t) \) we want to detect the presence or absence of local maximums around the global maximum. Notice that we want to detect the local maximums that corresponds to the data lobes. In Figure 4.9 the wanted local maximums are marked with a red circle, notice that besides of having different amplitude all of them are in the same position and symmetric to the other frequency. The shape of the spectrum correlation function depends on many factors such as:

- **Noise level**: we can see in Equation 4.14 that the noise level has influence in the correlation. In Figure 4.9 we can see that the level of the correlation signal depends on the noise level. The higher the noise level is the higher the correlation amplitude became.

- **Frequency of the sub carrier**: the frequency of the sub carrier \( f_{sc} \) is important because the position of the secondary peaks of the correlation function depends on \( f_{sc} \), if \( f_{sc} \) is too low then the secondary peaks may be mask by the central peak. In Figure 4.9 the local maximums are in the same position besides of having different levels of noise, that indicates that the position of the local maximum depends only on the \( f_{sc} \). For the PM on carrier modulations a similar effect occurs with the symbol-rate \( br \).

- **Modulation index \( \beta \)**: the modulation index is an index that controls the ratio of power between the carrier and the data, the bigger the power allocated in the data is the bigger the secondary peaks will be (see Figure 4.10).

- **The bit rate is proportional to the width of the lobes, and in the PM on carrier case, to the position of the secondary peaks of the spectral correlation** (see Figure 4.12).

The correlation function is formed by mainly by two components (see Equation 4.14) the first one is a triangular shape that depends on the noise power and the second one is the correlation of the data. In Figure 4.9 can be seen the effect of the noise, observe how while the noise level decreases the shape of the correlation has less triangular shape. The search local maximum block, looks for all the candidates to be local maximums, and then using different criteria discards the candidates. If at the end of this algorithm there are remaining candidates, then the incoming signal is estimated as an Analog signal, if there are no remaining candidates to be local maximum then the modulation is estimated as a Digital signal. A block diagram of the local maximum detector is shown in Figure 4.13.

In the following the different blocks of the algorithm are briefly commented.
**Max detector**  this block looks for local maximum points in the correlation function. It checks every point if is bigger than a certain window of neighbor points. If it is bigger then is stored in the local maximum candidate list.

**Center**  this block centers the candidates to local maximums around the central peak (point of symmetry or global maximum). If the candidates are very off centered then it discards the candidates and estimates the signal to be digital. The number of candidates after this block is always odd (symmetry point + the same number of candidates in each side of the symmetry point)

**Max descendant**  The different max candidates should have decreasing amplitudes, if there is one candidate that has more amplitude that other candidates that are closer to the symmetry point, then this closer candidates are discarded (see Figures 4.9 and 4.10 to check for the decreasing amplitudes).

**Flat detector**  some times the algorithm detects local maximums in to the flat areas of the digital spectrum correlations (see Figure 4.11), this block checks if the found maximums belong to a flat area, if they belong then the signal is estimated to be digital.

**Slope detector**  The data maximums should be bigger than the other maximums caused by the noise. This blocks approaches the effect of the noise as a perfect triangle and checks if all the candidates to be local maximums are above this noise triangle. For high noise only.

**Peak rate**  The final step to detect the data maximums is to check which maximum candidates are the highest ones. For low noise only.

**Feedback**  the feedback to the max detector block is used when there are two candidates with similar amplitude, then the algorithm is run again but with more strict parameters in the max detector block.
Figure 4.9: Correlation of the spectrum of a PM on square subcarrier under different noise levels. The parameters of the simulations are: $f_{sc} = 500KHz$, factor = 4 and $\beta = 0.7rad$. The point marked with red circles are the local maximums that the algorithm is supposed to find.

Figure 4.10: Correlation of the spectrum of a PM on square subcarrier. Here we can see the effect of the modulation index $\beta$ to the correlation of the spectrum. The parameters of the simulations are: $f_{sc} = 500KHz$, factor = 4 and SNR = 8.
Figure 4.11: Correlation of a OQPSK spectrum, with different SNR values. The parameters are: symbol rate: 1Mbps, roll-off $\alpha = 0.1$.

Figure 4.12: Correlation of a carrier modulation spectrum (SPL) with different br values. The snr is $8dB$ and the modulation index $\beta = 0.7$.
Figure 4.13: Scheme of the search peak algorithm, Max detector builds a candidate list, and the other blocks discard the candidates, if at the end exist candidates then the modulation is Analog if not the modulation is digital.
4.2.3 Full Analog vs. Digital classifier

Once we have designed the 3rd block that belong to the analog vs. digital modulation classifier we can describe the structure of the full analog vs. digital classifier. In Figure 4.14 we can see the scheme of the classifier, first the signal is split in different overlapping segments (time segments) then for each segment the Welch spectrum estimation is computed. Afterward from each spectral estimation the smooth correlation function will be obtained, finally these smooth correlation functions will be averaged. The final step is the local maximum search algorithm that checks for the presence of local maximums in order to classify the signal as analog signal or digital signal.

Figure 4.14: Scheme of the analog vs. digital classifier
4.3 Experimental results: analog vs. digital classifier

4.3.1 Simulation parameters

In the following section the exact parameters of the different modules from the A.vs.D.MC will be explained, as well as details of the simulation and some extra features of the algorithm. The algorithm has features (length of the filters, FFT resolution, number of averages, etc.) that may be adjusted or modified in order to increase the performance for specific conditions. The current proposed features values has been decided experimentally.

4.3.2 Multiple estimation

A common method to increase algorithms performances is to run the algorithm more than one time and to decide for the most frequent option. For each data stream, we can run an independent MC algorithm for different time slots of the data. That is to split the data stream in different frames, and run the A vs. D MC independently for each frame. Unfortunately due to computational constrains the performance of this multiple estimation method will be analyzed only analytically. If we suppose a performance \( P \) (probability of correct classification), and suppose that the data stream is divided in to \( N \) frames. Choosing the classification that appears more times in the \( N \) classifications (more than \( N/2 \)): the total probability of classification \( P_N \) is

\[
P_N = \sum_{i=\frac{N}{2}+1}^{N} \binom{i}{N} P^i (1 - P)^{N-i}
\]

(4.20)

4.3.3 Simulation parameters

- Incoming signal
  - Modulation: uniformly chosen between GMSK, OQPSK, PN on sin sub carrier, PM on square subcarrier and PM on carrier.
  - Subcarrier frequency: uniformly distributed between 300 and 600 KHz.
  - Modulation index \( \beta = 1 \)
  - Bit rate factor: randomly selected between 4 and 5 (only for Analog modulations)
  - Roll-off \( \alpha = 0.1 \) (for OQPSK)
  - Bit randomly selected between 0.5 and 0.25 (only for GMSK)
  - Bit rate: 2048 kbps (for GMSK and OQPSK) 512 kbps (PM on carrier)
- SNR values: 2 3 3.5 4 4.5 5 6 7 8 9 10 15 30 dB
- Total number of samples \( 2^{17} \) samples
- Number of samples of each frame\(^1\): \( 2^{14} \) samples
- Number of frames: 16 (50 % overlap)
- Total number of samples \( 2^{17} \) samples
- The total time of acquisition is 0.032 seconds.

---

\(^1\)Here frame is defined as in equation [4.19] not as it has been explained in Section 4.3.2
• Spectrum Correlation Module
  – Welch spectrum estimate
    * FFT length $2^{12}$
    * Window type: Hamming
    * Number of averages: 7
  – Peak delete function: delete 5 samples average with 5 samples
  – Smooth filter: Hamming window 50 samples
• Peak search module
  – Max detector window 5 samples (first iteration) 50 samples (second iteration)
• Theoretical improve of multiple estimation (see Section 4.3.2): 5 frames.

4.3.4 Experimental results

The different modulation schemes are GMSK, OQPSK, PM on sin subcarrier, PM on square subcarrier and PM on carrier. The simulation were done for different SNR, for each SNR value, 400 random modulations where analyzed. The Results are shown in 5 different Figures. In Figure 4.15 the Probability of classify an Analog signal correctly is shown. The performance is nearly 100% for 4.5 dB for subcarrier modulations, nevertheless by the PM on carrier modulations the performances is close to the 95% and reaches the 100% at 7 dB. The best performance is achieved by the PM on sinusoidal subcarrier (100% at 4 dB), the second best performance is by the PM on square subcarrier (96% at 4.5 dB and 100% at 6 dB). Both subcarrier modulations have better performance than the PM on carrier modulation that achieves the 100% of correct estimations at 7 dB.

The performance for the subcarrier modulations varies about $\pm 1 \text{dB}$ depending on the subcarrier shape. The spectrum of the square subcarriers is wider than the sinusoidal subcarriers. As a consequence, the secondary lobes of the spectrum correlation are masked. As a result the detection becomes harder. In addition the performance of the PM on carrier, is affected by the fact that it is a modulation without subcarrier. As a consequence the data lobes are closer to the data and each other. For this reason, in the presence of noise the two data lobes can be mask and confused with only one data lobe (digital modulation).

For the Digital signals the classification performance is shown in Figure 4.16, is it important to notice that it is nearly perfect: in most of the cases the Digital modulations are classified as Digital modulations. They are correctly classified because the classification algorithm assumes that the signal is Digital unless secondary peaks in the correlation are found. In situations of low SNR the spectrum becomes flat (only noise contribution). As a consequence the secondary peaks are less powerful, for this reason the correct classification of digital signals is almost 100% for very low SNR.

In Figure 4.17 we use the method of Section 4.3.2 to compute the theoretical performance (see Equation 4.20) of the MC with the Analog signals. The performance increase is about 0.5 dB for the subcarrier modulations and about 3 dB for the carrier modulation. In Figure 4.18 there is the performance of the MC for the Digital modulations using the
multiple estimation method, notice that the performance here is perfectly flat, the Digital modulations are always correctly classified. In addition in Figure 4.19 the average performance of Analog vs. Digital MC is shown. The performance is about 93% at 4 dB, in addition the multiple estimate method increases the performance about 0.5 dB and the performance at 4dB is 98%.

**Figure 4.15:** Probability of correct classification, for the Analog signals PM on sin subcarrier and PM on square subcarrier. The parameter of the signals are described in Section 4.3.3.
Figure 4.16: Probability of correct classification, for the Digital signals GMSK and OQPSK. The parameter of the signals are described in Section 4.3.3. Notice that there is a decrease of performance around 10 dB, its caused because besides of at low SNR the spectrum correlation shape is almost triangular (only noise contribution) when the SNR increase the secondary lobes of the GMSK and OQPSK modulations can create confusing peaks in the triangular shape of the correlation spectra. For higher SNR the correlation spectrum has a pretty small noise contribution, as a consequence it has no triangular shape and the possible peaks are very low.
Figure 4.17: Theoretical performance improve of the multiple estimation (see Section 4.3.2). Probability of correct classification for the Analog modulations.

Figure 4.18: Theoretical performance improve of the multiple estimation (see Section 4.3.2). Probability of correct classification for the Digital modulations.
4.3.5 Performance dependence of different parameters

4.3.5.1 Modulation index $\beta$

The modulation index is the parameter that controls the relation between the amount of power allocated in the carrier and the amount of power allocated at the data. Since the Analog vs. Digital classifier designed detects the data lobes presence in the correlation spectrum. As a consequence the modulation index will have an important influence in the performance.

In Figures 4.20, 4.21 and 4.22 the performance of the algorithm for different modulation index is shown. In Figure 4.20 the performance for a PM on sinusoidal subcarrier is shown. Notice that since the modulation index increases (the data power increases\(^1\)) the performance is improved. In Figure 4.21 the performance for a PM on square subcarrier is plot, notice that the performance does not increase with the modulation index, although the data power $P_d$ increases (see Table 4.1). This phenomena is caused by the intermodulation product power $P_l$. $P_l$ generates interferences in the correlation of the spectrum and makes the local data maximums harder to detect. The effect of the modulation index $\beta$ in PM on carrier modulations is similar to the effect in the PM on sinusoidal sub carrier modulations.

More information about the power distribution can be found in Appendix A. Finally is it important to notice the poor performance of the algorithm when the modulation index is 0.1 because of the power allocated in the data is less than the 1%.

---

\(^1\)In Table 4.1 there is the power distribution with the different modulation index values.
Figure 4.20: Performance of the MC for a PM on sinusoidal subcarrier modulation with different modulation indexes. Parameters $f_{sc} = 512 \text{kbps}$, $br = 100 \text{kbps}$.

Figure 4.21: Performance of the MC for a PM on square subcarrier modulation with different modulation indexes. Parameters $f_{sc} = 512 \text{kbps}$, $br = 100 \text{kbps}$.
Table 4.1: Distribution of the power in the PM on subcarrier modulations. $\beta$ is the modulation index in radians. $P_d$ is the ratio of data power, $P_c$ is the ratio of carrier power and $P_I$ is the ratio of intermodulation power (only for PM on square subcarrier). For more information about the power distribution see Appendix A.

<table>
<thead>
<tr>
<th>Modulation index $\beta$</th>
<th>PM sin</th>
<th>PM sq and SPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta = 0.1$</td>
<td>0.99</td>
<td>0.01</td>
</tr>
<tr>
<td>$\beta = 0.5$</td>
<td>0.7702</td>
<td>0.2298</td>
</tr>
<tr>
<td>$\beta = 1$</td>
<td>0.2919</td>
<td>0.7081</td>
</tr>
<tr>
<td>$\beta = 1.5$</td>
<td>0.005</td>
<td>0.995</td>
</tr>
<tr>
<td>$\beta = 2$</td>
<td>NC</td>
<td>NC</td>
</tr>
</tbody>
</table>

Figure 4.22: Performance of the MC for a PM carrier modulation with different modulation indexes. $br = 512kbps$.

4.3.5.2 Frequency offset

In order to study the effect of the frequency offset in the MC performance we have made some simulations with different frequency offset values. In Figure 4.23 it can be seen how the frequency offset has no effect in the performance. In Figure 4.24 it can be seen how the frequency offset has more effect into the SPL modulations than in the PM on subcarrier modulations, however the performance is varies between $\pm 0.5dB$. 


Figure 4.23: Frequency offset effect in the MC for PM on subcarrier modulations. Parameters $\beta = 1.5$, $f_{sc} = 512KHz$, symbol-rate $br = 100KHz$.

4.3.5.3 Subcarrier frequency $f_{sc}$ and symbol-rate $br$

In the previous simulations (see Section 4.3.4) we have check the performance of MC for PM on subcarrier modulations with a subcarrier frequency uniformly distributed between 300 and 600 KHz. In order to enter in detail about the effect of the subcarrier frequency in the MC performance, we have done simulations for specific values of the subcarrier
frequency $f_{sc}$. In Figure 4.25 there is the performance of the MC for several PM on subcarrier modulations with different $f_{sc}$ values. There it is shown how the performance is almost constant for $f_{sc}$ values between 700 KHz and 400 KHz, however the performance is slightly decreased for $f_{sc}$ values below 300 KHz. However when the range of $f_{sc}$ values is increased, the performance decrease between a high $f_{sc}$ value and a lower one may be more important. For instances the performance decrease between $f_{sc} = 1024 KHz$ and $f_{sc} = 102 KHz$ is about 2 dB (see Figure 4.25). The reason for this behavior is that the smallest the $f_{sc}$ is, the more closer the local maximums in the correlation spectrum are. As a consequence the local maximums are harder to detect, because the local maximums are close to the global maximum.

![Figure 4.25: Subcarrier frequency effect in the MC for PM on subcarrier modulations. Parameters $\beta = 1$](image)

The effect of the symbol-rate while classifying PM on carrier modulations is not clear; since it seems that there is no obvious trend of dependence between the symbol-rate and the probability of correct classification. There are two effects to take into account. In one hand if the $br$ is increased the distance between the two data lobes grows up. As a consequence the performance should be increased. However, in the other hand, the bandwidth of the data lobes is also increased with the effect of mixing both data lobes making the detection harder. In Figure 4.26 there is the performance under different symbol-rates. It seems that the second effect is more important than the first one because the performance slightly decreases when the symbol-rate increases.
4.3.6 Gaia and Bepicolombo missions performance

In order to check the performance of our Analog vs. Digital MC with real parameters, the Bepicolombo and Gaia missions will be used. The special characteristics of this missions are:

- **Bepicolombo**
  - Modulation scheme: PM on sinusoidal sub carrier
  - Modulation index $\beta = 1.25$
  - Bit rate: $60kbps$
  - Subcarrier frequency: $262.144KHz$
  - Doppler: $\mp400KHz$

- **Gaia**
  - Modulation scheme: PM on sinusoidal sub carrier
    * Modulation index $\beta = 1.25$
    * Bit rate: $36.697kbps$
    * Subcarrier frequency: $146.789KHz$
    * Doppler: $\mp400KHz$
  - Modulation scheme: PM on carrier
    * Modulation index $\beta = 1.5$
    * Bit rate: $512kbps$ and $285kbps$
    * Doppler: $\mp400KHz$
The performance for the PM on subcarrier frequencies is the expected. The PM on sinusoidal subcarrier with $f_{sc} = 262KHz$ has a similar performance as the shown on Figure 4.20 (black line: PM on sinusoidal subcarrier, $\beta = 1.5$ and $f_{sc} = 512KHz$) besides of the effect of the Doppler and the expected decrease of performance because of the lower $f_{sc}$ used its performance is almost the same. The other PM on sinusoidal subcarrier modulation ($f_{sc} = 146KHz$ green line), has worse performance because of the really low subcarrier frequency value, however at 5 dB the ratio of correct classification is 100%. The PM on carrier modulation with higher symbol-rate ($br = 571kbps$ blue line) has an acceptable performance since it is not affected by the Doppler (see Figure 4.26), however the when the symbol-rate decreases to 285kbps the Doppler interferences became important and the performance is decreased.

![Figure 4.27: Performance of the Analog vs. Digital classifier for the Analog modulations of the Bepicolombo and Gaia Missions](image)

### 4.3.7 Conclusions

We have developed one algorithm which is capable to distinguish between the modulations with data around the carrier (analog) and modulations with a suppressed carrier (digital). The algorithm has been designed to be robust against the Doppler and to work under low SNR conditions. In addition the Analog vs. Digital Classifier has been considered as the first step of an Hybrid modulation classifier designed to classify the different Telemetry signals in 2 steps. The performance of the A. vs. D. classifier depends on different factors

- Incoming modulation
- Modulation parameter
  - Bit rate
Autonomous Receivers Report
December 3, 2008. Page 69 of 152

- Subcarrier frequency
- Modulation index

• Signal to Noise Ratio (SNR)

The performance depends on the modulation parameteres and also on the modulation scheme. Specifically, the digital modulations are a 99.9% of the times correctly classified. In side the analog modulations group, the best performance is for the subcarrier modulations with high subcarrier frequencies and high modulation index ($\beta$ close to 1.5) the 95% of correct classification is achieved for 4.5 dB. On the other hand, PM on carrier modulation with low modulation index have lower performance, for instance for a $\beta = 0.5$ the 99% is achieved at 8 dB. In addition, the correct classification of the analog modulation depends on different parameters, in the following we have summarized its dependences.

• PM on sinusoidal subcarrier:
  - Subcarrier frequency $f_{sc}$: The performance is increased when $f_{sc}$ is high, however this effect is only important for $f_{sc}$ values below 200 KHz.
  - Modulation index $\beta$: As much power is allocated to the data better is the performance. An increase of 0.5 rad in $\beta$ increases the performance 0.5 dB (see Figure 4.20).
  - Bitrate $br$: the bitrate values are small compared to the subcarrier frequency values, as a consequence its influence is not important.
  - Frequency offset: there is no effect of the frequency shift in the performance.

• PM on square subcarrier:
  - Subcarrier frequency $f_{sc}$: the same as the PM on sinusoidal subcarrier
  - Modulation index $\beta$: The presence of intermodulation products makes the relation between performance and $\beta$ foggy.
  - Bitrate $br$: the same as the PM on sinusoidal subcarrier
  - Frequency offset: there is no effect of the frequency shift in the performance.

• PM on carrier:
  - Modulation index $\beta$: the same as the PM on sinusoidal subcarrier.
  - Bitrate $br$: The performance for high symbol-rates is 1 dB below the performance for low symbol-rates, however this is is only appreciable between very different symbol-rates.
  - Frequency offset: its presence decreased affect the performance. However its effect is less than 1 dB.

In the case of modulations with subcarrier, the A. vs. D. classifier provides also an estimation of its subcarrier frequency $f_{sc}$. Since the position of the data lobes in the correlation spectrum coincides with the frequency of the subcarrier. As a consequence the location of the local maximum in the correlation spectrum provides an estimation of the
subcarrier frequency\(^1\).

The proposed algorithm, has shown an acceptable flexibility in terms of modulation parameters, it has an acceptable performances in a range of subcarrier between 200\(\text{KHz}\) and 1000\(\text{KHz}\) it also can work under different modulation indexes. In addition it is also flexible with the symbol-rate of the incoming signal. Additionally the robustness against changes in the parameters of the digital signals is very high. As a consequence the A. vs. D. classifier can work with a broad range of different incoming signals. In comparison with other PR classifier it has a high performance since it achieves a 95\% at 5 dB while the other PR methods need about 10-15 dB to achieve an acceptable performance. Nevertheless the performace of the ML methods proposed in [17] is much better than ours. On the contrary A. vs. D. MC does not need any aid from other estimation modules, since it is perfectly blindy and does not need any estimation of signal parameters. On the contrary the ML methods need estimation of additional parameters of the incoming signal.

As a conclusion we have designed and algorithm which performance is better than most of the PR methods, but lower than the ML methods. The advantage of this method is that is very flexible against the signal parameters, that does not need for additional estimation of signal parameters and it is robust against Doppler shift. In addition it provides an estimation of the subcarrier frequency \(f_{sc}\) and of the symbol-rate \(br\) of the PM on carrier modulations that may be used by other modules of the AR.

### 4.4 Further work and new proposals

There are different points of view to increase the performance of the Analog vs. digital classifier. In addition there are some methods that has been proposed in [17] but there are no published test. For instances, the estimation of the modulation index can be useful in this case because the modulation index of the Digital modulations is always \(\pi/2\) on the contrary for the Analog modulations the modulation index depends on the configuration but it is never \(\pi/2\). In [17] they present some ML methods to estimate the modulation index however this methods needs many information of the signal to be used in an AR. In addition in [17] is presented an interesting add-hoc method to estimate the modulation index, which could be tested in order to compare its performance. Unfortunately this method can not be used in the first iterations of an AR because it needs the estimation of additional parameters, in addition when the modulation index is close to \(\pi/2\) its performance is expected to decreases because the modulation index of the digital signals and analog signals is very similar\(^2\). Alternatively our A. vs. D. modulation classifier should have no problems in this kind of situations because it does not look at the data to carrier ratio.

Alternatively we have not used the fact that besides of the presence of two data lobes in the analog modulation, there is also the presence of a carrier. Its presence can be

---

\(^1\)The accuracy of this estimate has been briefly studied and achieves an estimation error of 10\% at 5 dB of SNR. Additionally the position of the local maximums corresponds also with the symbol-rate of the PM on carrier modulations.

\(^2\)We have tested briefly some of algorithms for the modulation index estimation purposed in [17]. The results have shown that the performance decreases when the modulation index is close to \(\pi/2\), additionally the performance is better for the PM on carrier than for the PM on subcarrier modulations. Besides of this facts we consider that this algorithm can also be used to classify the signal between analog and digital modulations.
exploited in our Analog vs. Digital classifier, however its hard to detect a single spectral line in the spectra in low noise situations. The carrier is supposed to be always in the 0Hz (in baseband) but due to the Doppler its positions may not be there which increases the difficulty of finding the carrier. Nevertheless the presence of a carrier peak can be used to distinguish Analog modulated signals from the Digital modulated signals.

**Acquire time:** If the acquire time is not a constrain, the performance can be improved by increasing the number of samples of the algorithm and, as a consequence, the length of the FFT transforms and the number of Welch averages, achieving more frequency resolution and less variance in the spectra estimations.

**Analog vs. Analog:** Once the coarse modulation estimation has been done and the signal has been classified as Analog, the following step is to distinguish between the carrier modulations and the subcarrier modulations. Since the Analog vs. Digital classifier, makes an estimation of the subcarrier frequency, we can check for the presence of data around the subcarrier estimated frequency, if there is data there then the modulation is a PM on subcarrier and if not the modulation is a PM on carrier.
Part III

Symbol-rate Estimation
Overview of the different Symbol-rate estimation methods

In this Chapter we will introduce and classify different methods for the symbol-rate estimation. The probability of correct classification $P_{cc}$ or the $MSE$ of the estimation are going to be used in order to compare the different algorithms. In most of the cases we assume our input signal to be linearly modulated. One important assumption, which is the basis of some methods and very useful for others, is the fact that the different symbol-rate values are related\footnote{Specifically most of the symbol-rates are a power of two} one to each other. Here we will compare this methods, and discuss the different criterion to decide for the best method to estimate the symbol-rate in a Satellite scenario.

5.1 Introduction

Symbol-rate estimation of a digital communication signal is an important task when performing passive signal analysis and automatic modulation classification. In this Chapter, we present a comparison of the different approaches to the problem of estimating the symbol rate of a linearly modulated signal.

Blind rate estimation has been considered in the literature from different points of view. Some methods exploits the fact that the received signal is cyclostationary\footnote{A stationary process is a stocastic process which statistical characteristics (such as mean, variance...) don’t change in the time, a cyclostationary process is a process which statistical characteristics are periodic in the time domain} with a period $T$, where $1/T$ is the symbol-rate. As an extension for those methods the filter bank structure was proposed in \cite{54} to increase the performance at low roll-off (excess of bandwidth) factors. A Maximum likelihood (ML) was introduced in \cite{10}, this method was designed to achieve a good performance for low SNRs and/or low roll-off factors, being at the same time robust to frequency offsets. A complete new method based on the SSME (Split Symbol Moment Estimator) was presented in \cite{17} together with different methods to reduce the effects of frequency offset, an extension to the SSME method in conditions of low bit rate was presented in \cite{22}. Alternatively a method based on the inverse Fourier
transform was presented in [34] with very interesting results.

5.2 Maximum Likelihood symbol-rate estimate

In [10] a Maximum Likelihood method is introduced, this method was designed to achieve a good performance for low SNR and/or low roll-off factors, being at the same time robust to frequency offsets $\gamma$. They focus on linearly modulated signals going through a frequency-flat channel. The ML methods exploits the dependence of both the shaping pulse and its repetition rate with symbol period $T$. They focus on the general case where all the synchronization parameters\(^1\) $\{T, \theta, F, \epsilon\}$ as well as the symbol sequence $a_k$ are unknown for the receiver. The ML tries to estimate the oversampling ratio $N_s$ defined as [10]:

$$N_s = \frac{T}{T_s}$$  \hspace{1cm} (5.1)

where $T_s$ is the sampling period and $T$ is the data period. Then if we can estimate $N_s$ the number of symbols received is [10]:

$$K \approx \frac{N}{N_s}$$  \hspace{1cm} (5.2)

where $N$ is the number of received samples. So the problem of estimate the symbol-rate becomes a problem of estimate the oversampling rate.

The ML function is [10]

$$\wedge (r|\psi, a, \sigma^2, h) = \frac{1}{(\pi\sigma^2)^N} \exp \left(\frac{-1}{\sigma^2} \|r - hG(\psi)a\|^2\right)$$  \hspace{1cm} (5.3)

where $\psi = [T, \theta, \gamma, \epsilon]^T$, $a$ is the vector containing the unknown symbols, $h$ is the channel transfer function, $r$ is the received signal and $G$ is a matrix that depends on the unknown parameters ($\{T, \theta, f_c, \epsilon\}$) and the raised-cosine shape\(^2\).

The ML function is averaged over the data $a_k$, assumptions of low SNR and hight SNR are done. Parameters such as the frequency offset $\gamma$ and the timing offset $\epsilon$ are estimated, with specific algorithms, before the ML estimates the symbol rate.

They proposed a two step algorithm: first a coarse estimate of the symbol-rate followed by a fine estimate. This two step algorithm is useful in our case because the number of possible symbol-rate is small, then once the coarse estimate is done the fine estimate just have to compare the two closer candidates to the coarse estimate and decide for the most likely. With this two step method we handle the problem of searching for the global maximum of a specified cost function.

In the Figure 5.1 is shown a comparison of the coarse ML function and the fine ML function. In Figure 5.2 It is shown a table of the performance of the coarse method. In Figures 5.3 and 5.4 the performance is plot in terms of $P_{cc}$ and $MSE$. Notice that there is a strong dependence between the performance and the amount of data (as much data is needed more delay has the algorithm) available to compute the algorithm.

\(^1\) $f_c$ denotes the carrier frequency, $\theta$ denotes the phase offsets and $\epsilon$ is the timing offset

\(^2\) Notice that here we assume that the pulses are raised cosines
**Figure 5.1:** Coarse and fine symbol-rate estimate. The ML functions for the estimate of the $N_s$ are plotted for the coarse and fine estimate. Extracted from [10].

**Table I**

*Mean estimated value of the coarse estimate after $10^4$ realizations. $N_s = 5$, $\alpha = 0.25$*

<table>
<thead>
<tr>
<th>SNR [dB]</th>
<th>-5</th>
<th>-4</th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N = 1250$</td>
<td>5.34</td>
<td>5.50</td>
<td>5.56</td>
<td>5.55</td>
<td>5.40</td>
<td>5.31</td>
<td>5.20</td>
<td>5.14</td>
<td>5.10</td>
<td>5.06</td>
<td>5.03</td>
</tr>
<tr>
<td>$N = 2500$</td>
<td>5.49</td>
<td>5.52</td>
<td>5.43</td>
<td>5.33</td>
<td>5.24</td>
<td>5.17</td>
<td>5.12</td>
<td>5.09</td>
<td>5.05</td>
<td>5.03</td>
<td>5.01</td>
</tr>
<tr>
<td>$N = 5000$</td>
<td>5.48</td>
<td>5.38</td>
<td>5.29</td>
<td>5.20</td>
<td>5.16</td>
<td>5.12</td>
<td>5.09</td>
<td>5.06</td>
<td>5.04</td>
<td>5.02</td>
<td>4.997</td>
</tr>
</tbody>
</table>

**Figure 5.2:** Table of the final estimate for the $N_s$, under different SNR values and different number of samples $N$. The true value for $N_s$ is 5 and the roll-off factor is $\alpha = 0.5$. Extracted from [10].
Figure 5.3: Coarse estimate: probability of acquiring the correct symbol rate within an error margin of 10% is plotted against the SNR and for different sample sizes ($N$). The true oversampling ratio is $N_s = 5$, and the roll-off factor $\alpha = 0.5$ is assumed to be known. Extracted from [10].
5.3 Cyclostationarity based symbol-rate estimation technique

This kind of symbol rate estimation is based on the basic observation that the oversampling of the received signal generates cyclostationarity statistics, and that the symbol-rate coincides with its cyclic frequency \( f_d = 1/T_d \), where \( f_d \) is the cyclic frequency (see Figure 5.5). In [20] it is shown that the cyclic frequencies of the Fourier transform of the autocorrelation function are a family of discrete values determined only by the symbol-rate \( f_d = 1/T_d \), where \( f_d \) is the cyclic frequency (see Figure 5.5). This means that the symbol-rate is one of the cyclic frequencies \([33, 38]\). The frequency position of the lower spectral line will correspond to the symbol-rate.

A discrete-time zero-mean (almost) cyclostationary signal, \( x(t) \), is characterized by the property that its time-varying autocorrelation function \( R_{xx}(t; \tau) = E\{x(t)x^*(t = \tau)\} \) admits Fourier series representation as

\[
R_{xx}(\alpha; \tau) = \sum_{\alpha} R_{xx}(\alpha; \tau)e^{j\cdot2\pi\alpha t} \quad (5.4)
\]

where the Fourier coefficient \( R_{xx}(\alpha; \tau) \) is the cyclic autocorrelation function and it is given by

\[
R_{xx}(\alpha; \tau) = \lim_{T \to \infty} \left( \frac{1}{T} \right) \sum_{t=0}^{T-1} R_{xx}(\alpha; \tau)e^{-j\cdot2\pi\alpha t} \quad (5.5)
\]

Figure 5.4: Fine search performance of the Maximum Likelihood estimator and its simplified version. There is no frequency offset. \( N = 3000 \). Extracted from [10].
where $\alpha$ is called the cycle frequency. The Fourier coefficients $R_{xx}(\alpha; \tau)$ of $R_{xx}(t; \tau)$ is a family of discrete values determined only by the symbol-rate $f_d = 1/T_d$. Now knowing that the symbol-rate is one of the cyclic frequencies \cite{33,38}, we have to detect the spectral line placed at the lowest frequency. The frequency position of this spectral line will correspond to the symbol-rate. See example in Figure 5.5, more information about second-order blind estimators can be found in \cite{48}.

Simulations were done by \cite{20}, a QPSK signal is examined with different sampling rates and the symbol-rate is estimated under different SNR scenarios. Figure 5.6 shows the results in terms of NRMSE (Normalized Root Mean Squared Errors). Notice that there is a strong dependence of the performance from the number of samples used to estimate (see Figure 5.6).

In conclusion: the cyclic-correlation based approaches such as \cite{20,38,33} are able to handle a received signal that has been pulse-shaped in the transmitter. These approaches rely on the observation that the received signal is cyclostationary and its cyclic frequencies are integer multiples of the symbol-rate \cite{20,38,33}. In the case that the excess bandwidth factor $\beta$ satisfies $0 < \beta < 1$, the amplitude of the Fourier transform of the cyclic-correlations will have the first peak at the frequency of symbol-rate. That is, the symbol-rate estimation can be done by searching the symbol-rate line in the spectrum.

![Figure 5.5: Example of the cyclic frequencies of a cyclostationary signal. The symbol-rate corresponds to the first non zero frequency. In a real case there will be much noise influence (because the noise is stationary) and as a consequence it can be very difficult to detect the first spectral line. Extracted from \cite{20}.](image-url)
Figure 5.6: Performance of the symbol-rate estimate of [20]. $T$ is defined as the sampling time, that is $T = 128T_d$ means that 128 symbols are sampled and processed to estimate the symbol rate. Extracted from [20].

5.3.1 Filter bank

The approaches of [20, 38, 33] will fail when the excess of band width is 0. Since the symbol-rate line vanishes [51].

Further, even though the excess bandwidth is not zero, the magnitude of the symbol-rate line is not guaranteed to be the global maximum of the spectrum if the signal-to-noise ratio (SNR) is low. As a consequence, it is not trivial to recognize the symbol-rate line even though it does exist.

In [54], is proposed a symbol-rate estimation algorithm that is able to work without knowledge of modulation type, pulse shaping filter, etc. The method uses a filter bank to filter the signal before enter in to the fourth-order nonlinearity (fourth-order cyclic-moment). Filtering the signal will make easy the search of the $1/T_d$ peak. The bandwidth of the filters is very important in the performance of this method, and its optimal value is symbol-rate dependent. For this reason a first coarse estimate is done, then the optimal bandwidth of the filter bank elements are decided. Finally a a fine estimation is done by using a $4^{th}$ order nonlinearity. Each element of the filter bank has a different bandwidth, the bandwidth of the $k^{th}$ element is:

$$B_{LPF,k} = f_a + \frac{f_b - f_a}{K(k-1)} \times 0.375$$

(5.6)

where $f_a$ and $f_b$ are the $3dB$ frequencies. Such arrangement ensures that several LPFs bandwidths are in the workable rang and some of them will be in the optimum bandwidth.
Each element of the filter bank produces one candidate to be the symbol-rate, then a
decision algorithm is run in order to decide for the true symbol-rate estimation.
Simulation results were done in [54] with a BPSK signal, the CFO (center frequency offset)
of the 10% of the symbol-rate, a filter bank of 10 elements and a roll-off factor of 0.2.
In Figure 5.7 the filter bank method is compared with the method presented in [38] based
on the cyclostationarity. Like in [20 38 33] methods the performance of the filter bank
method increases with the excess of bandwidth.

![Figure 5.7: Performance of the symbol-rate estimate of [38] vs the filter bank method [54] notice that the roll-off factor is 0.2. Number of symbols 4096 sampling rate is 4 times the symbol rate $f_s = 4f_d$. Extracted from [54]](image)

5.4 Inverse Fourier transform

This method consists, introduced in [34] also in a two step algorithm, the first step consist in
a coarse estimation of the symbol-rate using the Inverse Fourier transform, and the second
consist in a symbol synchronizer for fine-tuning. The coarse estimation of the symbol-rate
uses the inverse fourier transform (IFFT) of the estimated spectrum of the incoming signal.
The IFFFT has a minimum next to 0 which corresponds to the oversampling rate $N_s$.
Figure 5.8 shows the performance of many different methods, such as filter bank, cyclic
correlation and Inverse Fourier Transform methods. The modulation scheme is a QPSK,
with $N_s = 4$, $\alpha = 0.2$, the time shift $\tau$ and the carrier frequency $\theta$ are randomly selected.
Notice that the simulations are done just with the coarse estimate not with the fine estimate.
Figure 5.8: Performance of the symbol-rate estimate of [34] vs the filter bank method [54] and notice that the roll-off factor is 0.2. the number of symbols is 4096, τ and θ randomly selected and the frequency offset is assumed to be small, the oversampling ratio is \( N_s = 4 \). Extracted from [34].

The fine stage consist in a estimate (timing recovery) of the timing offset \( \tau_L \), then the new estimate of the oversampling is \( \hat{N}' = \hat{N}(1 + \frac{\tau_L}{L}\hat{N}) \) (5.7)

5.5 Symbol-rate and SNR joint estimation based on the Split Symbol Moment Estimation (SSME)

This method, introduced in [17], works under the consideration that the data rate come from a set of known values, in particularly the data rates are assumed to be related by the integral power of an integer base \( B \). The estimation of the symbol-rate can be done jointly with the SNR. Also an extension of the algorithm is proposed in order to reduce the effect of the symbol-timing error, producing a jointly symbol-timing, SNR and symbol-rate estimation. The possible data rates \( R \) comes from a known finite set of values of the form

\[
R = B^l \cdot R_b, \quad 0 \leq l \leq l_{\text{max}}
\] (5.8)

The algorithm works in the following way: it uses the SSME estimation in order to estimate the SNR. One important characteristic of that estimation is that the mean of
the SNR estimation increases monotonically by a factor of $B$ each time the data rate is lowered until the true data rate, and hence the true SNR, is reached. Using this property the algorithm works in the following way:

- Run the SSME assuming the highest symbol-rate (lets call this symbol rate $R_{l_{\text{max}}}$).

- Decrease the symbol-rate (by a factor 2) and run SSME (lets call this symbol rate $R_{l_{\text{max}}-i}$ with $i$ being the number of realizations of this step).

- If the mean of the SNR estimate has increased go to step 2.

- If the mean of the SNR estimate hat decreased the true symbol-rate value is the previous one $R_{l_{\text{max}}-i+1}$ (because of the property mentioned above).

This scheme for the SSME algorithm is presented in Figure 5.9. This method can be modified to reduce the effect of the symbol-timing error and also can be optimized to low symbol-rates or to high symbol-rates [22]. The performance of this algorithm is shown in Figure 5.10.

![Figure 5.9: Split-symbol moments estimator (SSME) for a rectangular NRZ pulse shape.](image)

**Figure 5.9:** Scheme of the SSME algorithm for the SNR estimate. Extracted from [17].
Figure 5.10: Performance of two different SSME methods. The first one is optimized for high data-rate values, and the second one is optimized for low symbol-rate values, both work with null symbol-timing error and zero frequency offset. Extracted from [17].

5.6 Other methods

In this section we will briefly comment some methods that have been discarded because of the low performance or because the difficulty in implementing the algorithm.

Most of this algorithms are designed to detect abrupt changes in the signal, this abrupt changes are supposed to be related with the bit transitions, a change of phase in the PSK cases, or a change of amplitude in the ASK case. In [13] proposed an algorithm capable to detect the symbol-rate without knowledge of the modulation type, it uses the Wigner
distribution to detect abrupt changes in the signal, and construct a frequency function of the abrupt changes frequency. After that some metrics are used to decide where there is an abrupt change or not, measuring the distance between abrupt changes one can estimate the symbol-rate, similarly in [53] and [52] the wavelet transform is used to detect the abrupt changes. In [50] an more accurate algorithm to decide for the more likely symbol-rate is proposed, but no abrupt changes detector is mentioned. In [31] supposes that the data is coded with convolutional codes and uses this fact to detect the symbol-rate.

5.7 Conclusion

One of the most challenging problems in the symbol-rate estimation is the big dispersion that exist between the possible incoming symbol-rates, such dispersion increase the difficulty when the algorithms try to estimate the symbol-rate. This problem is increased because some algorithms [17, 34] decrease its performance when the signal is oversampled. The oversampling occurs because the signal has to be always sampled at a sampling frequency bigger than the higher possible symbol-rate\(^1\), then when the incoming signal has a low symbol-rate the oversampling ratio is increased and the performance of the estimate decreased. The oversampling ratio is

\[
N_s = \frac{f_s}{f_d} \tag{5.9}
\]

where \(f_s\) is the sampling frequency (always bigger than the higher symbol-rate) and \(f_d\) the symbol-rate.

The simulations done in the previous methods mentioned just consider oversampling factors lower than 5 (that means a sampling frequency 2.5 times the symbol-rate), however in the Satellite scenario the oversampling factor can be higher. The lowest symbol-rate for a OQPSK telemetry signal is 8Kbps and the highest is about 2Mbps, that produces a maxim oversampling factor of 512 (see Appendix B). This large range of oversampling factors (from 2 to 512) is the most challenging problem in the symbol-rate estimation.

One of the desired characteristics of the algorithm is to be as much autonomous as possible, for this reason a possible architecture could be the use of a first coarse estimate can be done, and then in the following iterations the estimation of the signal parameters can be used to estimate the symbol rate in a more precise way (using the extensions\(^2\)). So the autonomy is an important factor to take into account. Since the AR uses many iterations to classify the signal, it could be a good idea to use a totally blind symbol-rate estimator for the first iterations and then change to a aided estimator for the final symbol-rate. The degree of robustness in the presence of jitter \(\epsilon\), frequency offset \(\gamma\), phase offset \(\theta\), excess of band width \(\alpha\) has not been studied in detail, and the different algorithms have not been compared under such kind of conditions.

In the other hand the performance of the algorithms in low SNR conditions has been largely checked and all the algorithms present a good performance under low SNR conditions.

\(^1\)frequency sampling \(f_s\) must be two times the bandwidth \(B_{m,ax}\), the \(B_{m,ax}\) is directly related with the symbol-rate

\(^2\)here extension refers to a modification of an algorithm in order to make the algorithm robust against some kind of interference, or to increase its performance adding estimating the interferences (jitter, frequency offset...).
Finally another factor to take in to account is the convergence of the algorithms, some algorithms need a certain minimum number of symbols to estimate the symbol-rate, that can be a problem when the symbol-rate is very low because it will introduce a large delay.

As a conclusion, the parameters to take into account in order to decide for the best symbol rate classifier are:

1. Degree of autonomy
   (a) which parameter of the signal does the method need (modulation type, phase offset, roll-off, timing...)?
   (b) can be implemented in a multiple stage algorithm (coarse + fine) ?

2. Robustness against jitter $\epsilon$, frequency offset $\gamma$, phase offset $\theta$, excess of band width $\alpha$
   (a) is it robust?
   (b) if it uses an estimation of $\epsilon$, $\gamma$, $\theta$ or $\alpha$ does it increases the robustness?(two stage)

3. Convergence of the algorithm (number of symbols(or samples) needed to estimate the symbol-rate) and acquisition time.

4. Degradation of performance because of the oversampling (can be solved with a two stage algorithm).
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Short description</strong></td>
<td>Exploit cyclostationarity and received pulse information</td>
<td>Cyclostationary frequencies are related with the symbol rate</td>
<td>Improved cyclostationary method against low roll-off</td>
<td>First minimum of the IFFT is related to the symbol-rate estimation</td>
</tr>
<tr>
<td><strong>Requirements</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modulation type</td>
<td>linear</td>
<td>linear</td>
<td>single carrier</td>
<td>linear</td>
</tr>
<tr>
<td>Pulse shape</td>
<td>SRRC $0 &lt; \alpha &lt; 1$</td>
<td>need excess of bandwidth $\beta \neq 0$</td>
<td>immune</td>
<td>SRRC</td>
</tr>
<tr>
<td>Autonomy</td>
<td>$T, \theta, F, \epsilon, a_k$ not needed</td>
<td>NA</td>
<td>modulation type not needed</td>
<td>NA</td>
</tr>
<tr>
<td><strong>Characteristics</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extensions</td>
<td>high/low SNR</td>
<td>no</td>
<td>no</td>
<td>high/low symbol rate, $\epsilon$ offset</td>
</tr>
<tr>
<td>Number of Stages</td>
<td>coarse + fine</td>
<td>1</td>
<td>coarse + filter bank + fine $\epsilon$</td>
<td>1</td>
</tr>
<tr>
<td>Joint estimate</td>
<td>$\epsilon, \gamma$</td>
<td>no</td>
<td>symbol-rate $f_d$</td>
<td>$\epsilon$</td>
</tr>
<tr>
<td>Estimated parameter</td>
<td>Over sampling $N_s$</td>
<td>Symbol-rate $f_d$</td>
<td>Over sampling $N_s$</td>
<td>Symbol rate $f_d$</td>
</tr>
<tr>
<td>Discrete Symbol-rate</td>
<td>not needed</td>
<td>not needed</td>
<td>not needed</td>
<td>yes</td>
</tr>
<tr>
<td><strong>Problems</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Roll-off</td>
<td>$\alpha \downarrow$ performance $\downarrow$ (Figure 5.4)</td>
<td>$\alpha \downarrow$ performance $\downarrow\downarrow\downarrow$ nearly immune</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Symbol timing $\epsilon$</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>simulations done</td>
</tr>
<tr>
<td>Frequency offset $\gamma$</td>
<td>decrease the performance</td>
<td>small $\gamma$ is not important</td>
<td>NA</td>
<td>simulations done in the SNR estimate</td>
</tr>
<tr>
<td>Number symbols $N$</td>
<td>$N \downarrow$ performance $\downarrow$ (Figure 5.3)</td>
<td>$N \downarrow$ performance $\downarrow$ (Figure 5.6)</td>
<td>performance decrease with oversampling</td>
<td>performance of SNR estimate decrease with oversampling</td>
</tr>
<tr>
<td>Peak search</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td><strong>Simulations</strong></td>
<td>1. if $\gamma$ present coarse performance $\downarrow$</td>
<td>1. performance studied under two data length 128$T_d$ and 64$T_d$ being $T_d$ the symbol rate (Figure 5.6)</td>
<td>1. simulations done with symbol timing error $\epsilon$, phase offset $\theta$ and frequency offset $\gamma$.</td>
<td>1. simulations of the estimate with different SNR</td>
</tr>
<tr>
<td></td>
<td>2. SNR $&gt; 2$dB</td>
<td>1. Number of symbols 4096</td>
<td>2. Effect of $\epsilon$ simulated</td>
<td>2. effect of $\epsilon$ simulated</td>
</tr>
<tr>
<td></td>
<td>3. simulations with different roll-off</td>
<td>3. sampling rate is 4 times the symbol rate $f_s = 4f_d$</td>
<td>3. performance under different symbol-rates simulated</td>
<td>3. performance under different symbol-rates simulated</td>
</tr>
<tr>
<td></td>
<td>4. simulations with echo channel</td>
<td>4. different SNR estimates are simulated</td>
<td>4. different SNR estimates are simulated</td>
<td>4. different SNR estimates are simulated</td>
</tr>
<tr>
<td></td>
<td>5. simulations vs. cyclic correlation based method [38]</td>
<td>2. simulations of the variance of the estimate under different data lengths.</td>
<td>5. comparison vs different methods see Figure 5.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6. convergence $N^{-2}$</td>
<td>3. different SNR estimates are simulated</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 5.1:** Summary of the characteristics of the different algorithms
In this Chapter we introduce a symbol-rate estimator designed to work in a broad range of different symbol-rates and in low SNR situations. The design of the algorithm is explained and reasoned out in the report. The algorithm uses multiple stages (inspired in different known methods) and is capable of working under low SNR and also under the presence of Doppler effect. Our efforts will be centered in designing a Symbol-rate estimation method robust against the signal oversampling, which is one of the most challenging points of the symbol-rate estimation methods.

6.1 Description of the satellite scenario and state of the art

6.1.1 State of the art

In Section 5 a general overview of the different symbol-rate estimation methods has been presented. After doing research on the different symbol-rate estimation methods (like the Maximumlikelihood [10], the cyclic correlation (cyclostationarity) [20, 38, 33], the Filter Bank [54], the Inverse Fourier method [34] or the Split Second Moment Estimator [22]). As a conclusion of the literature research about symbol-rate estimation we have seen that the performance of the different algorithms is strongly related with the oversampling\(^1\) ratio (the higher the oversampling is the lower the performance is). In addition the frequency offset of the channel has also an important role in the general performance. For the symbol-rate estimate methods based on the second order statistics (such as cyclostationarity) the excess of bandwidth plays and important role in the performance. In addition sampling time (time of acquisition) has is directly related with the performance.

6.1.2 Satellite scenario: general requirements

ESA is supporting a broad range of different mission types. Mission in Mars such as Mars Express with very low signal to noise ratio, or satellites that are in different orbits

\(^1\)the oversampling \(N\) is defined as the amount of samples per symbol, and it is directly related with the symbol-rate and the sampling frequency. \(N = \frac{f_m}{br}\)
around the Earth (with better signal to noise but with smaller satellite pass lengths). The symbol-rate changes in each mission and there are also different symbol-rates for the different modes of operation for a specific mission (see Appendix B). E.g. if a satellite with very low SNR has to send information related to the state of the satellite it will use a low symbol-rate (because it is required to send a small amount of information), but in the other case if the satellite has to send pictures in a very small time it will use a high symbol-rate. The requirements of the different mission are briefly explained in Appendix B but for more detailed information please refer to [45]. Due to the special nature of the satellite scenarios, our requirements for a symbol-rate estimation system are different from the requirements of the other published symbol-rate systems, presented in the previous reports. Our requirements are:

- Signal to noise ratio (SNR): the algorithm should have an acceptable performance in scenarios with SNR above -1dB.
- Frequency offset ($\gamma$): We will require to have an acceptable performance in scenarios with Doppler effect up to $\pm 400 KHz$.
- Symbol-rate ($br$): We will assume that the symbol-rates are between 8kbps and 2048kbps and that all the symbol-rates are a power of two.
- Convergence time: because of the limited time of visibility of the satellites, the algorithm should be fast enough to acquire and to estimate the symbol-rate in a reasonable short time.
- Complexity: the complexity of the algorithm is not one of the challenging design parameters since it will be implemented in a FPGA.

### 6.1.3 Design of the algorithm: challenging design parameters

To the best of the authors’ knowledge, no results for high oversampling levels have been published so far in the open literature. In [17] a study about the effect of the oversampling in the performance of the proposed method is done, however the maximum oversampling achieved is 8. In other publications like [34] and [54] the maximum oversampling achieved is 5. In contrast, besides having a limited number of possible symbol-rates, the satellite scenarios have a big difference between the different symbol-rates. Considering that the highest possible symbol-rate is 2048kbps, we have to assume that the sampling frequency $f_m$ will be at least $4096 \text{ samples/s}$. Sampling a 8kbps signal at $4096 \text{ samples/s}$ produces an oversampling rate of 512 samples per symbol. Is it clear, then, that the oversampling ratios of the satellite scenario are about 100 times bigger than the oversampling ratios studied in the open literature. For this reason we will have to design an algorithm capable of working with high oversampling levels.

The frequency offset introduced by the Doppler effect is also very important and, to the best of the authors’ knowledge, it is higher than the frequency offsets studied in the symbol rate estimation open literature. Owning to this fact, we will take care of designing an algorithm robust against the frequency offset.
6.2 Broad range symbol rate estimate for satellite scenario

Owing to the literature research done in Section 5, we have discarded the methods [20] and [10]. The algorithm [20] has been discarded because of its low performance under situations of small excess of bandwidth (low roll-off factors) the algorithm presented in [10] is also discarded because it’s low general performance, since it has an acceptable performance above SNR of 2 dB but not for lower SNR such as -1dB.

Alternatively the algorithm presented in [54] is designed to have a good performance in situations of low roll-off: despite of being an algorithm based on the ciclostationarity it also presents a good performance under the presence of frequency offset, for this reason this algorithm will be taken into consideration. We have also decided to consider the algorithm presented in [34] since it is the one that has the best performance and it’s quite simple to implement. For the design we will use tips and ideas of the IFFT method [34] and from the Filter Bank method [54].

One of the aims of this research project is to check and to look for methods to build an autonomous receiver. As a consequence, implementing new methods and checking their performance is more interesting than just confirm its performance. For this reason we will not implement the SSME method from JPL [17] and [22], because, besides being one of the most promising algorithms, we want to check if there exist alternatives.

6.2.1 Design of the algorithm

One of the most challenging problems of designing the broad range symbol-rate estimate is the oversampling factor. If we design an algorithm with a structure like the one in Figure 6.1 being the Symbol rate estimation block one of the algorithm presented in [54], the algorithm will have to work under a broad range of symbol rates, and, as a consequence with high oversampling ratios which will lead to a decrease in the performance. Experimentally we have confirm that both algorithms [54] and [34] decrease its performance with oversampling factors higher than 16.

Considering a system with a feedback, such system should be capable to, iteratively, adapt its sampling frequency to an acceptable value in order to have a oversampling ration inside an acceptable range. In the Figure 6.2 is presented a sketch of this feedback method.
One of the advantages of our scenario is that the number of symbol rates is small. Consequently there also exist a limited number of optimal sampling frequencies. Owning to this fact we guess that the algorithm will be capable to achieve the convergence with just one iteration. In addition, control the sampling frequency of the whole receptor is hard task and also some times is important to have oversampling for the symbol synchronization algorithms and for the PLL to hook into the signal [9], for this reason we propose an equivalent solution: instead of decreasing the sampling frequency we will decimate the input signal and consequently we will reduce the oversampling (see Figure 6.3). Supposing that we have a signal $y(t)$ and we sample that signal at a sampling frequency $f_{s1}$ then the equivalent sampled signal will be

$$y_1(n) = y(t)|_{t=n \frac{1}{f_{s1}}} \quad (6.1)$$

Now if we decimate the signal $y_1(n)$ by a factor $\beta$ bigger than 1, we obtain:

$$y_{1d}(k) = y_1(k \cdot \beta) = y(t)|_{t=k \frac{1}{f_{s1} \cdot \beta}} \quad (6.2)$$
that is equivalent to sample the $y(t)$ signal by a sampling frequency equal to

$$f_{seq} = \frac{f_{s1}}{\beta} \quad (6.3)$$

So we have seen that decimate the signal by a $\beta$ factor is the same as sampling the signal by a sampling frequency $\beta$ times smaller.

![Diagram](image)

**Figure 6.3:** Symbol-rate estimate with decimate for oversampling correction

One of the characteristics of the IFFT [34] and Filter Bank [54] algorithm is that they consist of two stages, one coarse estimate followed by a fine estimate of the symbol-rate. For the design of our broad range symbol-rate estimate we will also use this characteristic, first a coarse symbol-rate estimate will be used to decide for a properly decimate factor. Then the decimated signal, with an acceptable oversampling factor because of the decimation module, will be analyzed for a fine symbol-rate estimate. In the Figure 6.4 there is sketch the scheme of the algorithm.

![Diagram](image)

**Figure 6.4:** Fine and coarse (2 stage) symbol rate estimate
6.2.1.1 Coarse estimation

The first stage of our system should be capable to work under different oversampling values (see Figure 6.12). In addition it should be robust against the frequency offset $\gamma$ caused by the Doppler effect, have good performance at low roll-off factors and a good performance under low SNR scenarios. For this coarse stage it is important to minimize the maximum error, however we don’t care about the estimator variance or bias, since the main objective of the coarse stage is to classify the input signal as a low oversampled signal, medium oversampled signal and high oversampled signal. For this coarse estimate we propose to use the same method used in [54] which consist of a simple estimation of the bandwidth.

Defining the input base band signal as $y(n)$ with a Fourier transform $Y(f)$, then the bandwidth is defined as:

$$\hat{B}_y = \arg(f : Y(f)/Y_{max} = 0.5)$$  \hfill (6.4)

The bandwidth is tightly related to the symbol-rate (see Appendix A). Unfortunately some times the SNR levels are not high enough to allow a proper detection of the bandwidth (i.e. if the noise level is high the signal will never decay to the half of its maximum), for this reason instead of using the fourier transform $Y(f)$, the Welch spectral estimator will be used (see Figure 6.5 for a comparison between both spectrum estimates). Averaging many Periodograms will reduce the effect of the noise, and the overlapping characteristic of the Welch spectrum estimator will reduce the variance of the spectrum estimate [55]. Defining the Periodogram $I_{yy}^p$ of the signal $y(n)$ as

$$I_{yy}^p(e^{jw}) = \frac{1}{P} |Y_P(f)|^2 = \frac{1}{P} \left| \sum_{n=0}^{P-1} y(n)e^{-jwn} \right|^2$$  \hfill (6.5)

where $P$ is the data length or the length of the Fourier Transform. The Welch averaging Periodogram consist of split the data in different overlapping frames, compute the Periodogram of each windowed frame and then average the different Periodograms. Each one of the data frames can be expressed, for $l = 0, \ldots, L - 1$, as

$$y_l(n) = y(n + l \cdot D)$$  \hfill (6.6)

where $l \cdot D$ is the starting point of the $l$th frame (parameter $D$ depends on the overlap of the different segments and on the segment length). According to Equation 6.6, the Periodograms are defined similarly to Equation 6.5 except for a multiplication by the window $w_M(n)$,

$$I_{W_y,W_y}^M(e^{jw}, l) = \frac{1}{M} \left| \sum_{n=0}^{M-1} y_l(n) \cdot w_M(n) \cdot e^{-jwn} \right|^2$$  \hfill (6.7)

the Welch estimator is defined as the average of the $L$ windowed Periodograms (see Equation 6.7)

$$\hat{C}_{y,y}^W(e^{jw}) = \frac{1}{L \cdot A} \sum_{l=0}^{L-1} I_{W_y,W_y}^M(e^{jw}, l)$$  \hfill (6.8)
Next, using the estimation $\hat{C}^W_{y,y}(e^{jw})$ of the spectrum of the signal $y(n)$, we have to estimate the bandwidth of the signal $y(n)$. Unfortunately the spectrum is rough and it is hard to design an algorithm to detect its bandwidth. For this reason we decided to smooth the spectrum estimation with a rectangular window of length $S$, then the smoothed spectrum is

$$
\hat{C}^W_{y,y}(e^{jw}) = \hat{C}^W_{y,y}(e^{jw}) * sqS(e^{jwn})
$$

(6.9)

In the coarse estimation it is not needed to be precise in the estimate, which can be rough. We assume that the two points where the transitions (first derivatives) are maximum and minimum are the two points at middle height in the main lobe of the spectrum, then we approach the Equation (6.4) for

$$
\hat{B}_y = f_l - f_h
$$

(6.10)

where, $f_l$ and $f_h$ are defined respectively as

$$
f_l = \arg\max \frac{\partial}{\partial f} \hat{C}^W_{y,y}(f)
$$

(6.11)

$$
f_h = \arg\min \frac{\partial}{\partial f} \hat{C}^W_{y,y}(f)
$$

(6.12)

once we have the estimation $\hat{B}_y$ of the bandwidth we can estimate the coarse oversampling ratio, we define the bandwidth measured in hertz as

$$
\hat{B}_{w_y} = \frac{f_s}{2} \cdot \frac{B_y}{M} = \frac{f_s}{2} \cdot \frac{f_l - f_h}{M}
$$

(6.13)

where $f_l$ and $f_h$ were previously defined in Equations (6.12) and (6.11) as the points of maximal and minimal transition, and $M$ is the length of the FFT. In Figure 6.6 there is a comparison between the two methods to compute the bandwidth, the differential method (Equation 6.13) and the method that follows more strictly the definition of bandwidth (Equation 6.4). In Figure 6.6 it can be seen that the variance of the derivatives of the spectrum estimate are very high. To reduce the variance we have decided to compute more Welch spectrum estimate (of a shorter length), differentiate each one, and then average the derivatives in order to have a low variance derivative and, consequently, a clearer maximum and minimum points. See Figure 6.7 for a comparison of both derivatives computation techniques.

The oversampling ratio $N$ is defined as $N = \frac{f_s}{br}$, assuming the following relation between the symbol-rate $Br$ and the bandwidth $B_{w_y}$: $br = \hat{B}_{w_y}$ the roughly estimated oversampling ratio is

$$
\hat{N}_c = \frac{f_s}{br} = \frac{f_s}{B_{w_y}} = \frac{f_s}{\frac{f_s}{2} \cdot \frac{f_l - f_h}{M}} = \frac{M}{f_l - f_h}
$$

(6.14)

In conclusion, to estimate roughly the oversampling of the signal, first we compute different estimations of the spectra using Welch averaging Periodogram (Equation 6.8), then we smooth the spectra (Equation 6.9) to avoid the noise peaks, after that we average the derivatives of the different spectra estimations and compute the maximal and minimal points, (Equation 6.10) finally with Equation 6.14 we get the oversampling estimate.
Figure 6.5: Comparison between the FFT spectrum estimate and the Welch spectrum estimate

Figure 6.6: Differential method to estimate the bandwidth, notice that the variance of the derivatives of the estimated spectrum is very high
Figure 6.7: Averaged derivatives of the spectrum, notice that the variance is much more lower than the non averaged case.

Figures 6.8, 6.9 and 6.10 present the histograms of the coarse estimate for the different oversampling levels. In Figure 6.11 there is a detailed scheme of the coarse estimate system.

---

1 A histogram is the distribution of values of a random variable, very similar to the probability density function.
Figure 6.8: Histogram of the coarse estimate of the symbol-rate. Oversampling factors: 2 (above), 4 (middle) and 8 (below).
Figure 6.9: Histogram of the coarse estimate of the symbol-rate. Oversampling factors: 16 (above), 32 (middle) and 64 (below).
**Figure 6.10:** Histogram of the coarse estimate of the symbol-rate. Oversampling factors: 128 (above), 256 (middle) and 512 (below).

**Figure 6.11:** Detail of the coarse scheme
6.2.1.2 Decimate module

The decimate module decimates the input signal in order to reduce the oversampling factor in order to increase the performance of the fine stage. The decimate factor $N_d$, which depends on the coarse estimated oversampling $\hat{N}_c$, and is chosen trying to reduce the oversampling factor below 32 samples/symbol but guaranteeing that the new oversampling factor is above 2 samples/symbol.

$$2 < N_{rd} < 32$$

(6.15)

where $N_{rd}$ is the new oversampling factor (before the decimate module), $N_{rd}$ is defined as

$$N_{rd} = \frac{N_r}{N_d}$$

(6.16)

where $N_r$ is the true oversampling of the signal $y(n)$ and $N_d$ is the decimate factor of the decimate module, $N_r$ is defined as

$$N_r = \frac{f_s}{br}$$

(6.17)

being $br$ the symbol-rate and $fs$ the sampling frequency.

Table 6.1 is the Decimate Table where the relation between the coarse oversampling estimation and the decimate factor is presented, the decision of the thresholds has been done experimentally looking in the histograms of the coarse estimate (Figure 6.8, 6.9 and 6.10).
The scheme of the system can be seen in Figure 6.13. Notice that the objective of this module is just to reduce the oversampling to an acceptable value for the fine oversampling estimate module. This is the reason because of the Decimate Table (see Table 6.1) is very rough, it assigns the same decimate value to a broad range of coarse oversampling ratios $\hat{N}_c$. The coarse estimate stage classifies the incoming signals in high oversampling, medium oversampling and low oversampling values, and then the decimate module reduces the oversampling $N_r$ trying to put all the signals in to a low oversampling value $N_{rd}$.

\[
\begin{align*}
&\hat{N}_c \quad \text{decide} \\
&\quad \text{decimate} \\
&\quad N_d \\
&y(n) \quad \text{Decimate} \\
&y_d(n)
\end{align*}
\]

**Figure 6.13**: Scheme of the decimation module. The input signals are the estimated oversampling $\hat{N}_c$ and the input baseband signal $y(n)$, the output of the module is the decimated signal $y_d(n)$.

<table>
<thead>
<tr>
<th>Coarse oversampling estimate $N_c$</th>
<th>Decimate factor $N_d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N_c &gt; 350$</td>
<td>$N_d = 64$</td>
</tr>
<tr>
<td>$\hat{N}_c &gt; 250$</td>
<td>$N_d = 32$</td>
</tr>
<tr>
<td>$\hat{N}_c &gt; 100$</td>
<td>$N_d = 16$</td>
</tr>
<tr>
<td>$\hat{N}_c &gt; 60$</td>
<td>$N_d = 4$</td>
</tr>
<tr>
<td>$\hat{N}_c &gt; 33$</td>
<td>$N_d = 2$</td>
</tr>
<tr>
<td>else</td>
<td>$N_d = 1$</td>
</tr>
</tbody>
</table>

**Table 6.1**: Decimate table: decimate values for the different coarse oversampling estimates $\hat{N}_c$.

In Figure 6.14 there is the performance of the coarse estimation and the decimate module. The performance for low symbol-rates is perfect and is not shown. Notice that we consider that the coarse and decimate module work properly when the decimated signal $y_d(n)$ has an oversampling ration $N_{rd}$ between 2 and 32. The performance for low symbol-rates (from 8kbps to 128 kbps) is perfect since the signals are always correctly decimated. For signals with high symbol-rate (from 2046 kbps to 512 kbps) the performance is decreased, most of the errors corresponds to the choice of a big decimate factor $N_d$ with the consequence of destroying the signal. The histograms of the coarse oversampling estimation are presented in Figure 6.8. There can be seen how the performance of the oversampling estimation is too rough because the histograms of the low oversampling ratios have long tails. These long tails may produce a misclassification of the coarse estimation of the oversampling factor $\hat{N}_c$ and then the wrong decision for a properly decimation factor $N_d$. 
Figure 6.14: Performance of the coarse estimation and the decimate module for high symbol-rates. We consider that if the decimated signal $y_d(n)$ has an oversampling ratio $N_{rd}$ between 2 and 32 samples/symbol it has been correctly decimated, if the oversampling ratio is bigger than 16 (fine stage can not work) or smaller that 2 (signal destroy) we consider that the signal has been uncorrectly decimated.

6.2.1.3 Fine Symbol-Rate estimate

The second step of the algorithm consists of a fine estimation on the symbol-rate, since the symbol-rate has been previously estimated and the signal has been correctly decimated, the oversampling rate should be between acceptable values. The procedure starts with the computation of the power spectrum of $y_d(n)$ by using the Welch estimate (see Equation 6.8), then the inverse (fast) Fourier transform (IFFT) of $C_{y_d,y_d}(e^{jw})$

$$\hat{g}(n) = IFFT(C_{y_d,y_d}(e^{jw}))$$ (6.18)

where $C_{y_d,y_d}(e^{jw})$ is the estimation of the spectrum of $y_d(n)$. Equation 6.18 provides the magnitude of the raised cosine pulse $g(n)$, which exhibits a minimum next to zero, which corresponds to the oversampling rate $N_{rd}$ of the decimated signal. Then to estimate the oversampling rate we have to look for the first local minimum close to 0 of $|\hat{g}(n)|$

$$\hat{N}_f = \arg\min_n |\hat{g}(n)| = \arg\min_n IFFT(C_{y_d,y_d}(e^{jw}))$$ (6.19)

where arg min is a function that returns the position of the first local minimum. In conclusion: the fine stage consists on compute the IFFT (see Equation 6.18) of the Welch estimated spectrum (see Equation 6.8) and then look for the first minimum close to zero (see Equation 6.19). In Figure 6.16 there is a detailed scheme of the fine stage. One of the characteristics of this method is that it is supposed to be immune to the Doppler effect because it doesn’t use any spectral reference, and it uses the module operation in the time domain, then the effect of the frequency offset should be nearly 0. In Figure 6.15 there

1See Appendix A for an analysis of the different signals
are plots of different $|\hat{g}(n)|$ functions under different symbol-rates (oversampling values) and different SNR scenarios. Notice that the $X$ axis is marked with a $N$ which indicates the position where the minimum of the $|\hat{g}(n)|$ should be (marks the oversampling factor). It is important to notice that the position of the minimum moves while increasing oversampling (and then the performance of the algorithm), the reduction of the oversampling factor is important to achieve a good performance. The effect of the noise is different from the effect of the oversampling, while increasing the noise power the minimum becomes less sharp.

**Figure 6.15:** Different plots of the IFFT of a QPSK signal under different oversampling levels and different SNR. Notice that if the SNR decreases the minimum becomes less evident and if the oversampling increases the minimum changes its position.

**Figure 6.16:** Detailed scheme of the fine oversampling estimate. Notice that the input signal is the decimated based band signal $y_d(n)$ and that the output is $\hat{N}_f$ the fine estimation of the oversampling factor.
6.2.1.4 Full system

Once the three different parts of the system are described (coarse estimate, fine estimate and decimate module), now it is time to describe the behavior of the full system and also to analyze its performance.

In Figure 6.17 the full system is presented, the coarse estimate has as an input the incoming baseband signal $y(n)$, the output of the coarse stage is a coarse estimation of the oversampling $\hat{N}_c$. The next stage is the decimate module, where concerning to the estimated oversampling $\hat{N}_c$ the incoming signal $y(n)$ is decimated by a factor $N_d$ in order to decrease the oversampling of the signal. The decimated signal is called $y_d(n)$ and it is the input of the fine estimation stage where the oversampling factor $\hat{N}_f$ of the decimated signal $y_d(n)$ is estimated. Finally the decimate factor $N_d$ is combined with the fine oversampling estimate $\hat{N}_f$ to get the final estimation of the oversampling factor $\hat{N}$ and the estimation of the symbol-rate $\hat{br}$.

The final estimated oversampling $\hat{N}$ is $\hat{N} = N_d \cdot \hat{N}_f$, then the estimated symbol-rate is $\hat{br} = \frac{f_m}{\hat{N}}$.

The system designed should be capable to estimate the symbol rate in a broad range of oversampling values, and should also be robust against Doppler frequency offsets. In the following section 6.3 we will analyze its behavior.

![Figure 6.17: Final structure of the broad range symbol-rate estimator](image)

6.3 Experimental results

6.3.1 Simulation parameters

In the following sections we will explain the exact parameters of the algorithm, details of the simulation parameters and some extra features of the algorithm. Such parameters can

\footnote{the factor is chosen in order to reduce the oversampling below the 32\textit{samples/symbol} but without destroying the signal}
be adjusted or modified to increase the performance for specific scenarios. The current proposed values of such parameters has been decided experimentally.

6.3.1.1 Frame division

In order to increase the performance we divide each input stream of signal into 5 frames (with an overlap of the 50%), then each one of these frames is processed independently. After that, concerning the coarse estimated oversampling ratio $\hat{N}_{c,frame}$ of each frame, a common decimate factor $N_d$ is decided for all the frames (the complete input stream). Next all the frames are decimated (by a $N_d$ factor) and processed independently by the fine stage, after that stage there are 5 fine estimates of the symbol-rate $\hat{N}_{f,frame}$. A final thin estimation $\hat{N}_f$ is decided concerning each one of the 5 partial fine estimations. In Figure 6.18 is sketched the scheme of the system.

Figure 6.18: Diagram of the simulated system, notice that the input signal is divided in 5 frames.

The first decision block (after the coarse estimate) is designed to avoid the large errors (discarding the $\hat{N}_{c,frame}$ that are very different of the other ones). The second decision block uses the fact that the possible incoming symbol-rates are known.

6.3.1.2 Assumptions

In order to increase the performance here we have made the assumption that all the symbol-rates are powers of 2. So there will be just 9 different symbol-rates to decide between. The possible symbol-rates of the incoming signal belongs to the following symbol-rate list: 8kbps, 16kbps, 32kbps, 64kbps, 128kbps, 256kbps, 512kbps, 1024kbps and 2048kbps. So once we have a symbol-rate estimation $\tilde{br}$ we choose the closest symbol-rate from the symbol-rate list.

6.3.1.3 Simulation parameters

- Incoming signal
– Modulation: QPSK.
– roll-off factor: 0.1.
– symbol-rate: 8, 16, 32, 64, 128, 256, 512, 1024 or 2048 kbps.
– Sampling frequency: $4096 \cdot 10^3$ samples/second.
– SNR: between $-5$ dB and 7 dB (1 dB step) and two additional points at $-10$ dB and 20 dB.
– Incoming signal divided in 5 frames (overlap 50%).
– Each frame has about $2^{21}$ samples.
– The total number of samples is $2^{22.7}$.
– The total time of acquisition is 1.6635 seconds.

• Coarse estimate
  – Each frame is divided in 16 frames more. The spectrum is estimated with Welch.
  – Welch estimator
    * Number of histograms: 135 (50% overlap).
    * FFT length: $2^{15}$ samples.
    * Window type: rectangular ($w_M(n)$).
  – Spectrum smooth: rectangular window 75 samples ($S_{qs}(n)$).
  – Differentiate spectrum smooth: rectangular window 75 samples.

• Fine estimate (Welch)
  – Parameters depend on the decimate factor.
  – FFT length: between $2^{12}$ and $2^{13}$ samples (depend on decimate factor).
  – Number of histograms (50% overlap): between 64 and 135 (depend on decimate factor).
  – Window type: hamming ($w_M(n)$).
  – Samples per frame: between $2^{21}$ and $2^{15}$ (depend on decimate factor).

6.3.2 Experimental results

The modulation scheme chosen for the test was QPSK. The simulations where done in different SNR scenarios (from -5 dB to 7 dB with 1 dB steep and two additional points -10 dB and 10 dB) for each SNR value 50 different input signals where analyzed. The effect of the Doppler frequency is not considered in this simulations. The results are presented in 3 different Figures. In Figure 6.19 the performance under modulations with low symbol-rate is presented, notice that having low symbol-rate produce high oversampling, and hence is supposed to have worse performance than the modulations with higher symbol-rate. In Figure 6.20 the performance for modulations with high symbol-rate is presented. Notice that the performance is better than the performance under low symbol-rate, the reason is that the oversampling is small. In Figure 6.21 is compared the average performance of the low symbol-rate, the average performance of the high symbol-rate and the general average performance. Notice that the performance of the signals with low oversampling ratio is
nearly the same as the performance of the coarse stage for this kind of signals. That means that if the oversampling ratio $N_r$ is correctly estimated by $\hat{N}_c$ in the coarse stage, the fine stage will also estimate the the oversampling ratio properly.

**Figure 6.19:** Performance of the algorithm under scenarios with low symbol-rate
Figure 6.20: Performance of the algorithm under scenarios with high symbol-rate
6.3.2.1 Doppler effect: Bepicolombo

In this section, in order to check the Doppler robustness, we will simulate Symbol-rate estimate for the Bepicolombo (see Appendix A) mission. The parameters of the system are the same as described in Section 6.3.1.3 except for the signal parameters in this case we have OQPSK modulation (1024kbps and 512kbps), with a roll-off factor $\alpha$ of 0.1 and with a frequency offset caused by the Doppler of $\pm 200KHz$. In Figure 6.22 the performance is shown, we can see that the performances is very similar to the performance without frequency offset (see Figure 6.20).
6.3.3 Conclusions

We have developed an algorithm which is able to estimate the symbol-rate of a signal scenarios where the range of possible symbol-rates is broad. We have seen that an algorithm based in the proposed architecture (coarse estimate, decimate, fine estimate) can achieve the requirements of the satellite scenarios. We also have seen that, if the algorithm knows the possible symbol-rates and this symbol-rates are not close to each other, the performance of the fine stage is very high (if the coarse and decimate modules work properly). On the other hand we have seen the importance of the coarse estimation of the oversampling, because the general performance of the system depends directly on the coarse performance. From the study of the performance plots (Figures 6.19 and 6.20) we have observed that:

- For most of the symbol-rate values (16, 32, 64, 512 and 1024 kbps) the performance is very high for values of the SNR above −5 dB.

- The symbol-rates 8, 128 and 256 kbps have a very high performance for SNR above 0 dB.

- For 2048 kbps the performance is always above 80% but it achieves the 100% just above 5 dB.

From these results we can conclude that the performance is not related to the oversampling any more, because signals with a medium level of oversampling have better performance.
than signals with low/high oversampling. It still seems not to be clear a relationship between the oversampling and the performance.

Our suggestion to explain that behavior is that the performance of the whole system is closely related to the coarse stage and the Decimate Table.

**Signal with high oversampling:** apart from the high performance\(^1\) of the coarse stage for signals with high oversampling, the performance of the full system is not good enough. The main reason for this low performance is that the fine stage is not working properly. But how we explain that the fine stage is not working properly for this high oversampled signals when for the low oversampled signals is working properly? The reason is that besides of the coarse and decimate module work properly, the out coming decimated signal \(y_{rd}(n)\) has a decimate factor \(N_{rd}\) very close to 16 which is the boundary of high performance for the fine stage. To increase the performance we have to:

- Increase the fine stage performance for oversampling factors \(N_{rd}\) close to 16.
- Modify the coarse and decimate module to decimate more the signals.

**Signal with low oversampling:** the performance is directly related to the coarse performance. The reason for most of the failures is that the decimate module destroys the signal (decimate for a factor too big and then \(N_{rd}\) is smaller than 2). To solve that problem we have to prepare the coarse and decimate module to decimate less the signals.

Our conclusion is that to increase the performance the threshold values of the decimate table have to be optimized, in order to decimate more the high oversampled signals, and preserve the low oversampled signals. Optimize the thresholds value leads to an interesting trade-off because if we increase the decimate value to increase the estimation of low symbol-rates, the performance of the low oversampled signals will decrease. But in the other hand if the decimate value is decreased to achieve better performance in the high symbol-rate estimates, the signals with high oversampling value will reduce its performance. In Section 6.4 some tips to increase the performance are introduced.

**Frequency offset:** the algorithm was designed to be robust against the frequency offset produced by Doppler, we have confirm that its robustness is total because in the simulations there was no difference between the performance with frequency offset and the the performance without.

### 6.4 Further work and new proposals

We have seen that an algorithm based on the one presented here, should be able to estimate the symbol-rate of an incoming signal. One of the new features of this algorithm is that is capable of working under high oversampling ratios (work with very different symbol-rates). Here we introduce some new proposals in order to increase its performance and reduce its complexity.

---

\(^1\)Remember that the definition of performance for the coarse stage (coarse + decimate) is: number of decimated signals \(y_{rd}(n)\) with an oversampling ratio \(N_{rd}\) between 2 and 16 samples/symbol divided by the total number of trails
6.4.1 Time and complexity constrains

Complexity: To reduce the complexity of the algorithm we can use the fact that the spectrum of the incoming signal is computed twice (in the fine stage we compute the spectrum of the decimated signal \(y_d(n)\) and in the coarse stage of the incoming signal \(y(n)\)). We can use the relation between one signal \(y(n)\) and the same signal decimated \(y_d(n)\) to find a relationship between its spectra: \( \hat{C}_{W}^{y_d,y_d}(e^{jw}) \) from the decimated signal \(y_d(n)\), where \(y_d(n) = y(N_d \cdot n)\). From [15] we know that

\[
\hat{C}_{W}^{y_d,y_d}(e^{jw}) = \frac{1}{N_d} \sum_{k=0}^{N_d-1} \hat{C}_{y,y}^{w}(e^{j \left( \frac{w}{N_d} \frac{2\pi}{N_d} k \right)})
\]

(6.20)

with this transformation we avoid a second computation of the \(y_d(n)\) spectrum but we have to realize a sum on \(N_d\) spectrum vectors.

Acquire time: If the acquire time is not a constrain, the performance can be improved by increasing the number of samples of the algorithm and, as a consequence, the length of the FFT transforms and the number of Welch averages, achieving more frequency resolution and less variance in the spectra estimations.

6.4.2 Coarse estimate and decimate module

The coarse estimate is one of the most delicate steps of the algorithm. In fact for high symbol-rates the performance of the full-system is almost dependent on the coarse module. For this reason we think that a review of the algorithm for the estimation of the bandwidth can increase the performance. The algorithm uses 2 smooth filters to smooth the Welch spectrum and its first derivative. The length of this filters has an important role in the performance with a trade-off: if the filter is too long it will mask two close peaks (the minimum and the maximum of the derivate) and the performance for low symbol-rate signals will be decreased but if the filter is too small the signal will be too noisy. For this reason a variable length of the smooth filter will increase the overall performance.

The Decimate Table: the objective of the the decimate module is to decrease the oversampling of the signal below 32 samples/symbol without destroying the signal. The design of the Decimate Table (see Table 6.1) can be improved in order to increase the overall performance. The values of the threshold have been decided in accordance with the histograms of the coarse performance (see Figures 6.8, 6.9 and 6.10). Although they are not the optimal ones. Finding the optimal ones can be done easily by writing a program that looks for the thresholds that minimize the error probability concerning the histograms of the coarse stage. The optimal thresholds are the decimate table values that maximizes the provability of having a decimated signal with an oversampling between 2 and 32. An other way to increase the performance is to have different Decimate Tables (one general, one for low SNR, and other for medium SNR and one for high SNR). Then if we can make and estimation of the SNR we can run the algorithm for the symbol-rate estimation with a dedicated Decimate table.
6.4.3 Fine estimate

One of the assumptions of the algorithm was that the symbol-rates belong to a symbol-rate list. Nevertheless this is a very strict assumption and it reduces the range of applicable scenarios for the broad range symbol-rate estimation algorithm. If we assume that the symbol-rates belong to a longer symbol-rate list, we will have to take some other aspects into account in order to adapt our algorithm to the new requirements:

- The oversampling ratio $N_r$ will not be integer any more.
- It is possible to have close symbol-rates.

The presence of non integer oversampling ratio $N_r$ is important because our estimation of the oversampling ratio $\hat{N}_f$ corresponds to the position of the minimum. As a consequence the estimated oversampling will be always an integer value. To solve this problem (see [34]) we can make the IFFT transform with zero-padding that will give us a better resolution in the time domain and as a consequence the algorithm will be able to estimate non integer oversampling ratios (the resolution of this non integer oversampling ratios is proportional to the zero-padding). If the list of possible symbol-rate is not known (just the highest and the lowest) the fine decision block of the algorithm is useless, because we don’t have information to make a decision. The decision block can be replaced by another estimation stage to make a more precise estimate. For example the method described in [54] is a good candidate to refine the fine stage, since the estimation of the oversampling ratio can be used as an input to design the filter-bank of the algorithm.

In addition, the satellites have special modes of operation. For instances a satellite can enter in safe mode, the symbol-rates under this mode are very low (around 20 bps) (see Appendix C). We think that the best solution will be to run a dedicated algorithm for this mode of operation. If the described algorithm detects an extreme high oversampling ratio, then it starts the dedicated algorithm to detect very low symbol-rates.
Conclusions

In this work we have developed a Broad Range Symbol-rate Estimator (see Chapter 6) and additionally an Hybrid Modulation Classifier has been also designed (see Chapter 4). Both of them are designed to work inside an Autonomous Receiver (AR) (see Chapter 2) architecture. In the design phase we have taken in to account that our knowledge of the signal parameters is limited, in addition the proposed methods has different estimation modules: coarse and fine modules, modules that require additional information and modules that are blind. All these characteristics are needed if it is desired to fit the designed algorithm inside an AR architecture. In the following we will explain how this two estimation modules can fit inside an AR architecture. In addition the Broad Range Symbol-rate Estimator and the Hybrid Modulation Classifier will be explained and compared with other known methods.

Autonomous Receiver Architecture

As far as the author knowledge goes, the only published document containing a complete architecture of an AR was published by JPL [17]. In the following the differences and similarities between the JPL AR architecture and a possible ESA AR architecture are commented. The scenarios differences between ESA and JPL missions may lead to different architectural designs, however there are some parts and ideas of the JPL design that can be useful for the ESA design. The requirements with which the JPL AR was designed, provide, in some aspects, easy solutions to complicate problems. In the other hand the general requirement for the ESA AR are not decided yet. The requirements for a design are in more detail: the degree of general purpose of the AR, the range of different scenarios, the performance of the algorithm, the level of interferences, the complexity constrains, the delay constraints, etc. control the degree of complexity of the design. As a consequence it is important to decide the requirements that will drive the architectural design of the AR. The main difference between the ESA and JPL scenarios and requirements is related with the modulation scheme:

- JPL: they just consider the presence of MPSK modulations (being \( M = 2, 4...N \)). Because of the similarities between these modulations, most of the estimation modules of the AR don't need an estimation of \( M \), because they are designed for arbitrary values of \( M \). For example the SNR estimator, the data format, coarse modulation index and the data rate don't need the estimation of \( M \).

- ESA: uses different kinds of modulations, that are quite different between them.
They use modulations with carrier (PM on carrier), with subcarrier (PM on sinusoidal/square sub carrier) and suppressed carrier modulations such as OQPSK ans GMSK, in addition spread spectrum modulations are used but are not considered in this work.

The most challenging design parameter are the similarities/differences between the different modulations. Because of this similarities, most of the different estimation modules of the JPL AR are designed to work for all kind of MPSK modulations. In addition in the JPL design the MC module is one of the last ones (See Section 2.2 and Figures 2.2 and 2.3). As a consequence the estimation of $M$ is just used in the fine estimation mode. On the contrary, due to the big difference between the modulations that ESA wants to demodulate autonomously the different estimation modules have to be specially designed to work for a specific modulations.

In this work it is proposed, for the first step of the algorithm, to make a coarse classification of the signal in two groups: Analog signals (carrier) and Digital signals (suppressed carrier). If this early classification is done, the other AR estimation modules design will became easier, because its design can be more dedicated to a specific small group of modulations. On the contrary without this first classification the estimation modules should be able of working under very different modulation schemes.

To be more specific, the symbol-rate estimation proposed in JPL is modulation type independent ($M$ independent), however it can not be used for all the different modulations of ESA. As a consequence, in the ESA case the coarse symbol-rate estimation can consist on different modules, one for the Analog signals and one for the Digital signals. These modules are less general than a simple Symbol-rate estimation for all the modulation schemes, and as a consequence they will be easier to design.

The first coarse modulation classification, provides enough modulation scheme information to run dedicated symbol-rate estimation algorithms. As a consequence the symbol-rate estimation will allow the use of ML methods, or more precise PR methods for the modulation classification.

In Figure 6.23 we shown the interaction between the MC module and the symbol-rate estimation module, the other estimation modules are not shown, just its outputs are used. Notice that the estimation of both parameters consist on different different steps, making at each steps finer estimates and using the results from the previous steps in order to ultimately estimate the modulation scheme and the symbol rate. It can be seen how at each step of the algorithm, there is more information available of the signal (estimations of other modules), once this additional estimations are available the estimations becomes more accurate. Notice also how the information of the previous steps is used to increase the accuracy of the current steps. Additionally it is shown how the different dedicated algorithm for the parameter estimation are selected once other parameters are known. For instance once the modulation is classified as a OQPSK, then the parameters of the broad range symbol-rate estimation are optimized for the OQPSK signals, however in the previous coarse symbol-rate estimation the parameters where set as general parameters for the GMSK and OQPSK modulations.
Figure 6.23: Scheme of the interaction of the modulation classifier (MC) and the symbol-rate estimator. Notice that the yellow boxes are the modules that have been designed in this work, in addition the blue boxes are the modules that have been explained and briefly designed.
Modulation classification and symbol rate estimation modules

Hybrid modulation classifier: Analog vs. Digital classifier

The Analog vs. Digital classifier has been designed to be a blind estimator (no need for information about signal parameters). For this reason it is capable of working in a broad range of signal parameters, modulation schemes and interferences. The A. vs. D. classifier detects the presence of two data lobes (analog signals) using the correlation spectrum. The detection of data lobes in the correlation spectrum allows the A. vs. D. classifier to distinguish between the carrier modulation and the suppressed carrier modulations. We have designed an algorithm which performance and requirements are between the Maximum Likelihood (ML) and the Patter Recognition (PR) methods. It has a good performance above 5 dB, which is better than the PR methods but worst than the ML methods. The proposed algorithm does not need a priori knowledge of the signals, on the contrary all the ML methods and some PR methods need additional information of the signal. In conclusion the A. vs. D. classifier is a flexible method capable of working in many different scenarios which has an acceptable performance besides not needing any a priori information of the signal.

In addition it has been proposed an Hybrid modulation classifier for the classification of the ESA modulations. It consists of a combination of PR methods and ML algorithms. It first classifies the signal with an A. vs. D. classifier, later on the estimation is refined using ML methods and advanced PR algorithms. The multiple step structure of the Hybrid modulation classifier is applicable to the AR architectures. In addition the required knowledge of the signal is zero for its first step which presents an advantage to the other modulation classification methods.

Broad range symbol-rate estimator

The design of the Broad range symbol-rate estimator has been centered in solving the problem of the signal oversampling. Since the symbol-rate of the incoming signal is unknown, the signals are always sampled at maximal ratio. As a consequence for low symbol-rates the signal is highly oversampled. In addition while the oversampling of the signal increases the performance decreases for most of the symbol-rate estimation techniques. The design has been focused on designing an algorithm capable of working under very high oversampling values.

The developed algorithm uses two stages to estimate the symbol-rate. The first stage makes a coarse estimation of the oversampling an decimates the signal. As a consequence the signal oversampling is reduced. The second step, which consist of a symbol-rate estimation based on the IFFT, is able of estimate the symbol-rate with an acceptable performance. In addition the algorithm presents an interesting robustness against the Doppler shift. On the contrary, the algorithm presents a lack of generality since it is designed to work only for GMSK and OQPSK modulations. The different parameters of the algorithm can be adapted to increase the performance for one of both modulation. Nevertheless the algorithm is not designed to work for the Analog modulations, as a consequence it is needed to check its performance or to look for some alternatives.
In conclusion we have developed a symbol-rate estimation method, that can work under large oversampling factors, and that has a very good performance under low SNR scenarios. In addition it is robust against the Doppler shift. However it needs information about the modulation scheme, as a result this method has to be used after the MC algorithm has done an estimation of the signal modulation because the broad-range symbol-rate estimation works only for the GMSK and OQPSK modulations.
Part IV

Appendixes
Appendix A

Analysis of the Telecommand and Telemetry Modulations

There are eight different modulation schemes the autonomous receivers have to detect and demodulate (see AD-6), even if only three of them can be in competition for a given scenario. In the following paragraphs the spectral characteristic of these modulations are summarized. In whole document we have called the three modulations with carrier (SPL/PM and the two PM on subcarrier) Analog modulations and the suppressed carrier modulations (GMSK and OQPSK) Digital modulations.

A.1 Carrier Modulations: SP-L/PM

Carrier and Sub Carrier modulations are commonly used in Telecommand and Telemetry functions. The power used by the transmitter is split between the power of the data and the power of the carrier. The presence of a carrier allows the receiver PLL to lock to the carrier, this kind of signals are also used to make measures of the Doppler and the Ranging of the satellite. However just a part of the transmitted power can be used to transmit data because part of the power is spend in the carrier.

This modulation carries out a Manchester (SP-L) coding on the PCM input data, and the resulting signal directly PM modules the $f_c$ carrier with a modulation index $\beta$ between 0.1 and 1.5 (radians).

$$y(t) = \sqrt{2}A \sin(2\pi f_0 t + \beta b(t))$$  \hspace{1cm} (A.1)$$

where the $\theta$ is the modulation index (in radians) and $f_0$ is the carrier frequency, and $b(t)$ is the bipolar data to transmit. The above equation can be written as

$$y(t) = \sqrt{2}A \cos(\beta) \sin(2\pi f_c t) + \sqrt{2}Ab(t) \sin(\beta) \cos(2\pi f_c t)$$  \hspace{1cm} (A.2)$$

we have used that

$$\cos(\beta b(t)) = \cos(\beta)$$
$$\sin(\beta b(t)) = b(t) \sin(\beta)$$
It is seen that the carrier power denoted by $P_c$, and the data power denoted by $P_d$, are given by

\[ \frac{P_c}{P} = \cos^2(\theta) \quad (A.3) \]
\[ \frac{P_d}{P} = \sin^2(\theta) \quad (A.4) \]

Then the baseband equivalent signal is

\[ y_I(t) = \cos(\beta) - jb(t)\sin(\beta) \quad (A.5) \]

Because of the SP-L coding there is no signal power around the $f_c$ carrier. The $f_0$ band $Bw_{f_0}$ is then

\[ Bw_{f_c} = 2\left(\frac{3}{4}br + \frac{br}{2}\right) = 2.5br \quad (A.6) \]

where $br$ is the bit-rate.

In Figure A.1 a typical spectrum of SP-L signal is shown, with a bit rate of 256Kbps and a modulation index of 0.7 rad.

![Figure A.1: Spectrum of a PM on carrier modulation. $\beta = 0.7$ and $br = 256$kbps](image)

### A.2 Sub Carrier Modulations

When it is desired to utilize residual carrier telemetry systems it is sometimes advantageous to employ a subcarrier to place the data away from the carrier in order to eliminate interference effects caused by the presence of a carrier in the same bandwidth of the data [25].

#### A.2.1 PCM/PSK(sine)/PM modulation

In this modulation scheme the PCM data (NRZ-L or NRZ-M coded) module BPSK a subcarrier at frequency $f_{sc}$. The subcarrier module PM the $f_c$ carrier with a modulation index $\beta$ between 0.1 and 1.5 radians [44]. With such modulation index range the information is concentrated in the first Bessel image, and the signal appears like amplitude modulated,
with a non constant envelop\cite{45}. Many times the sinusoidal sub carriers are used for ranging purposes or for band limited systems that would not pass a squarewave subcarrier without losing the harmonics anyway (see subsection A.2.2). We model the signal as

\[ y(t) = \sqrt{2}A \sin(2\pi f_c t + \beta \sin(2\pi f_{sc} t + \pi b(t)) + \theta_0) \]  \hspace{1cm} (A.7)

where \( A^2 \) is the transmitted power, \( b(t) \) is the NRZ data, and \( \theta_0 \) is the carrier phase. Expanding, we have

\[
y(t) = \sqrt{2}A \sin(2\pi f_c t + \theta_0) \cos\{\beta \sin(2\pi f_{sc} t + \pi b(t))\} \\
+ \sqrt{2}A \cos(2\pi f_c t + \theta_0) \sin\{\beta \sin(2\pi f_{sc} t + \pi b(t))\} \\
= I(t) \cos(2\pi f_c t + \theta_0) - Q(t) \sin(2\pi f_c t + \theta_0) \]  \hspace{1cm} (A.8)

Using the Bessel decomposition, we obtain the following relationship between the components, form \cite{25}

\[
\frac{P_c}{P} = J_0^2(\theta) \hspace{1cm} (A.9)
\]

\[
\frac{P_d}{P} = 2J_1^2(\theta) \hspace{1cm} (A.10)
\]

with the remaining power contained in inter modulation components. The baseband expression is

\[
y_i(t) = I(t) + jQ(t) \\
= \sin\{\beta \sin(2\pi f_{sc} t + \pi b(t))\} - j \cos\{\beta \sin(2\pi f_{sc} t + \pi b(t))\} \\
= -je^{j\beta b(t)} \sin(2\pi f_c t) \]  \hspace{1cm} (A.11)

The \( br \) is selected to be a submultiple of the subcarrier frequency normal values for the bitrate are 4 or 5 times less than the subcarrier frequency. Being a double sideband signal, the bandwidth \( Bw_{fc} \) of the signal is

\[ Bw_{fc} = 2(f_{sc} + \frac{Bw_{BB}}{2}) \]  \hspace{1cm} (A.12)

where \( Bw_{BB} \) is the base band bandwidth, roughly equal to the bit rate \( br \). In figure A.2 there is an example of PCM/PSK(sin)/PM spectrum.
A.2.2 PCM/PSK(square)/PM(modulation)

This modulation scheme differs from the PCM/PSK(sine)/PM for the fact that the subcarrier is a square wave instead of a pure sine. The power spectrum is wider, but if only the first Bessel lobes are filtered the signal characteristics are similar to the PCM/PSK(sine)/PM\cite{15}. The signal can be modulated in the form

\[ y(t) = \sqrt{2}A \sin(2\pi f_c t + \beta f_{sc} \text{rect}(2\pi f_{sc} t) + \theta_0) \] (A.13)

Expanding, we have

\[
y(t) = \sqrt{2}A \sin(2\pi f_c t + \theta_0) \cos(\beta \text{rect}(2\pi f_{sc} t)) \\
+ \sqrt{2}A \cos(2\pi f_c t + \theta_0) \sin(\beta \text{rect}(2\pi f_{sc} t)) \\
= I(t) \cos(2\pi f_c t + \theta_0) + Q(t) \sin(2\pi f_c t + \theta_0) \] (A.14)

It is seen that the carrier power denoted by \(P_c\), the data power denoted by \(P_d\), and the total power denoted by \(P\), are given by

\[
\frac{P_c}{P} = \cos^2(\theta) \\
\frac{P_d}{P} = \sin^2(\theta) \] (A.15) (A.16)

Which is the same relationship is the same as the SP-L modulation. The baseband expression is

\[
y_I(t) = I(t) + jQ(t) \\
= \sin(\beta \text{rect}(2\pi f_{sc} t + \pi b(t))) - j \cos(\beta \text{rect}(2\pi f_{sc} t)) \\
= -j e^{j\beta \text{rect}(2\pi f_{sc} t)} \] (A.17) (A.18)

The \(br\) is selected to be a submultiple of the subcarrier frequency normal values for the bitrate are 4 or 5 times less than the subcarrier frequency. If we select the \(f_{sc}\) to be the
75% of the \( br \) and if the input data is randomly generated this modulation can model a carrier modulation (SP-L/PM). In figure A.3 there is an example of PCM/PSK(square)/PM spectrum.

![Figure A.3: Spectrum of a PCM/PSK(square)/PM modulation with \( f_{sc} = 400kHz \) and factor = 4 with modulation index \( \beta = 1 \)](image)

A.3 Suppressed carrier modulation

Suppressed carrier modulations (Digital modulations) are commonly used in Telemetry functions, when the symbol-rate required is to big to use an analog modulation (modulations with carrier or sub carrier). Suppressed carrier modulations are commonly used for symbol-rates above 0.5Mbps.

A.3.1 OQPSK

The modulation Offset-QPSK consist in a QPSK modulation where the maximum instantaneous phase shift has been limited. As long as in a normal QPSK modulation exist phase changes of \( \pm \pi \) rad. between two neighbors symbols, in the OQPSK modulation this phase changes are reduced to the half \( \pm \frac{\pi}{2} \) rad. avoiding the the in phase component \( I(t) \) and the quadrature component \( Q(t) \) to change simultaneously. In satellite scenarios where high power amplifier are used, the OQPSK presents advantages because of the phase change limitation since the non linearity of the high power amplifiers is increased with the amplitude of the phase changes[11]. To avoid both components change simultaneously in the transmitter and reduce the amplitude variation, the quadrature component \( Q(t) \) is delayed half symbol \( \frac{T}{2} \). The expression for a OQPSK signal is [42]

\[
y(t) = \sqrt{P} I(t) \cos (2\pi f_c t + \theta_c) - \sqrt{P} Q(t) \sin (2\pi f_c t + \theta_c),
\]

\[
I(t) = \sum_{n=-\infty}^{\infty} a_{I_n} p(t - nT_s)
\]

\[
Q(t) = \sum_{n=-\infty}^{\infty} a_{Q_n} p(t - (n + \frac{1}{2})T_s)
\]

(A.19)

Then the baseband expression for the OQPSK modulation is
\[ y_t(t) = I(t) + jQ(t) \]
\[ = \sum_{n=-\infty}^{\infty} a_{I_n} p(t - nT_s) + j \sum_{n=-\infty}^{\infty} a_{Q_n} p(t - (n + \frac{1}{2})T_s) \]  \hspace{1cm} (A.20)

\[ \cos(2\pi f_c t + \theta_c) \]

\[ \sin(2\pi f_c t + \theta_c) \]

Figure A.4: OQPSK transmitter

Figure A.5: Evolution of the in phase component \( I(t) \) and the quadrature component \( Q(t) \) of a OQPSK modulation.
In the ESA standards the OQPSK modulation uses always a raised cosine $\text{SRRC}$ as a pulse $p(t)$ to filter the data $a(n)$, the we define $p(t)$

$$p(t) = \frac{\sin(\pi t/2)}{(\pi t/T)} \cdot \frac{\cos(\pi \alpha t/T)}{(1 - 4\alpha^2 t^2/T)}$$  \hspace{1cm} (A.21)$$

where $T$ is the symbol period and $\alpha$ is the roll-off factor. Because of this data shaping the envelop is not constant. In Figures A.6 and A.7 the spectrum of two OQPSK signals is shown.

![Figure A.6: OQPSK spectrum with roll-off factor of 0.1 and 1Mbps](image)

**Figure A.6:** OQPSK spectrum with roll-off factor of 0.1 and 1Mbps

![Figure A.7: OQPSK spectrum with roll-off factor of 0.4 and 1Mbps](image)

**Figure A.7:** OQPSK spectrum with roll-off factor of 0.4 and 1Mbps

### A.3.2 GMSK

GMSK was first introduced by Murota, Kinoshita, and Hirada [29] in 1981 as a highly bandwidth-efficient constant envelope modulation scheme for communications in the 900-MHz land mobile radio environment [42]. Recently The GMSK modulation has been adopted by CCSDS (Consultative Committee for Space Data Systems) as a Standard for future space missions, due to its compact power spectrum, high immunity against interferences and capacity of supporting various receiver structures [35]. The GMSK can be
obtained by filtering a 2-PAM\(^1\) signal with a Gaussian filter and then using an FM analog modulator, as shown in figure A.8. There are other ways to obtain a GMSK modulation, for example a couple of decades ago Lauren \(^{32}\) described the representation for CPM in the form of a superposition of phase-shifted amplitude-modulation pulse (AMP). Here to generate the GMSK modulation we will use the the Gaussian filter and FM modulator approach presented in \(^{42}\).

![Figure A.8: Basic GMSK modulator](image)

The first step is conversion of a binary data stream \(a(n)\), \(a \in (0, 1)\), to an antipodal sequence \(b(n)\), \(b \in (-1, 1)\), and to a stream of rectangular pulses \(r(t)\)

\[
r(t) = \sum_n b(n) P(t - nT_b)
\]

where

- \(P(t) = 1\) \(t \in (0, T_b)\)
- \(P(t) = 0\) otherwise
- \(T_b = \frac{1}{T_b}\) is the symbol interval.

The next operation is Gaussian filtering of \(r(t)\)

\[
g(t) = r(t) * h(t)
\]

where \(g(t)\) is the filtered pulse stream and \(h(t)\) is the response of Gaussian filter\(^2\). Modulation of the carrier wave frequency by \(g(t)\) around a center frequency \(f_c\) implies modulating its phase \(\Phi\) by \(\int g(t)\)

\[
\Phi(t) = 2\pi f_c \cdot t + 2\pi \cdot f_m \cdot \int_0^t g(\tau) d\tau
\]

where the phase has been normalized to 0 at \(t = 0\). The instantaneous frequency of the modulated signal will be

\[
f(t) = \frac{1}{2\pi} \frac{\partial \Phi}{\partial t} = f_c + f_m \cdot g(t)
\]

where \(f_c\) is the carrier frequency and \(f_m\) is the peak frequency deviation. A modulated carrier is generated. The voltage controlled oscillator (VCO) or equivalent digital synthesis methods can produce directly

\[
y(t) = \cos(\Phi(t))
\]

that can be also written as

\[
y(t) = \cos(2\pi f_c t) \cos \left(2\pi f_m \int_0^t g(\tau) d\tau\right) - \sin(2\pi f_c t) \sin \left(2\pi f_m \int_0^t g(\tau) d\tau\right)
\]

\(^1\)Pulse Amplitude Modulated
\(^2\)Gaussian filter in the frequency domain: \(H(\omega) = TF(h(n)) = A_0 e^{-\alpha \omega^2} e^{-j\omega \tau_0}\).
or,
\[ y(t) = I(t) \cdot \cos(\omega_c \cdot t) - Q(t) \cdot \sin(\omega_c \cdot t) \]  
(A.28)

\( I(t) \) and \( Q(t) \) can be written as
\[ I(t) = \text{Re}\{\exp(j2\pi f_m \int_0^t g(\tau) d\tau)\} = \text{Re}\{u(t)\} \]  
(A.29)

and
\[ Q(t) = \text{Im}\{u(t)\} \]  
(A.30)

Then the expression of the baseband equivalent of a GMSK modulated signal is
\[ y_I(t) = I(t) + jQ(t) = \text{Re}\{u(t)\} + j\text{Im}\{u(t)\} = u(t) = e^{j2\pi f_m \int_0^t g(\tau) d\tau} \]  
(A.31)

**Pulse Response:** to study the response of the Gaussian filter to an incoming pulse train \( r(t) \), we will study the response \( p(t) \) of the Gaussian filter to an incoming pulse \( P(t) \) corresponding to one bit, from [19]
\[ p(t) = P(t) * g(t) \]  
(A.32)

\[ p(t) = \frac{1}{2}\{\text{erf}(\beta(t - t_0)) - \text{erf}(\beta(t - t_0 - T_b))\} \]  
(A.33)

where \( \text{erf}(x) \) is the error function, defined as [37]
\[ \text{erf}(x) = \int_0^x \frac{2}{\sqrt{\pi}} e^{-x^2} dx = -\text{erf}(-x) \]  
(A.34)

and \( \beta \) is
\[ \beta = \sqrt{\frac{2}{\ln(2)}} \pi 4f_m B_b T \]  
(A.35)

**Pulse Train Response:** Linearity allows us to apply superposition to find the response to a train of NRZ pulses, the response is expressed as a the combination of a pulse train varying between 0 and 2 minus a constant -1
\[ g(t) = \sum_n a(n)\{\text{erf}(\beta(t - t_0)) - \text{erf}(\beta(t - t_0 - (n + 1)T_b))\} - 1 \]  
(A.36)

Equation (A.33) shows that the pulse response extends theoretically from \(-\infty\) to \(+\infty\), so the response of to a pulse inside a given symbol interval has interference from all past and future symbols. In practice however the extent of the pulse response is usually considered limited to only the two nearest neighbors [18]. From [18] we know that using the two nearest neighbors approximation the response in the symbol interval is
\[ g_m(t) \approx a_{m-1}(1 - \text{erf}(\beta t)) + a_m(\text{erf}(\beta t) - \text{erf}(\beta(t - T_b))) + a_{m+1}(1 + \text{erf}(\beta(t - T_b))) - 1 \]  
(A.37)
As the pulses can have one or two possible polarities ($a_m = 1$ or 0), we can have $2^3 = 8$ possible curves. We can show that there are only three independent possible curves (values of $g(t)$). Since the frequency of the modulated signal is proportional to $g(t)$, we can show that there are just 3 independent possible frequency trajectories for the modulated signal. The concept of frequency trajectory allows us to express the modulating signal $g(t)$ as a train of baseband pulses, as follows

$$g(t) = \sum_n f_{l(n)}(f - nT)P(t - nT) \quad (A.38)$$

where $l(n) = 4a_{n-1} + 2a_n + a_{n+1}$ is the index of the trajectory. In table A.1 there are the formulas for the different trajectories that corresponds to the different binary sequences.

<table>
<thead>
<tr>
<th>Index</th>
<th>Equation</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>$f_0(t) = -1$</td>
<td>$f_0(t) = -f_7(t)$</td>
</tr>
<tr>
<td>001</td>
<td>$f_1(t) = \text{erf} (\beta (t - T_b))$</td>
<td>$f_1(t) = -f_6(t)$</td>
</tr>
<tr>
<td>010</td>
<td>$f_2(t) = \text{erf} (\beta t) - \text{erf} (\beta (t - T_b)) - 1$</td>
<td></td>
</tr>
<tr>
<td>011</td>
<td>$f_3(t) = \text{erf} (\beta t)$</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>$f_4(t) = -\text{erf} (\beta t)$</td>
<td>$f_4(t) = -f_3(t)$</td>
</tr>
<tr>
<td>101</td>
<td>$f_5(t) = 1 - \text{erf} (\beta t) + \text{erf} (\beta (t - T_b))$</td>
<td>$f_5(t) = -f_2(t)$</td>
</tr>
<tr>
<td>110</td>
<td>$f_6(t) = -\text{erf} (\beta (t - T_b))$</td>
<td>$f_6(t) = -f_3(t - T_b)$</td>
</tr>
<tr>
<td>111</td>
<td>$f_7(t) = 1$</td>
<td></td>
</tr>
</tbody>
</table>

Table A.1: Frequency trajectories

Finally using Equations A.38 and A.31 the baseband expression $y_I(t)$ for a GMSK modulated signal is

$$y_I(t) = e^{j2\pi f_m \int_0^t \sum_n f_{l(n)}(t-nT)P(t-nT)\,dt}$$

$$y_I(t) = e^{j2\pi f_m \sum_n \int_0^t f_{l(n)}(t-nT)P(t-nT)\,dt} \quad (A.39)$$

To implement the GMSK in MATLAB we have to work in the discrete domain, that is: we have to sample the baseband signal by a sampling frequency $f_m$

$$y_I(k) = e^{j2\pi f_m \sum_n \sum_{i=0}^k f_{l(n)}(\frac{i}{f_m} - nT)P(\frac{i}{f_m} - nT)} \quad (A.40)$$

In Figures A.10 and A.11 there are two examples of GMSK functions with common parameters in the ESA missions, in Figure A.9 the different frequency trajectories are shown.
Figure A.9: Frequency trajectories of a GMSK signal with $BT = 0.5$

Figure A.10: Spectrum of a 1Mbps GMSK with a $BT = 0.5$ and a $f_m = 4Ms/s$

Figure A.11: Spectrum of a 1Mbps GMSK with a $BT = 1$ and a $f_m = 4Ms/s$
A.4 MATLAB code

A.4.1 GMSK

function\ [\text{\text{baseband data options}}}]=gmsk\_fm(bt,os,num\_sym,snr,f\_offset)
% %%% input parameters
% bt: parameter
% os: oversampling parameter
% num\_sym: number of symbols
% snr signal to noise ratio of the system
% f\_offset: doppler effect of the channel

% %%% output parameters
% baseband: signal modulated in baseband
% data: bits of the modulation
% options: parameters of the modulated signal

fm=4096; %sampling frequency
br=fm/os; %bit rate
freq\_dev=br/4; %frequency deviation
Tb=1/br; % symbolt time (bit time)
B=bt/Tb;
bits\_x\_symbol=1;% number of bits per symbol
beta=pi*bt*sqrt(2/log(2))*br; % beta parameter
t=1/(2*fm):1/fm:Tb-1/(2*fm); %times of sampling of a single bit
% total times of sampling of the whole signal
t\_tot=1/(2*fm):1/fm:num\_sym/br-1/(2*fm);
data=randint(num\_sym,1,[0 1]); % data bits

%%%%%Table of the filtered pulse by a gaussian filter %%%%%%
f(1,:)=-1.*ones(size(t));
f(2,:)=erf(beta*(t-Tb));
f(3,:)=erf(beta*t)-erf(beta*(t-Tb))-1;
f(4,:)=erf(beta*t);
f(5,:)=-f(4,:);
f(6,:)=-f(3,:);
f(7,:)=-erf(beta*(t-Tb));
f(8,:)=-f(1,:);
smf=cumsum(f'); %integral of each filtered pulse

% generate the base band signal
kk=1;
index(kk)=data(kk)*2+data(kk+1)+1;
p1=1+(kk-1)*os;
p2=os*kk;
int\_sum\_phi(p1:p2)=sumf(:,index(kk));
for(kk=2:num_sym-1)
    % get the 3 bits and decide for the index of the table
    index(kk)=data(kk-1)*4+data(kk)*2+data(kk+1)+1;
    p1=1+(kk-1)*os;
    p2=os*kk;
    % sum of the different integrals
    int_sum_phi(p1:p2)=sumf(:,index(kk)) + int_sum_phi(p1-1);
end

kk=num_sym;
index(kk)=data(kk-1)*4+data(kk)*2+1;
p1=1+(kk-1)*os;
p2=os*kk;
int_sum_phi(p1:p2)=sumf(:,index(kk)) + int_sum_phi(p1-1);

phi=int_sum_phi;
phase=phi.*2*pi*freq_dev/fm;
baseband=exp(j*phase).*exp(j*2*pi*f_offset*t_tot); % frequency offset

%%% add noise
if(isinf(snr)==1) % add noise to the signal
    EmNo=snr +10*log10(bits_x_symbol) - 10*log10(os);
    baseband= awgn(baseband,EmNo,'measured') ;
else
end

options.time=t;
options.snr=snr;
options.mod='GMSK';
options.oversampling=os;
options.bits_x_symbol=bits_x_symbol;
options.simulation_symbols=num_sym;

A.4.2 OQPSK

function [mod_signal, data_bits, options]=
oqpsk( alpha, os, simulation_symbols,cod_type,snr)

% os= oversampling
% alpha rolloff factor
% simulation_symbols=number os symbols of the simulation
% cod_type type of codification of the data
% snr signal to noise ratio
bits_x_symbol=2;
ini_phase=0;

data=randint(simulation_symbols,1,[0 3]);

% codification of data
if (strcmp(cod_type,'no_cod')==1)
    cod_data=data;
elseif (strcmp(cod_type,'NRZ-L')==1)
    cod_data=cod_NRZ-L(data);
elseif (strcmp(cod_type,'Q-DNRZ')==1)
    cod_data=cod_Q-DNRZ(data);
end

% create the modmap
modmap = zeros(1,4);
modmap(1) = cos(ini_phase + pi/4) + j* sin(ini_phase + pi/4);
modmap(2) = cos(ini_phase + 7*pi/4) + j * sin(ini_phase + 7*pi/4);
modmap(3) = cos(ini_phase + 3*pi/4) + j * sin(ini_phase + 3*pi/4);
modmap(4) = cos(ini_phase + 5*pi/4) + j * sin(ini_phase + 5*pi/4);

% modulate
y = genqammod(cod_data,modmap);

% separate the signal into I and Q
yI = real(y);
yQ = imag(y);

% filter I and Q
[fyI,t] = rcosflt(yI,1,os,'default',alpha);
[fyQ,t] = rcosflt(yQ,1,os,'default',alpha);

% introduce the timing offset in the quadrature channel.
yQ = [zeros(os/2,1); fyQ(1:end,:) ];%shift T/2
yI = [fyI;zeros(os/2,1)];
y = yI + j*yQ;

% introduce noise
if(isinf(snr)==1)
    mod_signal=y;
else
    EmNo=snr +10*log10(bits_x_symbol) - 10*log10(os);
    mod_signal= awgn(y,EmNo,'measured') ;
end

% save data
data_bits=cod_data;
options.cod=cod_type;
options.t=t;
options.snr=snr;
options.mod='oqpsk';
options.oversampling=os;
options.alpha=alpha;
options.bits_x_symbol=bits_x_symbol;
options.simulation_symbols=simulation_symbols;
Appendix B

Review of the application scenarios

B.1 ESA missions

Information is extracted from [45]. Additional not classified information is available at the ESA we page.

1. BepiColombo (Deep Space): an ESA and JAXA mission aimed at exploring Mercury. It consists in two different spacecraft orbiting around the planet: the Mercury Planetary Orbiter (MPO) to study the surface and the internal composition of the planet and the Mercury Magnetospheric Orbiter (MMO), designed by the Japanese Space Agency, to study the Magnetosphere (the region around Mercury where the magnetic field is dominant).

2. Lisa Path Finder (Lagrange X-Band): an ESA mission aimed at detecting the gravitational waves produced by massive black holes in the frequency range [10^{-4}, 10^{-1}] which is not accessible from interferometers at the Earth surface. The LISA mission consists of three identical spacecrafts located on vertices of an equilateral triangle with edge length of 5 \cdot 10^6 Km. The center of the triangle is in the elliptic plane 1 AU from the Sun and 20 degrees behind the Earth. The triangle belongs to a plane which is inclined by 60 degrees with respect to the ecliptic. Each spacecraft is equipped with a 30 cm steerable antenna used for transmitting data to the Earth.

3. Gaia (Lagrange X-Band): an ESA mission aimed at making the largest, most precise, 3-dimensional map of our galaxy. The spacecraft will be placed in an orbit around the sun at the second Lagrange Point L2. For the Sun-Earth system the L2 point lies at a distance of 1.5 million Km from the Earth in the anti-Sun direction and co-rotates with the Earth in its first-year orbit around the Sun. One of the principal advantages of the L2 orbit is that it offers an uninterrupted observation since the Earth, the Moon and the Sun lay within the orbit of the L2 point. From L2 the entire celestial sphere can be observed during the course of one year.

4. Cryosat (Near Earth S-Band): an ESA mission aimed at monitoring precise changes in the thickness of the polar ice sheets and floating sea ice over a 3-year period. The observations made during the lifetime of the mission aim at determining whether or not the Earth’s ice masses are actually thinning due to a changing climate. The orbit
of the spacecraft has a high inclination of 92 degrees which takes it almost to the poles. Furthermore it is not Sun-synchronous and will drift through all angles to the Sun in 8 months.

5. GOCE (Near Earth S-Band): The objective of the GOCE mission is to provide global and regional models of the Earth’s gravity field and of the geoid, its reference equipotential surface, with high spatial resolution and accuracy. Such models will be used in a wide range of research and application areas, including global ocean circulation, physics of the interior of the Earth and levelling systems based on the Global Positioning System (GPS).

6. GALILEO: Galileo will be Europe’s own global navigation satellite system, providing a highly accurate, guaranteed global positioning service under civilian control. The fully deployed Galileo system consists of 30 satellites (27 operational + 3 active spares), positioned in three circular Medium Earth Orbit (MEO) planes at 23 222 km altitude above the Earth, and at an inclination of the orbital planes of 56 degrees with reference to the equatorial plane.

7. Commercial GEO: this scenario includes the generic geostationary commercial satellites.

B.2 Overview of the scenarios specifics

<table>
<thead>
<tr>
<th>Telecommand uplink</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Modulation scheme</td>
<td>PCM/PSK(sine)/PM</td>
</tr>
<tr>
<td>Data rate</td>
<td>7.8125 bps, 15.625 bps, 250 bps, 1000 bps, 2000 bps, 4000 bps</td>
</tr>
<tr>
<td>Subcarrier frequency</td>
<td>16 KHz coherent with data rate</td>
</tr>
<tr>
<td>Ranging</td>
<td>Simultaneous Ranging and Telecommand is possible</td>
</tr>
<tr>
<td>Maximum doppler shift</td>
<td>±200 KHz</td>
</tr>
<tr>
<td>SNR min</td>
<td>10 dB</td>
</tr>
</tbody>
</table>

Table B.1: Deep Space BepiColombo Telecomand uplink

<table>
<thead>
<tr>
<th>Telemetry downlink</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Modulation scheme</td>
<td>PCM/PSK(square)/PM, SP-L/PM, OQPSK</td>
</tr>
<tr>
<td>Data rate PCM/PSK/PM</td>
<td>up to 60 Kbps</td>
</tr>
<tr>
<td>Data rate SP-L/PM</td>
<td>up to 1400 Kbps</td>
</tr>
<tr>
<td>Data rate OQPSK</td>
<td>524288, 1048576 sps</td>
</tr>
<tr>
<td>Subcarrier frequency</td>
<td>8.192 kHz 262.144 kHz</td>
</tr>
<tr>
<td>Modulation index $\beta$</td>
<td>$&lt; 1.25$ rad peak</td>
</tr>
<tr>
<td>Ranging</td>
<td>Simultaneous Ranging and TC is possible except in OQPSK</td>
</tr>
<tr>
<td>Maximum doppler shift</td>
<td>±400 KHz</td>
</tr>
<tr>
<td>SNR min</td>
<td>4.5 - 4.9 dB</td>
</tr>
</tbody>
</table>

Table B.2: Deep Space BepiColombo Telemetry downlink
### Telecommand uplink

<table>
<thead>
<tr>
<th>Modulation scheme</th>
<th>PCM/PSK(sine)/PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data rate PCM/PSK(sine)/PM</td>
<td>125, 4000 bps (GAIA) 2000 bps (LISA)</td>
</tr>
<tr>
<td>Subcarrier frequency</td>
<td>16 KHz coherent with data rate</td>
</tr>
<tr>
<td>Modulation index $\beta$</td>
<td>0.2 to 1.4 rad peak</td>
</tr>
<tr>
<td>Ranging</td>
<td>Simultaneous Ranging and TC is possible</td>
</tr>
<tr>
<td>Maximum doppler shift</td>
<td>$\pm 220 \text{KHz}$</td>
</tr>
<tr>
<td>SNR min</td>
<td>11 dB</td>
</tr>
</tbody>
</table>

**Table B.3:** Lagrangian X-Band (Lisa and Gaia)

### Telemetry downlink

<table>
<thead>
<tr>
<th>Modulation scheme</th>
<th>PCM/PSK(sine)/PM, SP-L/PM, GMSK (BTs = 0.25) (GAIA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subcarrier Frequency</td>
<td>32000 Hz (LISA) 146789 Hz (GAIA)</td>
</tr>
<tr>
<td>Data-rate PCM/PSK(sine)/PM</td>
<td>143, 2293, 4587, 18349, 36697 sps (GAIA) 1000 sps (LISA)</td>
</tr>
<tr>
<td>Data-rate (SP-L/PM)</td>
<td>285714, 571429 sps (GAIA) 60241, 120482 sps (LISA)</td>
</tr>
<tr>
<td>Data-rate (GMSK)</td>
<td>10 Msps</td>
</tr>
<tr>
<td>Modulation index $\beta$</td>
<td>1.25 rad peak</td>
</tr>
<tr>
<td>Ranging</td>
<td>Simultaneous Ranging and Telemetry</td>
</tr>
<tr>
<td>Maximum doppler shift</td>
<td>$\pm 400 \text{KHz}$</td>
</tr>
<tr>
<td>SNR min</td>
<td>3.7 dB</td>
</tr>
</tbody>
</table>

**Table B.4:** Lagrangian X-Band (Lisa and Gaia)

### Telecommand uplink

<table>
<thead>
<tr>
<th>Modulation scheme</th>
<th>PCM/PSK(sine)/PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subcarrier Frequency</td>
<td>16 KHz coherent with data rate</td>
</tr>
<tr>
<td>Data-rate PCM/PSK(sine)/PM</td>
<td>2000 bps (Cryosat) 4000 bps (GOCE)</td>
</tr>
<tr>
<td>Modulation index $\beta$</td>
<td>0.2 &lt; 1.4 rad peak</td>
</tr>
<tr>
<td>Maximum doppler shift</td>
<td>$\pm 70 \text{KHz}$</td>
</tr>
<tr>
<td>Ranging</td>
<td>Simultaneous Ranging and TC is possible except in OQPSK</td>
</tr>
<tr>
<td>SNR min</td>
<td>10 dB</td>
</tr>
</tbody>
</table>

**Table B.5:** Near Earth S-Band (Cryosat and GOCE)

### Telemetry downlink

<table>
<thead>
<tr>
<th>Modulation scheme</th>
<th>PCM/PSK(sine)/PM (Cryosat), SP-L/PM and OQPSK(GOCE)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subcarrier Frequency</td>
<td>64 KHz</td>
</tr>
<tr>
<td>Data rate PCM/PSK/PM</td>
<td>8 Kbps</td>
</tr>
<tr>
<td>Data rate SP-L/PM:</td>
<td>55565 bps</td>
</tr>
<tr>
<td>Data rate OQPSK:</td>
<td>1 Msp</td>
</tr>
<tr>
<td>Modulation index $\beta$</td>
<td>&lt; 1.5 rad peak</td>
</tr>
<tr>
<td>Ranging</td>
<td>Simultaneous Ranging and TC is possible except in OQPSK</td>
</tr>
<tr>
<td>Maximum doppler shift</td>
<td>$\pm 140 \text{KHz}$</td>
</tr>
<tr>
<td>SNR min</td>
<td>5.9 dB</td>
</tr>
</tbody>
</table>

**Table B.6:** Near Earth S-Band (Cryosat and GOCE)
### Telecommand uplink

<table>
<thead>
<tr>
<th>Modulation scheme</th>
<th>PCM/PSK(sine)/PM and PCM/UQPSK (SS mode)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data-rate PCM/PSK(sine)/PM</td>
<td>2000 bps on 8 KHz subcarrier</td>
</tr>
<tr>
<td>Data-rate PCM/UQPSK</td>
<td>2000 bps</td>
</tr>
<tr>
<td>Modulation index $\beta$</td>
<td>$&lt; 1.2$ rad peak</td>
</tr>
<tr>
<td>Maximum doppler shift</td>
<td>$\pm 50KHz$</td>
</tr>
<tr>
<td>Ranging</td>
<td>Simultaneous Ranging and TC is possible except in OQPSK</td>
</tr>
<tr>
<td>SNR min</td>
<td>22 dB</td>
</tr>
</tbody>
</table>

**Table B.7: Galileo**

### Telemetry downlink

<table>
<thead>
<tr>
<th>Modulation scheme</th>
<th>PCM/NRZ/PSK(sine)/PM and PCM/OQPSK (SS Mode)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subcarrier Frequency</td>
<td>64 KHz</td>
</tr>
<tr>
<td>Data-rate PCM/PSK(sine)/PM</td>
<td>50 ksps</td>
</tr>
<tr>
<td>Data-rate PCM/UQPSK</td>
<td>50 ksps</td>
</tr>
<tr>
<td>Modulation index $\beta$</td>
<td>$&lt; 1.2$ rad peak</td>
</tr>
<tr>
<td>Maximum doppler shift</td>
<td>$\pm 100KHz$</td>
</tr>
<tr>
<td>Ranging</td>
<td>Simultaneous Ranging and TC is possible except in OQPSK</td>
</tr>
<tr>
<td>SNR min</td>
<td>6 dB</td>
</tr>
</tbody>
</table>

**Table B.8: Galileo**

### Telecommand uplink

<table>
<thead>
<tr>
<th>Modulation scheme</th>
<th>PCM/PSK(sine)/FM and PCM/UQPSK (SS mode)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data-rate PCM/PSK(FM)</td>
<td>2250 bps, 1000 bps, 2000 bps, 4000 bps</td>
</tr>
<tr>
<td>Data-rate PCM/UQPSK</td>
<td>4000/2n with $n = 0,1,9$</td>
</tr>
<tr>
<td>Maximum doppler shift</td>
<td>in FM mode: +/- 400 kHz in SS Mode: +/- 20 KHz</td>
</tr>
<tr>
<td>Ranging</td>
<td>Simultaneous Ranging and TC is possible</td>
</tr>
<tr>
<td>SNR min</td>
<td>11 dB</td>
</tr>
</tbody>
</table>

**Table B.9: Commercial Geostationary**

### Telemetry downlink

<table>
<thead>
<tr>
<th>Modulation scheme</th>
<th>PCM/NRZ/PSK(sine)/PM and PCM/OQPSK (SS mode)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subcarrier Frequency</td>
<td>64 KHz</td>
</tr>
<tr>
<td>Data-rate PCM/PSK/PM</td>
<td>64sps 20 ksps</td>
</tr>
<tr>
<td>Data-rate PCM/OQPSK (SS mode)</td>
<td>$2^n$ with $n = 6..14$</td>
</tr>
<tr>
<td>Modulation index $\beta$</td>
<td>$&lt; 1.25$ rad peak</td>
</tr>
<tr>
<td>Ranging</td>
<td>Simultaneous Ranging and TC is possible except in OQPSK</td>
</tr>
<tr>
<td>Maximum doppler shift</td>
<td>$\pm 200KHz$</td>
</tr>
<tr>
<td>SNR min</td>
<td>3.8 dB</td>
</tr>
</tbody>
</table>

**Table B.10: Commercial Geostationary**
Appendix C

TT&C: Telemetry, Tracking, and Commanding

Telemetry (TM)  The Telemetry mode of operation is used for the satellites when they have to send data to the ground station, most of this data is scientific data from the satellite sensors, however there are also data about the satellite state. Usually this signal has high symbol-rates and it is received at low SNR.

Telecommand (TC)  The telecommand mode is used for the ground stations to send commands to the satellite, this commands can be any kind of order to the satellite (for example, change the transmission parameters, or mover the photovoltaic panels...). These number of operations are usually small, this signal is very reliable and the satellite has to send always confirmation of the received telecommand.

The Ranging and Doppler tracking system use a satellite tracking system capable of providing information on the range (distance) rate between a spacecraft and the ground station. It uses an active transponder on-board the satellite for the retransmission to the ground of an Earth-to-Space link signal: ranging signal generation and measurement are performed in the Earth station. For more detailed information please refer to [2].

The ranging and Doppler system consist on

- Earth to Space link function, employing ground communication, process control, ranging signal generation and Earth-to-Space communications
- transponder function
- Space to Earth link, function employing Space to Earth communication Doppler measurement ranging replica generation ranging correlation process control and ground communication
- Link control function
- Data acquisition function.
Doppler performance  The accuracy of the Doppler measurement is affected by several error sources, systematic errors (bias) due to the stability of the different oscillators and noise induced errors (jitter).

Ranging performance  The accuracy of the ranging measurement is affected by several error sources, systematic errors (bias) due to station and transponder group delay and noise induced errors (jitter) such as quantization error, measurement accuracy.
Modulation index estimation

In [17] it is proposed different estimation techniques for the modulation index $\beta$. However no simulation results about the algorithms performance are published. In this work we have briefly tested some of the proposed algorithms in order to check if this algorithms could be considered for a more detailed study. In [17] is first pursued a maximum likelihood (ML) estimation approach to estimating modulation index along with appropriate approximations of the nonlinearities that result to allow for practical implementations at low and high SNR scenarios. It is also considered ad hoc estimation algorithm for where the carrier synchronization has not yet been established, in addition it is described how this scheme may be applied when the modulation type (MPSK), symbol timing, SNR, and data rate are also unknown.

**Estimation in absence of knowledge of the modulation (MPSK), symbol-rate, symbol timing and SNR** the ML methods introduced in [17] need information about the signal modulation, the symbol-rate, symbol-timing and SNR to make the estimation, however here is introduced a blind method that does not need this information. Thus and as hoc estimator of the modulation index $\beta = \cot^{-1} \sqrt{P_c/P_d}$ is given by [17]

$$
\hat{\beta} = \cot^{-1} \left[ \frac{2 \left( \int_0^{KT^*} y_c(t)dt \right)^2 + \left( \int_0^{KT^*} y_s(t)dt \right)^2}{KT^* \int_0^{KT^*} \left[ (y_c(t) - y_c(t - T^*))^2 + (y_s(t) - y_s(t - T^*))^2 \right]^2} \right]
$$

(D.1)

Where $y_c(t)$ and $y_s(t)$ are the in phase and quadrature component of the signal, $T^*$ is the maximal period of symbol.

**Noncoherent estimation in absence of carrier frequency knowledge** is a modification of the previous method for signals where the carrier frequency is unknown. For instances a received signal affected by Doppler. From [17] the ad hoc estimator is

$$
\cos \beta = \sqrt{\frac{\int_0^{KT} \tilde{y}(t)\tilde{y}^*(t - T)dt}{\int_0^{KT} |\tilde{y}(t)|^2 dt}}
$$

(D.2)
Figure D.1: Performance of the algorithm of estimation in absence of knowledge of the modulation (MPSK), symbol-rate, symbol timing and SNR.
Figure D.2: Performance of the algorithm noncoherent estimation in absence of carrier frequency knowledge
Bibliography

[2] Ranging and Doppler tracking ECSS-E-50-02A.


[34] W. Gappmair & O. Koudelka M. Flohberg, W. Kogler. Symbol rate estimation with


[37] W. Wei & J. M. Mendel. A new maximum-likelihood method for modulation classifi-

[38] E. Sepedin & G. B. Giannakis P. Ciblat, P. Loubaton. Asymptotic analysis of blind

[39] Davidson & Goldschneider & Gazzanti & Pitton. Feature-based modulation classifi-


[41] Achilleas Anastasopoulos & Andreas Polydoros Prokopios Pangiotou. Likelihood ratio

space communications.* Jet Propulsion Laboratory, 2001.

[43] Yawpo Yang & Samir S. Soliman. A suboptimal algorithm for modulation classifica-


TN1: Review of the application scenarios.*

TN3: Autonomous Reciever Architecture and Algorithm Definition and Design.*


2420, 2005.


