Optimization of demodulation performance of the GPS and GALILEO navigation messages

Axel Javier Garcia Peña

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par *l'Institut National Polytechnique de Toulouse*
Discipline ou spécialité : *Signal, Image, Acoustique*

Présentée et soutenue par *Axel Javier Garcia Peña*
Le *7/10/2010*

Titre :

Optimisation de la Performance de Démodulation des Messages de Navigation GPS et GALILEO

Optimization of Demodulation Performance of the GPS and GALILEO Navigation Messages

---

JURY
Professeur Marco Luise
Professeur Emmanuel Boutillon
Professeur Bernd Eissfeller
Professeur Marie-Laure Boucheret
Docteur Christophe Macabiau
Lionel Ries Chef du Département DCT/RF/SR

Ecole doctorale : *Mathématiques, Informatique et Télécommunications de Toulouse*
Unité de recherche : *IRIT/ENAC/TeSA*
Directeur(s) de Thèse : *Marie-Laure Boucheret et Christophe Macabiau*
Rapporteurs : *Emmanuel Boutillon et Bernd Eissfeller*

Abstract

The demodulation performance achieved by any of the existing GNSS signals, such as GPS L1 C/A, GPS L2C or GPS L5, is satisfactory in open environments where the available C/N_0 is quite high. However, in indoor and in urban environments, several characteristics degrade the demodulation performance. In particular, in these environments, the C/N_0 level of the received signal is often very low. Also, when the receiver is in motion, the C/N_0 suffers additional fast variations due to changing diffraction conditions which can further affect the GNSS messages demodulation. Therefore, since the mass-market applications being conceived nowadays are aimed at indoor and urban environments, it is necessary to study and to search alternative demodulation/decoding methods which improve the GNSS messages demodulation performance in these environments.

It is also needed to consider new GNSS signals, such as GPS L1C and GALILEO E1, which were developed recently. These signals aim at providing satellite navigation positioning service in any kind of environment, giving special attention to indoor and urban environments. Therefore, this dissertation also analyses the demodulation performances of the new GNSS signals as they are defined in the current public documents. Moreover, new GALILEO E1 message structures are proposed and analysed in order to optimize the demodulation performance as well as the quantity of broadcasted information.

Therefore, the main goal of this dissertation is to analyse and to improve the demodulation performance of the GPS L1 C/A, GPS L2C, GPS L5, GPS L1C and GALILEO E1 signals, specifically in indoor and urban environments, and to propose new navigation message structures for GALILEO E1. A detailed structure of the sections of this dissertation is given next.

First, the subject of this thesis is introduced, original contributions are highlighted, and the outline of the report is presented.

Second, this dissertation begins by a description of the current structure of the different analysed GNSS signals, paying special attention to the navigation message structure and the implemented channel code. Also, the different channel code decoding techniques used in subsequent sections of this dissertation are fully described.

In the third section, two types of transmission channel models are presented. These models represent the correlator outputs as used for carrier tracking and demodulation in the two environments, assuming ideal code tracking. On one hand, the open environments, and even some indoor environments, are modelled by an AWGN channel. Therefore, this report describes the AWGN channel mathematical model and presents its effects on the signal carrier phase tracking process, since the phase tracking performance altogether with the code tracking performance condition the demodulation performance. On the other hand, the urban environments and some indoor environments are modelled as a mobile channel. In this case, there exists several possible mathematical mobile channel models, and thus one of them is selected and this choice is justified. Moreover, the problems affecting the signal carrier phase tracking process are also presented, altogether with two techniques employed during the simulations in order to track the signal carrier phase. These techniques are the PLL and the channel estimation. Additionally, for both transmission channels, the structure implemented during the simulations in order to reproduce the channel characteristics is presented.

In the fourth and fifth sections, efforts to improve the demodulation performance of the existing GNSS signals are presented.

In the fourth section, a tentative to make a binary prediction of a part of the GPS L1 C/A navigation message is presented. Note that the binary prediction introduced is a prediction of the bits forming the navigation message, not a prediction of the physical magnitude of different message fields. More specifically, the prediction of the broadcasted satellite ephemeris is tried using the GPS L1 C/A almanacs data, a long term orbital prediction program provided by TAS-F, and some signal processing methods. These signal processing methods can be separated in two types. The first type searches the ephemeris data prediction by using the past ephemeris data values, or, in other words, the history of the ephemeris data. The methods tried are the ephemeris data spectral estimation, the PRONY method, and a neural network. The second type of methods tries to exploit the correlation between the different GPS satellite orbits. More precisely, these methods try to exploit the fact that some GPS satellites share the same orbit with their position delayed in time.

Then, in the fifth section, improvements to the GPS L2C and GPS L5 navigation message demodulation performance are brought by using their channel codes in a non-traditional way. The proposed method consists in combining the navigation message inner and outer channel codes in order to correct more received words. The technique is such that the receiver accepts as the transmitted word the most probable word provided by the Viterbi decoding, or inner channel code, which meets the outer channel code verification. In fact, this technique follows the same principle as the list Viterbi decoding method although the proposed algorithm is completely different. Also, the technique solves some additional problems due to the GPS navigation messages. The proposed algorithm is described, its advantages and drawbacks presented, some possible modifications are given and its performance is compared with the performance obtained by traditional GPS L2C and GPS L5 decoding methods. Additionally, another method used to improve this performance is presented. This method consists in using the ephemeris data probabilities in order to improve the traditional Viterbi decoding.

In the sixth section, the GPS L1C and GALILEO E1 Open Service demodulation performance is analysed in different environments. More specifically, this section presents a brief study of the structure of both signals to determine the received C/N_0 in an AWGN channel. Then, the demodulation performance of these signals is analysed through simulations in different environments. The environments analysed are the open, indoor and urban environments modelled by the AWGN channel and the mobile channel. Therefore, this section presents the demodulation performance obtained when the GNSS signals are transmitted through an AWGN channel assuming perfect tracking, an AWGN channel with thermal noise affecting the PLL tracking, an AWGN channel with thermal noise and dynamic stress error affecting the PLL tracking, and a mobile channel with a GNSS receiver travelling at 5km/h, 30 km/h and 50 km/h. In this last case, the carrier phase tracking is achieved either by using a PLL or by applying a channel estimation technique.

In the seventh section, efforts to improve the different GALILEO E1 signal performances are presented. Specifically, this dissertation presents a new navigation message structure which improves the demodulation performance and increases the signal information transmission rate.

The new proposed navigation message structure consists in adopting a message structure similar to the GPS L1C message structure but also in including a signalling technique known as Code Shift Keying, or CSK, which increases the information transmission rate. In fact, the CSK technique consists in shifting the PRN code of each transmitted symbol in order to map for each code shift a fixed quantity of bits. This results into an increased quantity of bits transmitted during a symbol length, from one to the number of mapped bits. The CSK implementation into a navigation signal analysis is detailed next.

First, the drawbacks introduced by the CSK implementation are analysed, and mainly the problems due to the fact that the data channel PRN code is no longer synchronized with the pilot channel PRN code. This induces that the data channel can no longer be used to acquire or to track the signal in the signal parts where the CSK is implemented. Therefore, the new acquisition performance is analysed. The optimal source mapping code shifts which reduce the acquisition false alarm arte when the data channel is employed are also presented. Initial conclusions concerning the impact of CSK on the tracking performance is also presented.

Second, this dissertation searches for the demodulation performance of different CSK source mapping configurations. The CSK source mapping configuration refers to the bits and their distribution into packets to be transmitted by the same CSK symbol. Therefore, in order to analyse the different CSK source mapping options, the following points are studied. First, the interest in using the CSK polarity in order to encode an extra bit is analysed. Second, the study determines whether it is best to transmit bits belonging to several packets inside the same CSK symbol, or if it is best to transmit only bits belonging to the same packet. Third, the theoretical BER curves for different numbers of bits forming a CSK are shown. Fourth and last, figures depicting the BER and WER of different CSK source mapping configurations are presented. Additionally, since some of the packets transmitted by the CSK symbols implement channel codes with soft inputs, the theoretical expressions of the likelihood ratios of the bits coded by a CSK symbol have been calculated and verified.

Finally, we conclude this study.

Résumé

La performance de démodulation des signaux GNSS existants, tels que GPS L1 C/A, GPS L2C ou GPS L5, est satisfaisante en environnements ouverts où le C/N_0 disponible est assez élevé. Cependant, à l'intérieur des bâtiments ou en milieu urbain, plusieurs phénomènes dégradent la performance de démodulation. En particulier, dans ces environnements, le niveau de C/N_0 du signal reçu est souvent très bas. De plus, lorsque le récepteur est en mouvement, le C/N_0 est affecté de rapides variations dues aux conditions de diffraction changeantes qui peuvent affecter sérieusement la démodulation des messages GNSS. En conséquence, puisque les applications du marché de masse en conception actuellement sont appelées à être déployées à l'intérieur des bâtiments et en milieu urbain, il est nécessaire d'étudier et de chercher des méthodes de démodulation/décodage qui améliorent la performance de démodulation des messages GNSS dans ces environnements.

Il est aussi nécessaire de considérer les nouveaux signaux GNSS, tels que GPS L1C et GALILEO E1, qui furent développés récemment. Ces signaux doivent fournir un service de positionnement par satellite dans tout type d'environnement, et spécifiquement à l'intérieur des bâtiments et en environnement urbain. Ainsi, cette thèse analyse aussi les performances de démodulation des nouveaux signaux GNSS tels que définis dans les documents publics actuels. De plus, de nouvelles structures de message GALILEO E1 sont proposées et analysées dans le but d'optimiser la performance de démodulation ainsi que la quantité d'information diffusée.

En conséquence, le but principal de cette thèse est d'analyser et améliorer la performance de démodulation des signaux GPS L1 C/A, GPS L2C, GPS L5, GPS L1C et GALILEO E1, spécifiquement à l'intérieur des bâtiments et en milieu urbain, et de proposer de nouvelles structures de messages de navigation pour GALILEO E1. La structure détaillée des chapitres de cette thèse est donnée ci-après.

En premier lieu, le sujet de cette thèse est introduit, ses contributions originales sont mises en avant, et le plan du rapport est présenté.

Dans le deuxième chapitre, la thèse décrit la structure actuelle des différents signaux GNSS analysés, en se concentrant spécifiquement sur la structure du message de navigation et le codage canal implanté. Aussi, les différentes techniques de décodage du code canal utilisées dans les sections suivantes de la thèse sont entièrement décrites.

Dans le troisième chapitre, deux types de modèles de canal de propagation sont présentés. Ces modèles représentent les sorties de corrélateurs utilisées pour la poursuite de la porteuse et la démodulation dans les deux environnements, en supposant une poursuite de code idéalement parfaite. Les environnements ouverts et les environnements indoor sont modélisés par un canal AWGN. Ce rapport décrit donc le modèle mathématique du canal AWGN et présente ses effets sur le processus de poursuite de la phase de porteuse, puisque la performance de la poursuite de la porteuse conjointement à la performance de la poursuite du code conditionnent la performance de démodulation. Puis, les environnements urbains et certains environnements indoor sont modélisés comme un canal mobile. Dans ce cas, il existe plusieurs modèles mathématiques de canal mobile, et l'un d'entre eux est sélectionné et ce choix est justifié. De plus, les problèmes affectant le processus de poursuite de la phase de la porteuse sont aussi présentés, de même que deux techniques employées durant les simulations dans le but de poursuivre la phase de la porteuse du signal. Ces techniques sont la PLL et l'estimation de canal. De plus, pour les deux canaux de propagation, la structure implantée pour les simulations dans le but de reproduire les caractéristiques du canal est présentée.

Dans les deuxième et troisième chapitres, on présente les efforts effectués pour améliorer la performance de démodulation des signaux GNSS existants.

Dans le quatrième chapitre, on présente une tentative pour effectuer une prédiction binaire d'une partie du message de navigation GPS L1 C/A. La prédiction binaire introduite est une prédiction des bits formant le message de navigation, pas une prédiction de la valeur physique des différents champs du message. Plus spécifiquement, on tente la prédiction des éphémérides diffusées en utilisant les almanachs GPS L1 C/A, grâce à un programme de prédiction à long terme fourni par TAS-F, et des méthodes de traitement du signal. Ces méthodes de traitement du signal peuvent être séparées en deux types. Le premier type recherche la prédiction des données d'éphémérides en utilisant les valeurs passées des données d'éphémérides, c'est-à-dire l'historique des données d'éphémérides. Les méthodes testées sont l'estimation spectrale des données d'éphémérides, la méthode de PRONY, et un réseau de neurones. Le deuxième type de méthodes tente d'exploiter la corrélation entre les différentes orbites des satellites GPS. Plus précisément, ces méthodes tentent d'exploiter le fait que certains satellites GPS partagent la même orbite avec une position retardée.

Puis, dans le cinquième chapitre, des améliorations à la performance de démodulation du message de navigation GPS L2C et GPS L5 sont apportées en utilisant leur code canal de manière non traditionnelle. La méthode proposée consiste à combiner les codes canal internes et externes du message de navigation dans le but de corriger davantage de mots reçus. La technique est telle que le récepteur accepte comme mot émis le mot le plus probable fourni par le décodage Viterbi, ou code canal interne, qui satisfait à la vérification du code canal externe. En fait, cette technique suit le même principe que la méthode de décodage de la liste de Viterbi bien que l'algorithme proposé soit complètement différent. Aussi, la technique résout des problèmes additionnels dus aux messages de navigation GPS. L'algorithme proposé est décrit, ses avantages et inconvénients présentés, des modifications possibles sont données et sa performance est comparée avec la performance obtenue par les méthodes de décodage traditionnelles de GPS L2C et GPS L5. De plus, une autre méthode utilisée pour améliorer cette performance est présentée. Cette méthode consiste à utiliser les probabilités des données d'éphémérides dans le but d'améliorer le décodage traditionnel de Viterbi.

Dans le sixième chapitre, la performance de démodulation des messages de GPS L1C et du signal Open Service GALILEO E1 est analysée dans différents environnements. Plus précisément, cette section présente une brève étude de la structure de ces deux signaux pour déterminer le C/N0 du signal utile reçu dans un canal AWGN. Puis, la performance de démodulation de ces signaux est analysée grâce à des simulations dans différents environnements. Les environnements analysés sont les environnements ouverts, indoor et urbain modélisés par le canal AWGN et le canal mobile. En conséquence, cette section présente la performance de démodulation obtenue lorsque les signaux GNSS sont émis à travers un canal AWGN en supposant une poursuite parfaite, un canal AWGN avec bruit thermique affectant la poursuite effectuée par la PLL, un canal AWGN avec bruit thermique et une erreur de dynamique affectant la poursuite de la PLL, et un canal mobile avec un récepteur GNSS se déplaçant à 5km/h, 30 km/h et 50 km/h. Dans ce dernier cas, la poursuite de la phase de la porteuse est effectuée soit en utilisant une PLL, soit en appliquant une estimation de canal.

Dans le septième chapitre, des efforts pour améliorer les différentes performances du signal GALILEO E1 sont présentés. Spécifiquement, cette thèse présente une nouvelle structure de message de navigation qui améliore la performance de démodulation et augmente le taux d'émission d'information par le signal.

La nouvelle structure de message de navigation proposée consiste à adopter une structure de message similaire à celle de GPS L1C mais en incluant aussi une technique appelée Code Shift Keying, ou CSK, qui augmente le taux de transmission de l'information. La technique CSK consiste à décaler le code PRN de chaque symbole émis de manière à associer à chaque décalage de code un nombre fixe de bits. Ceci résulte en une augmentation du nombre de bits transmis pendant la durée d'un symbole, de un au nombre de bits associés. L'analyse de l'implantation du CSK dans un signal de navigation est détaillée dans la partie suivante.

Tout d'abord, les inconvénients introduits par l'implantation du CSK sont analysés, et principalement les problèmes dus au fait que le code PRN de la voie données n'est plus synchronisé avec le code PRN de la voie pilote. Ceci induit le fait que la voie données ne peut plus être utilisée pour acquérir ou poursuivre le signal dans les portions du signal où le CSK est implanté. En conséquence, la nouvelle performance d'acquisition est analysée. Les décalages de code optimaux qui réduisent le taux de fausse alarme d'acquisition lorsque la voie données est employée sont aussi présentés. Des conclusions initiales concernant l'impact du CSK sur la performance de poursuite sont aussi présentées.

Ensuite, cette thèse recherche la performance de démodulation des différentes configurations des associations de source CSK. La configuration de l'association de la source CSK se réfère aux bits et à leur distribution en paquets devant être transmis par le même symbole CSK. Donc pour analyser les différentes options d'association de source CSK, les points suivants sont étudiés. Tout d'abord, l'intérêt de l'utilisation de la polarité CSK permettant d'encoder un bit supplémentaire est analysé. Deuxièmement, l'étude détermine s'il vaut mieux transmettre les bits appartenant à plusieurs paquets à l'intérieur du même symbole CSK, ou s'il vaut mieux transmettre seulement les bits appartenant au même paquet. Troisièmement, on montre les courbes théoriques de BER pour différentes valeurs de nombres de bits formant un CSK. En quatrième point, les figures montrant le BER et le WER de différentes configurations d'association de source CSK sont présentées. De plus, puisque certains paquets émis par les symboles CSK implantent des codes canal avec entrée soft, les expressions théoriques des rapports de vraisemblance des bits codés par un symbole CSK ont été calculées et vérifiées.

Finalement, on conclut cette étude.

Acknowledgements

First and foremost I would like to express my gratitude to my primary supervisors, Dr. Christophe Macabiau and Prof. Marie-Laure Boucheret, who have been a constant source of motivation and knowledge.

For her help in the beginning of my Ph.D., I would like to thank Dr. Anne-Christine Escher. I would also like to thank Dr. Oliver Julien for many helpful conversations and suggestions.

I gratefully acknowledge the financial support of CNES (Centre National d'études Spatiales) and the guidance of Lionel Ries, as well as the financial support of TAS-F (Thales Alenia Space France) and his many advisors Stéphane Corazza, Jean-Louis Damidaux and Charles Fernet.

I would also like to acknowledge the administrative support of CNS-ENAC department and TéSA laboratory.

I would also like to express my gratitude to my two dissertation reviewers, Prof. Emmanuel Boutillon and Prof. Bernd Eissfeller for accepting and reviewing my long and, sometimes unending, thesis. Moreover, I would also like to thank Prof. Marco Luise for being the president of the Ph.D. defense jury and for asking "different" questions.

I wish to thank my family, Pere, Marisa and Sergi, for their support through my education and for much needed loving moments. Thanks to you I made it until the very end although sometimes it seemed impossible.

I would also like to thank my dear friends in Toulouse, David P., Anne-Lise M., Roger B., Xavi M., Roger F., Isidre M., Nil G., Laura R., Francesc Ll., Margarita del R., Nuria T., and my dear friends in Barcelona, Francesc S., Mamen P., Albert F., Neus A., Roger F. and Nuria T., the good moments we have spent together which gave my energy to keep going.

And I cannot forget all the officemates which has shared with me the hardships of a Ph.D., Mathieu R., Benjamin C., Anaïs M., Christophe O., Na T., Paul T., Adrien C., Pier N., Daniel S., Damien S., Kevin E., Sébastien C. and Myriam F. Thank you all.

Finally, I would like to thank my love, Marta, which I did not know at the moment of my Ph.D. but I wanted her to be here anyway.

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

The motivation of this dissertation and the dissertation organization are presented in this chapter.

1.1. Motivation

The GNSS systems history begins during the World War II, when the first systems allowing the location of a receiver into large Earth surface areas were developed. Some examples of these systems were the LORAN [HECKS, 1990] and the Decca Navigator [MALLESON, 1944] which used ground-based radio navigation structures instead of satellites since this latter technology was still far from being exploitable. In fact, it was not until the sixties when the first system using satellite technology was developed; this system was called Transit and it was conceived by the U.S. Navy [PARKINSON et al, 1995]. Nevertheless, the conception of the original Global Positioning System had still to wait until 1973 when it was proposed for the American military as a defensive system [PARKINSON et al, 1995]. The GPS conception was materialized in 1978 with the launch of the first experimental Block-I GPS, and the GPS satellite constellation was completed in 17th of January of 1994 with the launch of the 24th satellite.

Years later, in 1983, the U.S. government announced that the GPS system would be opened to civilian users as soon as the system was completely deployed [PARKINSON et al, 1995]. The reason was that the commercial aircraft KAL 007 had been recently shot down by the Soviets after that the civilian airliner went into prohibited aerospace. Nevertheless, the U.S. government decided to broadcast first a degraded GPS signal and it was not until 2nd of May of 2000 when the civilian users were allowed to receive a non-degraded GPS signal [CLINTON, 2000].

The first GPS signal used for civilian users was the GPS L1 C/A signal [ARINC, 2004]; a signal which was originally conceived during the eighties in order to ease the military signal acquisition process, GPS L1 P(Y) signal. Since the first version of the GPS L1 C/A, several upgrades correcting mainly original conceptions errors and improving the positioning service accuracy have been developed. Nevertheless, all these improvements were limited by the GPS L1 C/A original structure and the main objective pursued during the first GPS conception: being able to navigate by satellite a launched missile in open air. Therefore, in 1998 the U.S. congress decided to deploy a new civilian signal in L2 frequency with the same characteristics as the GPS L1 C/A signal in order to improve the navigation, positioning and time services [GORE, 1998]. This signal was later called GPS L2C. Besides, a few months later, another announcement was made where a second GNSS civil signal would be added in the L5 frequency in order to provide the needs of critical safety-of-life applications such as civil aviation [GORE, 1999]. This signal was later called GPS L5.

The signals characteristics were first presented in 2000 for GPS L5 signal [VAN DIERENDONCK et al, 2000] [ARINC, 2005], and in 2001 for GPS L2C [FONTANA et al., 2001] [ARINC, 2004] where this last signal presented a major revolution in relation to the GPS L1 C/A signal since instead of being just a copy, it was conceived with a completely new structure. These new GPS signals, in addition to enhancing user positioning accuracy and reliability, also had as objective to improve the performance of the three fundamental processes of a GNSS receiver which allow providing the fundamental measurements; the

signal acquisition, tracking and demodulation processes. In this Ph.D. manuscript, the analyses conducted are focused on different GNSS signal demodulation performance parameters and thus the GNSS signal characteristics which are commented are mainly centered on this signal aspect.

The main improvements and signal structures designed for the new GPS L2C and GPS L5 signals from the demodulation point of view were the introduction of a dataless channel, called pilot channel, and the implementation of a new navigation message structure. More specifically, on one hand, the pilot channel allowed the implementation of more powerful PLL discriminators and removed the tracking performance dependence on the symbol duration. On the other hand, the introduction of a convolutional channel code over the navigation message allowed the correction of the received words in addition to the error detection capability provided by the initial implemented channel code. Moreover, the channel code responsible for detecting the erroneous words was improved from the simple extended Hamming (32, 26) hamming code to the powerful CRC-24Q channel code.

Nevertheless, although these signals provided a great improvement from the demodulation, tracking and acquisition performance point of view with respect to the original GPS L1 C/A signal, they were still not sufficiently adapted to environments with non line-of-sigh satellites. Therefore, due to the continuous growth of demand of positioning service in indoor and in urban environments, a new GPS civilian signal was developed and its specifications published in 2006 [ARINC, 2006]. This signal is called GPS L1C and in addition to providing better general performance than the GPS L2C and GPS L5 signals, its navigation message structure has been adapted to allow the signal demodulation in indoor and urban environments.

More specifically, from the demodulation point of view, the increase of the GPS L1C signal word length responsible for carrying the ephemeris sets allowed the signal designers to implement a much more powerful channel code, finally a LDPC channel code. Moreover, the introduction of an interleaver allowed the signal processing adaptation to channels which introduce burst of errors, mainly urban environments. Also through the new introduced LDPC channel code, the signal designers could provide more power to the pilot channel so that the tracking performance could be improved.

All the GPS signals upgrades and improvements have provided a very accurate global positioning service which have made its commercial deployment a major success. This great success altogether with the desire of independence and the strategic advantage as a military system that GPS represents have made that several countries have decided to develop their own GNSS; some examples are the GALILEO for the European Union, the COMPASS for China, the QZSS for Japan and the IRNSS for the India. Moreover, the Russian GNSS system known as GLONASS was developed at the same time as the GPS system. Nevertheless, among these future systems, only the GALILEO system and more specifically the GALILEO E1 OS signal is further inspected and commented in this dissertation.

The GALILEO system is the European GNSS system and the GALILEO E1 OS signal [ESA, 2008] is the signal in charge of providing free positioning service with the highest possible accuracy and availability. This signal is designed to achieve the maximum degree of interoperability with GPS L1C signal whereas keeping its own independence at the same time. Moreover, in comparison with GPS L1C, GALILEO E1 OS also provides additional services such as integrity and Sol (Safety of Life) services.

The GALILEO E1 OS signal development is nearly completed but the GALILEO satellite constellation is far from been deployed. In fact, in 2014 it is planned to have 14 operational satellites in orbit in addition to the 4 IOV satellites [EC, 2010]. The same occurs for the GPS

signals of new generation where it is expected to have 24 satellites broadcasting the GPS L2C signal in 2016, 24 satellites broadcasting the GPS L5 signal in 2018 and 24 satellites broadcasting the GPS L1C signal in 2021 [ROPER, 2010]. This means that in the next years to come, the GNSS positioning service should still be provided by GPS L1 C/A signal, progressively by GPS L2C signal and also by GLONASS since 21 satellites are currently operational [RSA, 2010]. Therefore, in addition to conducting demodulation performance studies on the modern GPS and GALILEO signals, some studies should be aimed at improving the GPS L1 C/A signal demodulation performance since this signal will still be used for several years. These two types of studies are conducted in this dissertation. However, none analysis is made of GLONASS although its role in the next years promises to be very interesting.

In this dissertation, the demodulation performance of the current and modern GPS and GALILEO signals is analyzed. The objective behind the analysis of the current GPS signals, namely GPS L1 C/A, GPS L2C and GPS L5, is to improve the demodulation performance obtained by previous studies [BASTIDE, 2004] [DAFESH et al., 2007] by means of new proposed demodulation or decoding techniques, such as the binary ephemeris prediction or a new decoding strategy. In fact, the ideal objective consists in obtaining the desired demodulation performance at the lowest C/N_0 value which is supported by the PLL before it loses its lock. Or, in other words, it consists in lowering the C/N_0 demodulation threshold to the level of the C/N_0 carrier phase tracking threshold. Besides, this objective is pursued in low C/N_0 scenarios, for example indoor environments, since the GPS L1 C/A, GPS L2C and GPS L5 signals have shown to obtain satisfying demodulation performance in open environments.

Moreover, in this dissertation, the GALILEO E1 OS and GPS L1C signals demodulation performance is also analyzed in order to assess the performance of these signals in any kind of environment, particularly urban environments which are modeled by mobile channels. And since both signals target the same mass market, it is also interesting to know which signal provides a better demodulation performance. Besides, the different conducted demodulation performance analysis is also used as a method through which the weak points of the GPS signals and GALILEO E1 OS signal as well as their strong points can be identified in order to help the conception of the future GALILEO open signal version.

Finally, in addition to providing the typical positioning service, the new GNSS signals will offer other types of services, such as integrity or commercial services. Besides, some other additional services can be found useful in the near future and thus the GNSS signals should be prepared to support and exploit them. Therefore, since the inclusion of additional services inside a GNSS signal means the increase of the amount of information to be broadcasted by a satellite, the GNSS signal should be able to support higher information transmission rates than the current GNSS signals can.

One possible solution inspected during this dissertation is the introduction of a signaling method specially designed to increase the information transmission rate of spread spectrum signals. This method is called Code Shift Keying (CSK) and increases the information transmission rate by transforming each shift of the signal spreading spectrum code into an information symbol, where each symbol represents a given amount of bits. The principal drawback of this signaling technique is its necessity of an additional signal carrying the spreading spectrum code without any implemented shift in order to have a reference for the CSK signal. However, the own structure of the modern GPS and GALILEO signals already contain this reference signal, the pilot channel; therefore, this technique shows great potential for its application in these signals.

1.2. Dissertation Organization

The dissertation is structured following the GNSS signals evolution in time and the reasons are given next. As the dissertation title indicates, the general objective of this dissertation is to optimize the demodulation of the GNSS navigation messages. The concept of optimization can be divided into two main groups depending on the targeted GNSS signal. The GNSS signals belonging to the first group are the already existing GNSS signals, GPS L1 C/A, GPS L2C and GPS L5. In this case, since the demodulation performance of the navigation messages of these GNSS signals is well known, their optimization process consists in searching different methods to optimize their demodulation performance, the decrease of the C/N_0 required to obtain a desired BER. The GNSS navigation messages belonging to this group are analyzed first in this dissertation, starting with GPS L1 C/A and continuing with GPS L2C and GPS L5, since their examination will provide significant information which could be used to propose improved navigation message structures for the future GNSS signals, another of the dissertation objectives. The GNSS signals belonging to the second group are the future GNSS signals, GPS L1C and GALILEO E1 OS, and are thus analyzed after the first group GNSS signals in this dissertation. In this case, the concept of optimization is different from the first group since the demodulation performance of the navigation message of GALILEO E1 OS signal has not been previously analyzed. Therefore, the demodulation performance of the navigation message of GALILEO E1 OS signal and GPS L1C are first analyzed and second compared to each other in order to identify the strong and weak points of each signal and of each navigation message. Finally, the optimization process for this second group consists in presenting a new navigation message structure which improves the current demodulation performance of GALILEO E1 OS signal from the analysis of the demodulation performance of the navigation messages of the first and second GNSS signals. This proposition is evidently found last in the dissertation in order to take advantage from the analysis of the demodulation performance of all the GNSS signals.

A more detailed structure of this Ph.D. manuscript is given below.

First of all, in chapter 2, the definitions of all the GNSS signals employed at some time during this dissertation are given. This chapter takes special attention to the signals navigation message structure, their implemented channel codes and the coding and decoding methods associated to the channel codes.

In chapter 3, the AWGN and mobile channel mathematical models are presented in order to be used during the remaining chapters of the dissertation. In this chapter, the AWGN mathematical model is presented from the literature and the demodulation performance losses due to the carrier phase tracking errors are presented. Moreover, the mobile channel mathematical model is developed and justified.

In chapter 4, the binary prediction of the GPS L1 C/A signal ephemeris is inspected. More specifically, the required bit probabilities of the ephemeris binary prediction, the different methods employed to conduct the prediction and the justification of the unsatisfactory results are presented.

In chapter 5, two new decoding methods for the GPS L2C and GPS L5 signals are presented. The first method is a modification of the traditional Viterbi algorithm and the second method is a new decoding strategy for the signals inner and outer channel codes. Moreover, in addition to calculating the new methods demodulation performance, we calculate the signals demodulation performance for the traditional Viterbi algorithm when the carrier phase tracking process is noised by thermal noise and dynamic stress error.

In chapter 6, the GPS L1C and GALILEO E1 OS signals demodulation performance is calculated and compared one to the other. First, this comparison is made through a theoretical study where the main signal differences conditioning the demodulation performance are identified. And second, this comparison is made through simulations using an AWGN channel model and a mobile channel model.

Finally, in chapter 7, the Code Shift Keying (CSK) signaling method is defined, the bits likelihood ratio expressions associated to this method are developed and verified, the optimal mapping source is found, the demodulation performance obtained by the GALILEO E1 OS signal when the CSK method is implemented over the data channel is calculated, and the CSK onto the signal acquisition and tracking performance is analyzed. Moreover, a new GALILEO E1 OS signal structure is presented.

Chapter 2. GPS and GALILEO Signals

This chapter gives a detailed description of the different GPS and GALILEO signals studied during this thesis. A good understanding of the signal structure and, more important, a good knowledge of the navigation message structure and the implemented channel code/s of each signal is necessary to follow the ideas, assumptions, developments and calculations presented during this dissertation.

In this chapter, first a general description of the message content of all the inspected navigation signals is presented. Second, the specific signal structure definition of each signal is given. Moreover, in addition to the definition, the channel code/s implemented for each navigation message as well as their associated decoding techniques are described.

The signals described in this chapter are the GPS L1 C/A, the GPS L2C, the GPS L5, GPS L1C and GALILEO E1 OS signals.

2.1. GPS and GALILEO signals navigation message content

The different GPS and GALILEO navigation messages have a common general content; therefore a general summary for all the inspected signals is presented in this section. The GNSS navigation message is absolutely necessary to obtain the position of any GPS or GALILEO receiver as it can be seen in annex A.1. The message has different types of information which have different degrees of importance. Some information such as the satellites ephemeris data and satellites clock data is completely fundamental to obtain the final user position and others such as the ionosphere parameters are a correction factor of the user pseudo-range measurement, or in other words, it refines the estimated user position accuracy.

Since the number of types of information is large and each GPS or GALILEO signal distributes it differently on its navigation message, a list of the most important contents is provided in this subsection [SPILKER and ASHBYe, 1996]:

- Ephemeris data: The ephemeris data predicts the satellite position with high accuracy inside a given interval of time which can vary for each GPS or GALILEO signal. This information is fundamental to obtain the final user position.
- Time and clock correction parameters: In order to estimate the user position, the user receiver needs to determine the satellite clock bias and drift. This information is also fundamental to obtain the final user position.
- Service parameters: navigation data set identifiers, satellite identifiers, and signal health indicators. This information is fundamental to obtain the final user position too.
- Ionospheric model parameters, Broadcast Group delay (BDG), Navigation Message Correction Table (NMCT): these parameters are used to make pseudo-range measurement corrections.
- Coordinated Universal Time (UTC) parameters: allows the correction of the GST time with respect to the UTC time.
- Almanac data: this information is used to predict the position of all the satellites in the constellation with a reduced accuracy but with a longer period of validity in comparison with the ephemeris data. The almanac data is a reduced-precision subset of the clock and ephemeris parameters of the active satellites in orbit. The almanacs

are broadcasted by every satellite in order to enable the user to improve the acquisition time of the satellite by reducing the search window.

2.2. GPS L1 C/A

In this subsection the structure of the GPS L1 C/A signal is presented paying special attention to the navigation message structure and to the channel code implemented over the navigation message. The document describing the GPS L1 C/A signal is [ARINC, 2004].

2.2.1. Signal Structure

The transmitted GPS L1 C/A signal can be modeled as the multiplication of three components: a carrier with frequency L1, a spreading spectrum code waveform that is a PRN code waveform, and the navigation message waveform, D(t). Figure 2-1 illustrates one possible modulator scheme.

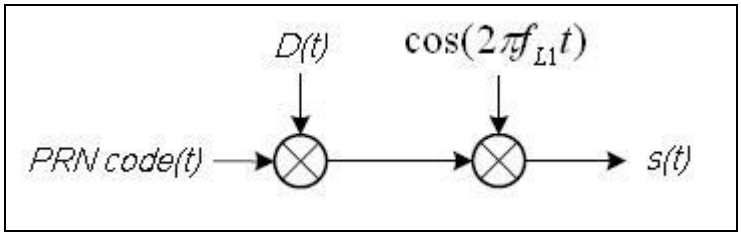

Figure 2-1: GPS L1 C/A signal modulation scheme

The navigation data message BPSK modulates the carrier with a rate of 50 symb/sec, meaning a symbol duration equal to 20ms. Therefore, each symbol contains 20 periods of the PRN code. At the receiver level, the data information is obtained after demodulating this navigation message and after verifying that it has been well received.

A more detailed description of the structure of this signal is given in annex D.1.

2.2.2. Navigation Message Structure

The navigation data message structure of the GPS L1 C/A signal has a basic information unit of 1500 bits. This basic information unit is called frame and is divided into 5 smaller units known as subframes. Each subframe has thus a length of 300 bits grouped into words of 30 bits. Therefore, 10 words of 30 bits form a subframe and 5 subframes form a frame.

Each subframe carries a different type of information and the information contained inside each subframe can be repeated in the next frame. More specifically, the normal satellite operation is that inside a period of 2 hours [ARINC, 2004], the subframes 1, 2 and 3 transmitted at time t contain the same kind of information as the subframes 1, 2 and 3 transmitted at time $t+T$, where T is the duration of a frame. However, the information of subframes 4 and 5 varies from one frame to the next one. In fact, the type of information transmitted in the subframes 4 and 5 is repeated after the broadcasting of 25 additional frames. Therefore, 25 frames form a superframe where the information carried by the subframes 4 to 5 changes from one frame to the next one and the information carried by the subframes 1 to 3 is the same when all the frames belong to the same 2h period. Note that this period is defined for the normal satellite operation, which means that the period duration is usually 2h but not always.

The channel code of the GPS L1 C/A signal is applied over each word, where the first 24 bits are information bits and the remaining 6 ones are parity bits. A more detailed description is presented in annex E.1.

The studies developed during this thesis are mainly focused on subframes 1 to 3 and also on subframe 5 but in a minor degree. Therefore the content of these subframes is presented next. Nevertheless, subframe 4 is superficially described in order to complete the general overview of the message.

Subframe 1 carries the information concerning the clock parameters and corrections, the satellite URA (User Range Accuracy) and health and group propagation delay corrections. A scheme of its content is illustrated below [ARINC, 2004].

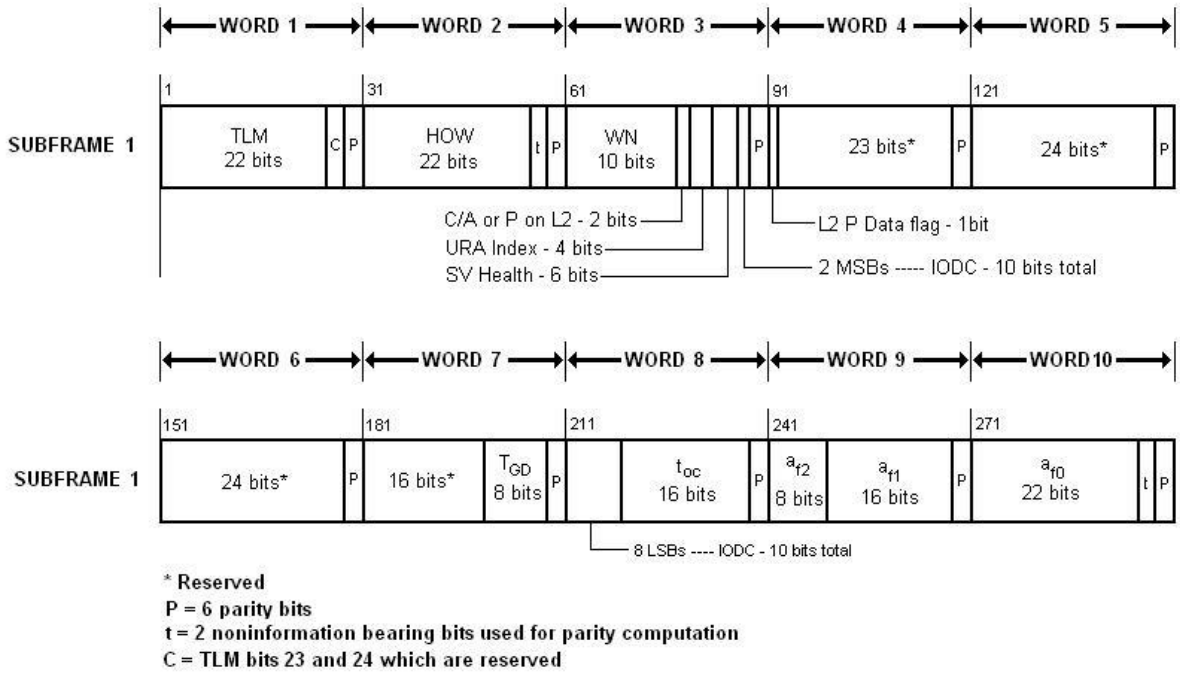

Figure 2-2: Subframe 1 of the GPS L1 C/A signal [ARINC, 2004]

Subframes 2 and 3 contain the ephemeris data of the transmitting satellites. For GPS L1 C/A signal, the interval of time inside which the ephemeris data have a minimum defined accuracy is usually set to 4h [ARINC, 2004], although the navigation message transmits a new ephemeris set usually each 2 hours [ARINC, 2004]. Additionally, note that each set of ephemeris data can be used outside its 4 hours interval but with a degraded accuracy.

The subframes 2 and 3 structures are depicted below [ARINC, 2004].

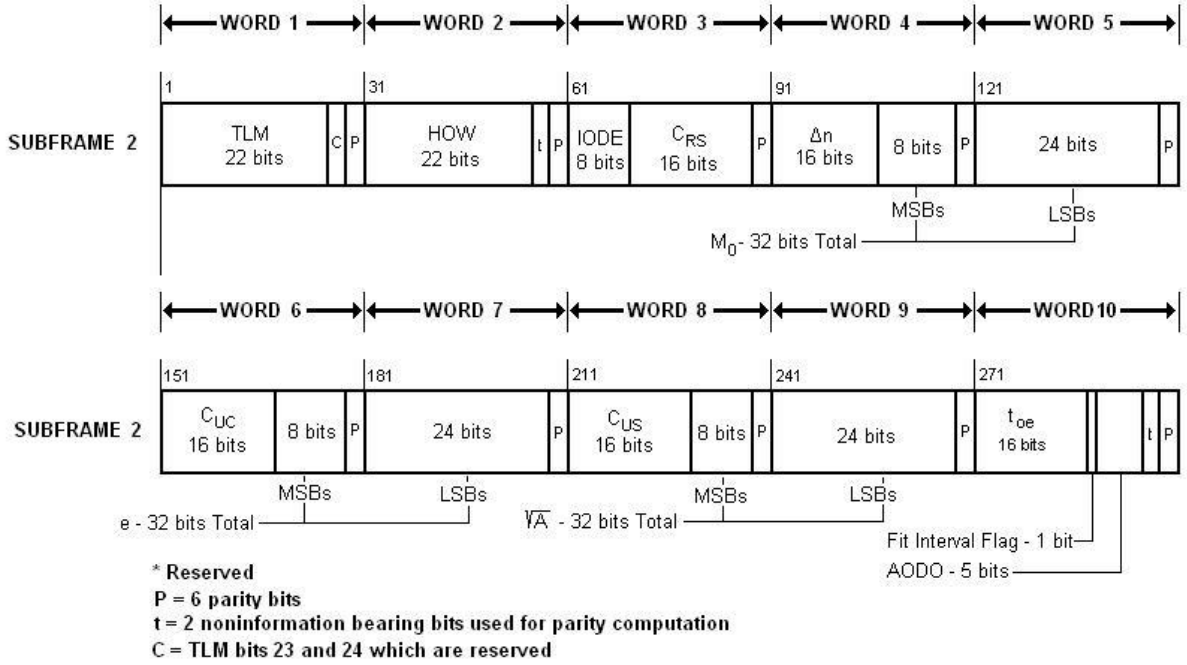

Figure 2-3: Subframe 2 of the GPS L1 C/A signal [ARINC, 2004]

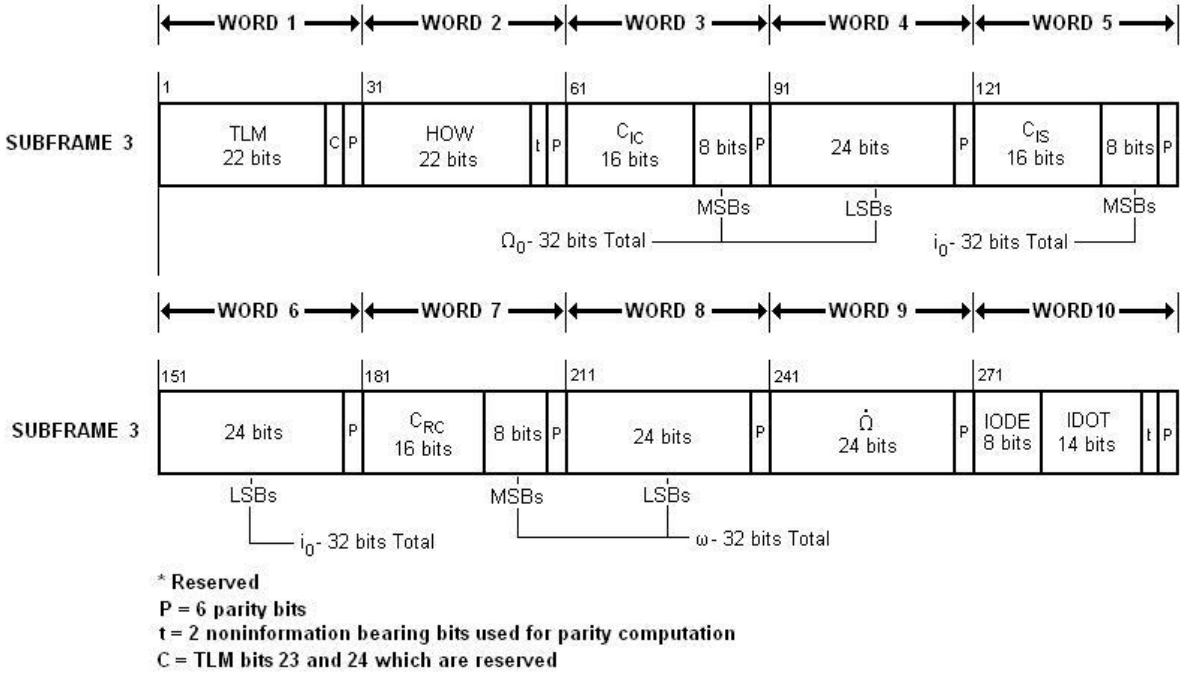

Figure 2-4: Subframe 3 of the GPS L1 C/A signal [ARINC, 2004]

Subframe 4 contains a lot of different types of information. The type of information depends on the position of the frame inside the superframe. Therefore, in order to identify the different frames inside a superframe, the frames are called pages and an identifier related to the page position is associated to each page all along this dissertation. For example, the page 13 is the 13^{th} transmitted frame inside a superframe and carries the pseudo-range measurement errors corrections. This subframe and its content are not important from the dissertation objectives point of view, and thus none subframe 4 scheme is presented. Nevertheless, in order to obtain

a general overview, the reader must know that subframe 4 also carries information related to satellites health and configurations, ionospheric and UTC data, NMCT and some satellites almanacs (page 2 to 5 and page 7 to 10) [ARINC, 2004].

Subframe 5 contains the almanac data of all the GPS L1 C/A constellation satellites. The almanac data is transmitted from page 1 to page 24, where each page carries the almanac data of 1 specific satellite. The almanac data has an interval of time of 6 days inside which the satellite position accuracy remains inside the desired values [ARINC, 2004]. Nevertheless a new set of almanac data is broadcasted everyday as it has been observed from the GPS L1 C/A message analysis.

Finally, page 25 of subframe 5 contains a summary of the satellite health and configurations, and the A-S (Anti-Spoofing) status.

Figure 2-5 illustrates subframe 5 pages 1 to 24 and subframe 4 pages 2 to 5 and 7 to 10 [ARINC, 2004].

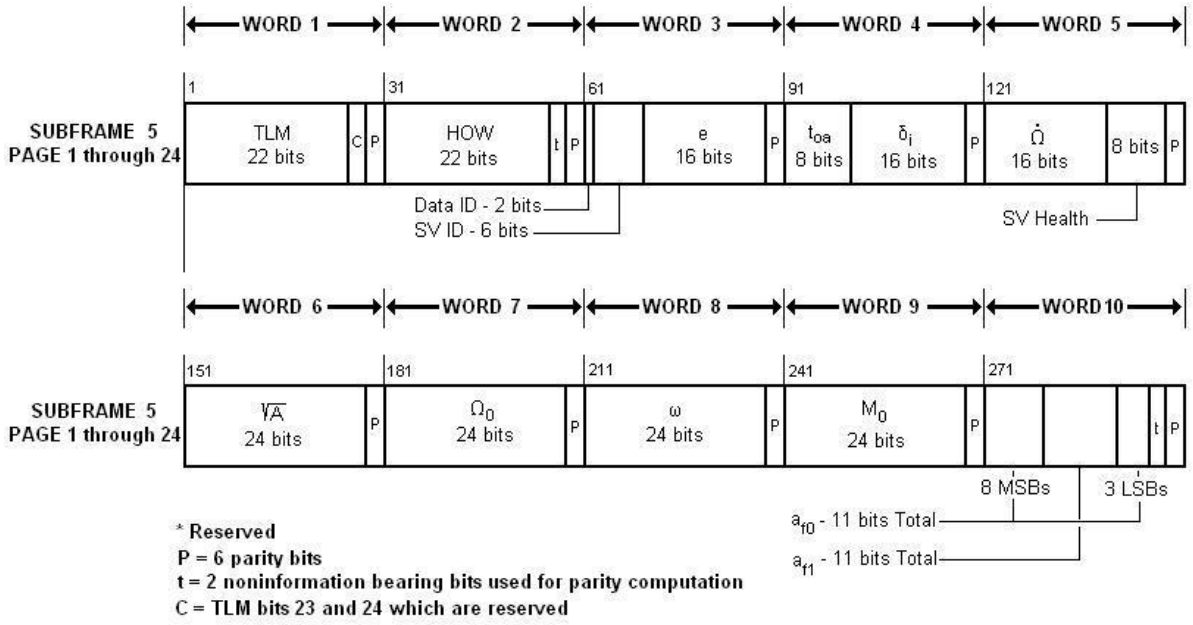

Figure 2-5: Page Almanac - Subframe 4 and 5 of the GPS L1 C/A signal [ARINC, 2004]

2.2.3. Channel Code

The channel code implemented on the GPS L1 C/A navigation message is an extended Hamming code (32,26), which is the result of extending a (31,26) Hamming code through the introduction of a parity bit.

The encoding and decoding processes of this code are presented in annex E.1.

2.3. GPS L2C

In this subsection the structure of the GPS L2C signal is presented paying special attention to the navigation message structure and to the channel code implemented over the navigation message. The document describing the GPS L2C signal is [ARINC, 2004].

2.3.1. Signal Structure

The transmitted GPS L2C signal can be modeled as the multiplication of 4 components: a carrier with frequency L2, two spreading spectrum code waveforms that are PRN codes waveforms called L2 CL code and L2 CM code, and the navigation message waveform, D(t) or $D_c(t)$.

The navigation message waveform (D(t) or Dc(t)) is multiplied by the L2 CM code waveform. The result is chip-by-chip time multiplexed to the other code waveform, the L2 CL code. Finally, the resulting signal is multiplied by the carrier L2 obtaining the GPS L2C signal. The next figure presents one possible GPS L2C modulator scheme.

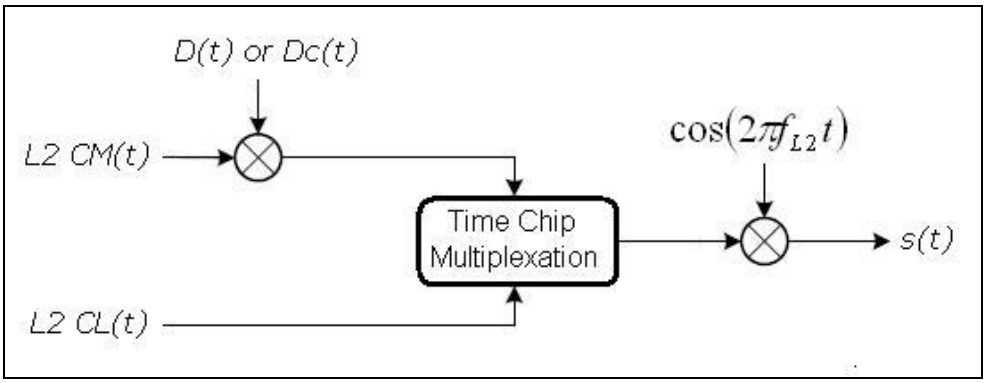

Figure 2-6: GPS L2C signal modulation scheme

The navigation data message content varies depending on the satellite configuration. The navigation data is materialized with a BPSK modulation, the resulting waveform is multiplied by the L2 CM code waveform, and the result modulates the L2 carrier with a 50 symb/s rate. This means that a symbol has a duration equal to 20ms and that each data symbol contains 1 and only 1 period of the PRN L2 CM code at the receiver level. The data information is obtained after demodulating this navigation message and after verifying that the message has been well decoded.

A more detailed description of the structure of this signal is given in annex D.2.

2.3.2. Navigation Message Structure

The navigation data message of the signal GPS L2C has 3 different structures depending on the transmitting satellite and its configuration. In fact, these 3 different messages are planned to allow a gradual mutation of the receivers from the GPS L1 C/A signal processing to the new GPS L2C processing. This means that the 3 message configurations are not radically different among them but rather they keep a similarity in order to smooth the change.

2.3.2.1. GPS L2C mode NAV without FEC data message structure

This structure can be broadcasted only by satellites block IIR-M and only during the initial period of deployment of this new block, before the initial operational capability of L2C signal [ARINC, 2004].

The GPS L2C mode NAV without FEC data message structure is exactly the same as the GPS L1 C/A navigation message structure and thus it is not presented again in this subsection. Section 2.2.2 has given a detailed description of this navigation message.

2.3.2.2. GPS L2C mode NAV with FEC data message structure

This structure can be broadcasted only by satellites block IIR-M and only during the initial period of deployment of this new block, before the initial operational capability of L2C signal [ARINC, 2004].

The GPS L2C mode NAV with FEC data message structure is quite similar to the GPS L1 C/A navigation message structure. In fact, the GPS L1 C/A message is used as input to a convolutional code (171, 133, r = ½), and the output is BPSK modulated and 2-modulo added to the L2 CM code. Therefore, the navigation message structure is the same as the structure of section 2.2.2 but adding a new channel code which provides the FEC (forward error correction) at the receiver level. In this case, the GPS L1 C/A message is generated at a rate of 25 symb/sec and the output of the channel code has a rate of 50 symb/sec. The convolutional code is described in the section 2.3.3.

2.3.2.3. GPS L2C mode CNAV data message structure

This structure can be broadcasted by SVs block IIF and by SVs block IIR-M after the initial operational capability of L2C signal [ARINC, 2004].

The GPS L2C mode CNAV data message structure is similar to the message structure of section 2.3.2.2 but, in this case, the message used as input to the convolutional code is completely new compared to the 2 previous structures. This new structure is also generated at a rate of 25 symb/s and the output of the FEC channel code has a rate of 50 symb/s. The convolutional code is described in the section 2.3.3.

The new navigation data message structure, before being the convolutional code input, contains the same type of information as the GPS L1 C/A message: ephemeris, clock parameters, clock corrections, UTC data, ionospheric corrections, group delay, reduced almanac, etc. However, the bit resolution and the bit precision of these different parameters are different from the GPS L1 C/A message and thus its frame and subframe structure are also modified. In fact, the CNAV data is also organized in subframes of 300 bits, but no frame or superframe is defined: the only structure requirements concern time restrictions specifying the maximum time-delay between transmissions of subframes carrying the same determined type of information: time outs [ARINC, 2004].

The GPS L2C mode CNAV data message subframe has a size of 300 bits out of which 276 are information bits and 24 are parity bits as it is illustrated in Figure 2-7. These 24 parity bits are coded from a cyclic code, CRC-24Q, which protects all the previous 276 information bits. This cyclic code is described in annex E.2. The duration of a subframe is 12 seconds.

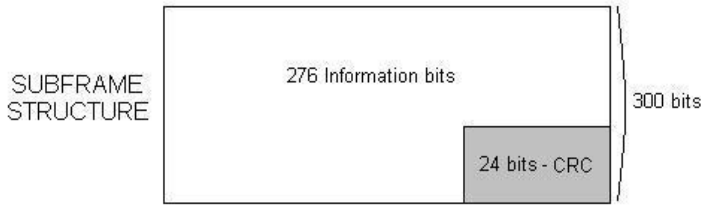

Figure 2-7: Subframe Structure of GPS L2C mode CNAV data [ARINC, 2004]

2.3.3. Channel Code

In this section, only the channel codes implemented in the structures described in section 2.3.2.2 and 2.3.2.3 navigation messages are described since the channel code implemented in the structure described in section 2.3.2.1 navigation message is the same as the GPS L1 C/A channel code.

Two channel codes in serial configuration are applied on both navigation messages, GPS L2C mode NAV with FEC data and GPS L2C mode CNAV data. These two channel codes are called the inner channel code and the outer channel code. The inner channel code is a convolutional code (171, 133, r = ½) and it is in charge of the FEC.

The outer channel code is either an extended Hamming code (32,26) for the GPS L2C mode NAV with FEC data signal or a cyclic code called CRC-24Q for the GPS L2C mode CNAV data signal. The outer channel code function is to detect any error introduced by the transmission channel which has not been corrected by the inner code. A serial implementation of two channel codes consists in first using the data information (276 information bits) as the input of the outer channel code, second in using the output of the outer channel code (276 information bits + 24 parity bits) as the input of the inner channel code, and third in transmitting through the channel the output of the inner channel code (600 coded bits). Figure 2-8 illustrates a scheme of a serial configuration of two channel codes.

Figure 2-8: Serial implementation scheme of the inner and outer channel codes.

2.3.3.1. Inner Channel Code

The inner channel code is the same for both navigation messages, GPS L2C mode NAV with FEC data and GPS L2C mode CNAV data. The inner channel code is a convolutional code with a code rate equal to ½ and with polynomial generators equal to G1 = 171 and G2 = 133 expressed in octal format. Its minimum free distance is 10 (d_{free} = 10) and the constraint length is equal to 7 (L = 7). The main characteristic of a convolutional code is that the code has memory: the new generated coded bits depend on the new input bit plus some of the previous input bits. Another important characteristic is that the input source is not bounded in size; the source can be a continuous stream of bits.

The coding and decoding process of the inner channel code are well detailed in annex E.3.

2.3.3.2. Outer Channel Code

Depending on the GPS L2C navigation message type, the outer channel code differs. The GPS L2C mode NAV data message has as outer channel code the extended Hamming code (32,26) implemented on the GPS L1 C/A message. All its characteristics can be found in annex E.1. The GPS L2C mode CNAV data message has as outer channel code a cyclic code known as CRC-24Q. Its characteristics are described in annex E.2.

2.4. GPS L5

In this subsection the structure of the GPS L5 signal is presented paying special attention to the navigation message structure and to the channel code implemented over the navigation message. The document describing the GPS L5 signal is [ARINC, 2005].

2.4.1. Signal Structure

The transmitted GPS L5 signal can be modeled as linear combination of 7 components: two carriers in phase-quadrature with frequency L5, two spreading spectrum codes waveforms that are PRN codes called I5-code code and QI5-code, two different synchronization sequences waveforms, and the navigation message waveform, D(t).

Each of the L5 carrier in phase-quadrature components is modulated by a different materialized train of bits. One is a composite bit train generated by the modulo-2 addition of the I5-code ranging code, a 10-bit Neuman-Hofman code synchronization sequence, and the downlink system data (referred to as L5 CNAV (civil navigation) data). The other one is also a bit train generated by the modulo-2 addition of the Q5-code ranging code and a 20-bit Neuman-Hofman code synchronization sequence. On possible GPS L5 signal demodulator scheme is shown below.

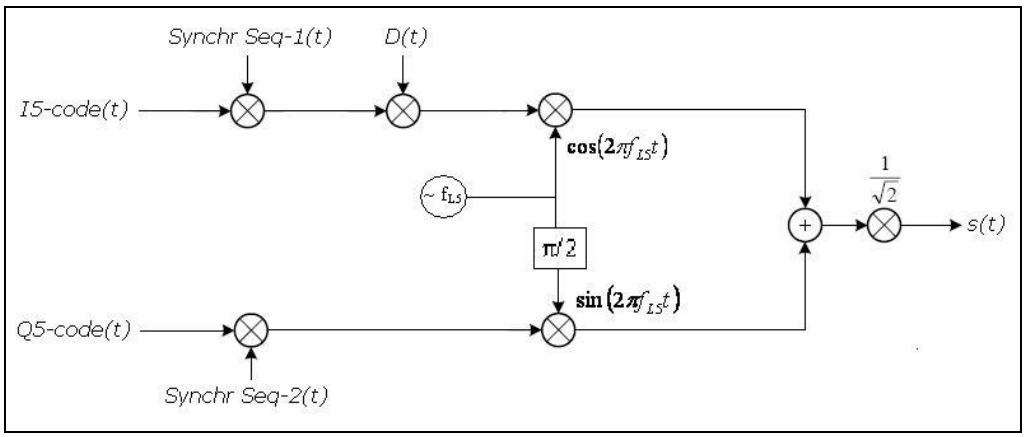

Figure 2-9: GPS L5 signal modulation scheme

The navigation data message is generated at a rate of 50 bit/s and encoded with a convolutional code of rate equal to ½ resulting into a final stream of bits at 100 bit/sec. This stream is BPSK modulated obtaining a duration of symbol equal to 10ms, and each symbol contains 1 period of synchronization sequence and 10 periods of the I5-code. The data information is obtained after demodulating this navigation message and after verifying that message has been well decoded.

A more detailed description of the structure of this signal is given in annex D.3.

2.4.2. Navigation Message Structure

The navigation message of GPS L5 signal is in general very similar to the 3rd version of the GPS L2C signal, the GPS L2C mode CNAV data message (see 2.3.2.3). The subframe size and the channel code implemented over the subframes are the same and the only slight difference resides in their content. However, this dissertation is not interested in the GPS L5 message content for the same reason that it was not interested in the GPS L2C one: the

message was not available when the signal performance was analyzed. Therefore, the only remark worth mentioning is that the message source is a continuous stream of 300 bit subframes encoded with a convolutional code of rate ½ and, out of these 300 bits, 276 are information bits and 24 are parity bits. Figure 2-7 illustrates the subframe structure.

2.4.3. Channel Code

The channel code of the GPS L5 message is the same as that of GPS L2C mode CNAV data message: two codes applied in serial implementation as it has been explained in section 2.3.3. The two channel codes are called the outer channel code and the inner channel code. The outer channel code is a cyclic code called CRC-24Q which generates 24 parity bits from the 276 information bits. And the set of both types of bits creates the code word of 300 bits. The inner channel code is a convolutional code (171, 133, r = ½) which uses as input the code words of the outer channel code and whose output bits are modulated and transmitted into the channel. For a more detailed description of the codes and their coding and decoding processes see annexes E.2 and E.3.

2.5. GPS L1C

In this subsection the structure of the GPS L1C signal is presented paying special attention to the navigation message structure and to the channel code implemented over the navigation message. The document describing the GPS L1C signal is [ARINC, 2006].

2.5.1. Signal Structure

The transmitted GPS L1C signal can be modeled as the linear combination of 7 components: a carrier with frequency L1, two spreading spectrum codes waveforms that are PRN codes called $L1C_P$ and $L1C_D$, two subcarrier waveforms called $SC_P(t)$ and $SC_D(t)$, an overlay code waveform called $L1C_O$ and the navigation message waveform, D(t).

The GPS L1C signal is separated into two channels, the pilot and the data channel. The data channel contains the PRN code waveform $L1C_D$ which is multiplied by the navigation message waveform. This channel is also multiplied by the $SC_D(t)$ subcarrier which Binary Offset Carrier (BOC) (1, 1) modulates the previous multiplication result. The pilot channel contains the PRN code waveform $L1C_P$ which is multiplied by the overlay code waveform L1 C_O. This channel is also multiplied by the $SC_P(t)$ subcarrier which Time-Multiplexed BOC (TMBOC) modulates the previous multiplication result. This TMBOC modulation technique is a combination of the BOC (1, 1) and the BOC (6, 1) modulations. Finally, both channels modulate the L1carrier frequency in-phase or in phase-quadrature (this option is specified by the navigation message).

A possible GPS L1C modulator scheme is shown below.

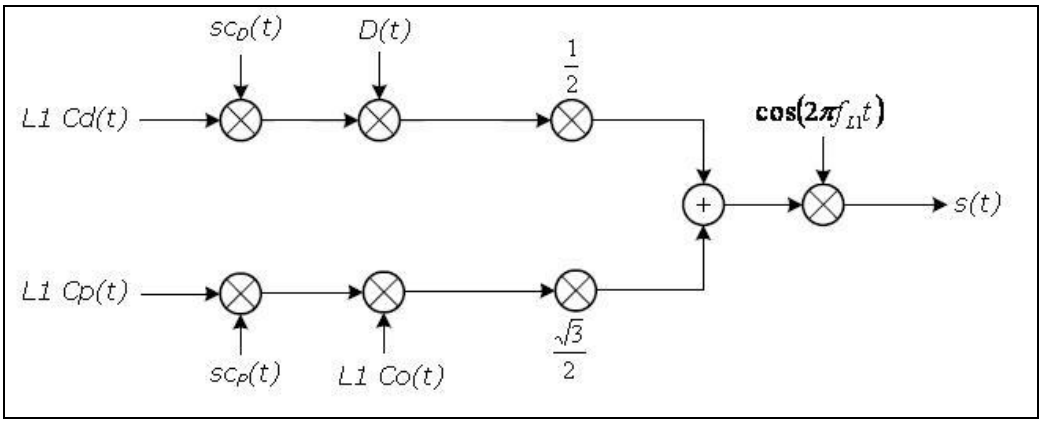

Figure 2-10: GPS L1C signal modulation scheme with in phase channels

Where:

- $sc_D(t)$ is the subcarrier responsible of the BOC(1,1) modulation
- $sc_P(t)$ is the subcarrier responsible of the TMBOC modulation resulting of the combination BOC(1,1) and BOC (6,1).

The information data is encoded with three different channel codes obtaining a navigation data frame of 1800 bits modulated with a BOC (1,1) and transmitted at a rate of 100 symb/sec. This means a symbol duration equal to the duration of a $L1C_D$ code period: 10ms. Finally, the frame duration is equal to the overlay code $L1C_O$ period duration which is transmitted on the pilot channel. Therefore, it is quite easy to accomplish the synchro-frame process due to the overlay code. The data information is obtained after demodulating this navigation message and after verifying that message has been well decoded.

A more detailed description of the structure of this signal is given in annex D.4.

2.5.2. Navigation Message Structure

The structure of the GPS L1C signal navigation message is completely different from any of the previous GPS signal navigation message structures. The GPS L1C navigation message consists in a frame constituted of 3 different subframes. Each subframe carries a different kind of information and is encoded with a different channel code. The size of the frame is 1800 bits transmitted at a rate of 100 bit/sec which means a frame duration of 18 seconds.

However, the type of information carried by each subframe is always the same for consecutive subframes 1 and for consecutive subframes 2, but varies for consecutive subframes 3. Moreover, the distribution of the different subframes 3 information follows a predetermined order defining a superframe. Therefore, the frames inside the superframe are called pages and they are identified by their position inside the superframe. Until now there are only 7 different types of subframes 3 which have been defined in [ARINC, 2006].

The GPS L1C frame is exactly constituted of 9 information bits belonging to subframe 1, plus 600 bits belonging to subframe 2 out of which 576 are information bits and 24 are parity bits, plus 274 bits belonging to subframe 3 out of which 250 are information bits and 24 are parity bits. The channel codes applied over the subframes are defined next.

The 24 parity bits belonging to either subframe 2 or to subframe 3 are encoded by the CRC-24Q channel code. This code has already been defined in annex E.2. Subframe 1 is encoded with a variation of a BCH code (51,8) resulting into 52 code bits. Subframe 2 is encoded by a

LDPC code of rate ½ resulting into 1200 code bits. Subframe 3 is encoded by a different LDPC code with a rate of ½ as well, resulting into 548 code bits. Finally, subframes 2 and 3 are interleaved together and are transmitted after subframe 1 coded bits. The description of the interleaver is given in annex D.4.

A GPS L1C navigation message structure scheme is shown below [ARINC, 2006].

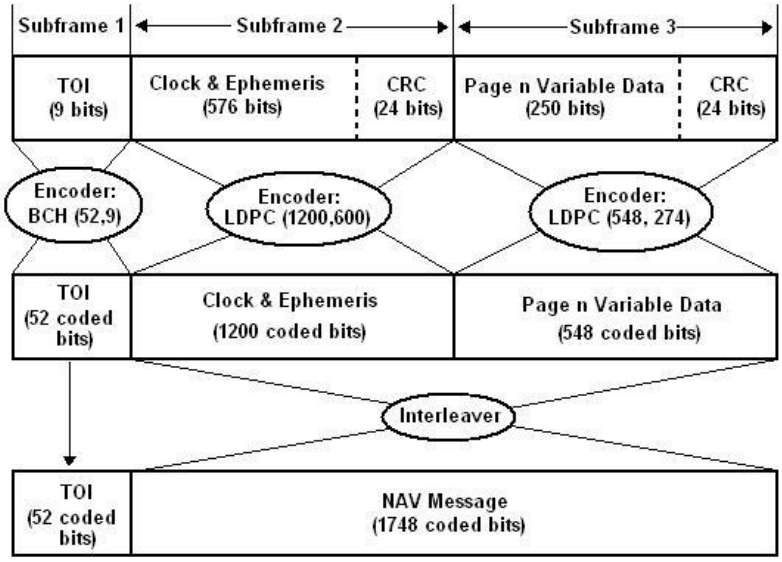

Figure 2-11: GPS L1C data message structure [ARINC, 2006]

2.5.2.1. Subframe 1

The content of subframe 1 is the TOI (Time of Interval). The TOI count uses a 9-bit data that represents the satellite time at the start of the next 18-second frame.

2.5.2.2. Subframe 2

The subframe 2 content is the ephemeris and clock correction data. Nominally, subframe 2 data is invariant for the nominal transmission interval of 15 minutes, although the ephemeris parameters describe the orbit of the transmitting satellite during a curve fit interval of 3 hours. The first 15 minutes of the curve fit interval shall thus coincide with the nominal transmission interval.

2.5.2.3. Subframe 3

At present, there are 7 different types of subframe 3 contents which could be extended in the future. Page 1 contains the UTC and ionospheric correction data. Page 2 contains the GPS/Global Navigation Satellite System (GNSS) Time Offset (GGTO) parameters and Earth Orientation Parameters (EOP). Page 3 transmits the reduced almanacs whereas page 4 contains the midi almanacs. Page 5 contains the Differential Correction (DC) parameters. These parameters provide users with sets of correction terms which apply to the clock and ephemeris data transmitted by the other satellites. Page 6 contains text message and, finally, page 7 carries signal component phase relationships of each satellite which indicates whether the pilot and the data channel are in-phase or in phase-quadrature.

2.5.3. Channel Code

The channel code implemented in the GPS L1C navigation message varies depending on the subframe of application. Nevertheless, the subframes can be divided into two different groups depending on the number of implemented channel codes.

Subframe 1 has only one channel code applied. This code is a variation/extension of a typical BCH (51,8) code resulting into a (52,9) code and, since this subframe is not studied during this thesis, the coding, the decoding and the code characteristics are only referenced [BOSE and RAY-CHADAUDHURI, 1960] [ARINC, 2006].

Subframes 2 and 3 have a serial channel code implementation whose objective is the same as the one pursued on the GPS L2C and GPS L5 navigation messages: the inner channel code serves to correct the errors introduced by the channel (FEC) and the outer channel code discards the words provided by the inner channel code which are not error free. The inner channel code varies depending on the subframe of application although both are LDPC codes with a rate equal to ½. The outer channel code is equal for both subframes and it is the same channel code as the one used on the GPS L2C and GPS L5 navigation messages: CRC-24Q.

2.5.3.1. Inner Channel Code

The inner channel code of subframes 2 and 3 is a LDPC (Low Density Parity Check) code with a code rate equal to ½. The LDPC codes were developed by Robert G. Gallager in 1963 [GALLAGER, 1963] and were later rediscovered by David J.C. MacKay and Radford M. Neal [MACKAY and NEAL, 1995].

A more detailed definition of these codes is given in annex E.5.

2.5.3.2. Outer Channel Code

The outer channel code of both subframes 2 and 3 is the same. This code is the CRC-24Q code explained and described for the GPS L2C and GPS L5 navigation message and thus will be not commented any further. The detailed description is given in annex E.2.

2.6. GALILEO E1 OS

In this subsection the structure of the GALILEO E1 OS signal is presented paying special attention to the navigation message structure and to the channel code implemented over the navigation message. Only the OS (Open Service) part of the navigation message is commented since the minimum information required by a user to obtain its final position is carried by this part and since this dissertation is only interested in these types of users. Moreover, this part of the navigation message is free and provides a service equivalent to the service brought by GPS L1C signal. Therefore, in this section, only the GALILEO E1 OS part of the navigation message is described. The document describing the GALILEO E1 OS signal is [ESA, 2008].

2.6.1. Signal Structure

The transmitted GALILEO E1 OS signal can be modeled as the linear combination of 6 components: a carrier with frequency L1, two spreading spectrum codes waveforms that are

PRN codes called $C_{E1\text{-}C}$ and $C_{E1\text{-}B}$, two subcarrier waveforms called $sc_{E1\text{-}B,a}(t)$ and $sc_{E1\text{-}B,b}(t)$ and the navigation message waveform, D(t).

The GALILEO E1 OS signal is separated into two channels, the pilot and the data channel. The data channel contains the PRN code waveform $C_{E1\text{-}B}$ which is multiplied by the navigation message waveform. This channel is also multiplied by a linear combination of the subcarriers $sc_{E1\text{-}B,a}(t)$ and $sc_{E1\text{-}B,b}(t)$. The $sc_{E1\text{-}B,a}(t)$ is equivalent to BOC (1,1) modulate the channel whereas the $sc_{E1\text{-}B,b}(t)$ is equivalent to BOC (6,1) modulate, and its linear combination is equivalent to Composite Binary Offset Carrier (CBOC) modulate the channel. The pilot channel contains the PRN code waveform $C_{E1\text{-}C}$ which is multiplied by a linear combination of the subcarriers $sc_{E1\text{-}B,a}(t)$ and $sc_{E1\text{-}B,b}(t)$. However, the pilot CBOC modulation is different from the data channel CBOC modulation. Finally, both channels modulate in-phase the L1carrier frequency.

A possible GALILEO E1 OS signal modulator scheme is shown below.

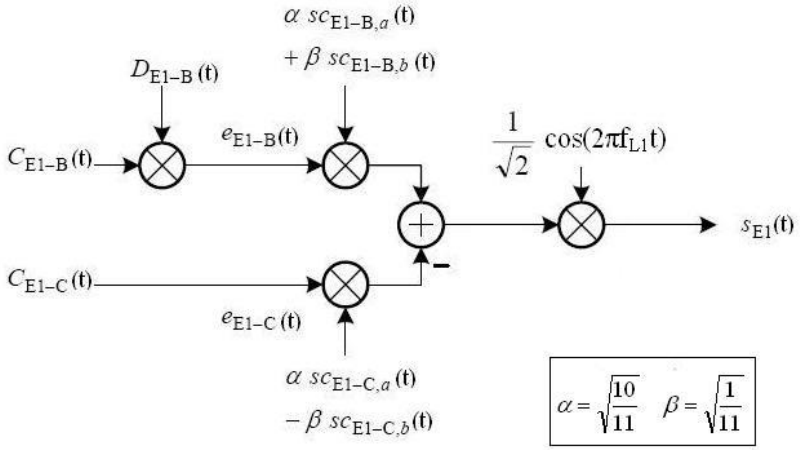

Figure 2-12: GALILEO E1 Modulation Scheme

The navigation data message is generated at a rate of 125 bits/s and encoded with a convolutional code of rate ½ resulting into a final stream of coded bits at a rate of 250 bits/s. Each symbol contains 1 period of the $C_{E1\text{-}B}$ code at the receiver level. The data information is obtained after demodulating this navigation message and after verifying that the message has been well decoded. This message contains unencrypted integrity messages and encrypted commercial data. In fact, the navigation message of E1 signal is called I/NAV message (Integrity Navigation) and supports Safety of Life service (SoL), Galileo system integrity (CS) and Open Service (OS). This dissertation is only interested in the OS part and thus it is the only part of the message described in the following subsections.

A more detailed description of the structure of this signal is given in annex D.5.

2.6.2. Navigation Message Structure

The GALILEO E1OS navigation message is called I/NAV message and contains SoL, CS and OS services. This study is only interested in the OS part since this part provides a service equivalent to GPS L1C signal. Therefore, the parts of the message referred to the Open Service are explained whereas the other ones are simply mentioned as reserved. Moreover, the public ICD GALILEO draft only explains the OS part.

The different contents described in section 2.1 are carried by part of pages, pages, subframes and frames which form the I/NAV message. A scheme representing the whole structure of the I/NAV message is given below [ESA, 2008].

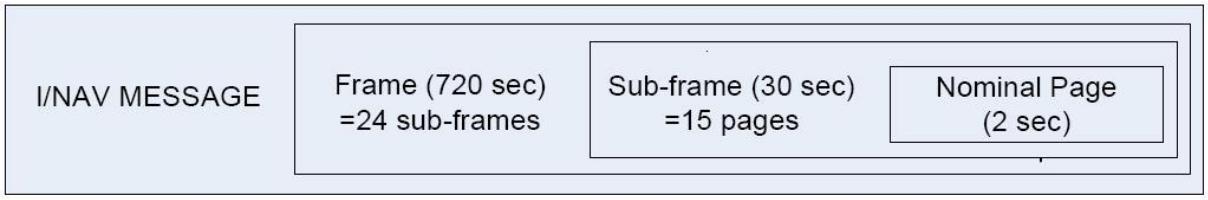

Figure 2-13: I/NAV Message Structure on the Nominal Mode [ESA, 2008]

Each page is divided into two parts denoted as parts of a page. Each part of a page carries 120 information bits which are encoded with a convolutional code of rate ½ resulting in 240 coded bits. The part of a page is completed after the insertion of 10 synchronization symbols. Each part of a page has a duration of 1 second since it is formed by 250 coded bits. Figure 2-14 shows the construction of a part of a page [ESA, 2008].

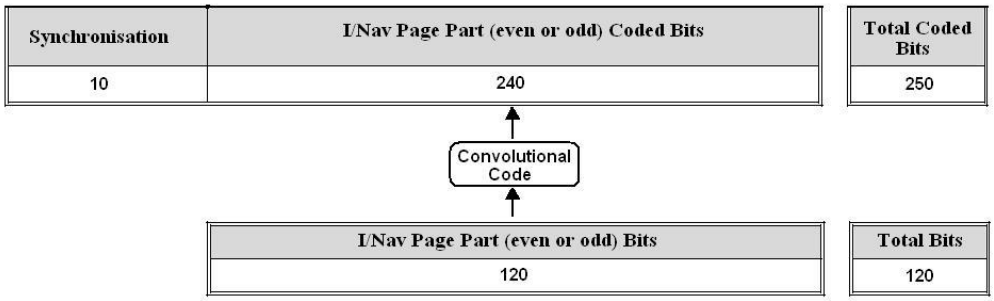

Figure 2-14: Even or Odd Part of a Page Structure Galileo E1 OS [ESA, 2008]

Besides, an interleaver is applied to each individual part of a page before introducing the synchronization symbols. A detailed explanation of the interleaver is given in annex D.5.1.2.

A page has a duration of 2 seconds and it is the minimal information unit that has to be received before passing the transmitted information to higher layers of the receiver. This means that the reception of a part of a page cannot be used until the other part of a page is correctly received. A page is formed by 240 information bits out of which 24 are parity bits calculated by the CRC-24Q channel code. These bits protect almost all the information bits of the page. The structure of a nominal page is described below [ESA, 2008].

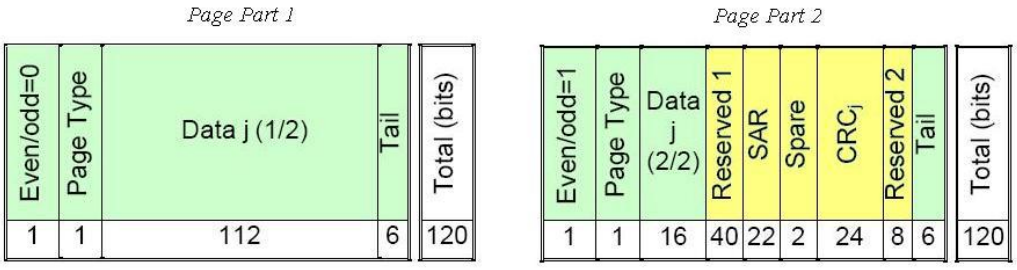

Figure 2-15: Page Structure Galileo E1 OS [ESA, 2008]

Additionally, the I/NAV message can contain alert pages which last 1 second since the two parts of a page are transmitted at the same time but in the GALILEO E1 and GALILEO E5-I signals.

A subframe is formed by 15 pages out of which 4 carry the satellite ephemeris data and clock correction error data. The same ephemeris data set and clock error correction data set are repeated for several consecutives subframes although the exact duration of the repetition period of time is not determined. A subframe has a duration of 30 seconds.

A frame is formed by 24 subframes, which is the number of subframes necessary to broadcast the almanac data of all the satellites. A frame has a duration of 720 seconds or 12 minutes.

A more detailed definition of each unit of information is given in annex D.5.1.

2.6.3. Channel Code

The channel codes implemented in the GALILEO E1 OS navigation message follow the same principles as the last GPS navigation message conceptions: a channel code serial implementation where the inner channel code provides the FEC (Forward Error Correction) and the outer channel code verifies if the parts of a page provided by the inner channel code are error free. If they are not, they are discarded; otherwise the information is stored.

The inner channel code is a convolutional code (171, 133, r = ½) which is used as a block channel code due to the tail bits introduction. The outer code is the cyclic code CRC-24Q.

2.6.3.1. Inner Channel Code

The inner channel code of the GALILEO E1 OS signal is the same convolutional code (171, 133, r = ½) implemented on the GPS L2C and GPS L5 navigation messages (see E.3). However, there is an important difference in its implementation compared to the GPS L2C and GPS L5 navigation messages cases. In this case, the insertion of tail bits at the end of each part of a page makes the initial and the final state of each coded part of a page known. Therefore, the decoding process is simplified. A more detailed explanation is given in annex E.3.2.3.

2.6.3.2. Outer Channel Code

The outer channel code of the GALILEO E1 OS navigation message is the same code as the outer channel code of the GPS L1C, L5 and L2C navigation messages: CRC-24Q. Its characteristics have been already described and commented in annex E.2.

2.7. Conclusion

With the description of the GALILEO E1 OS signal outer channel code, all the main characteristics of the signals inspected along this Ph.D. manuscript have been described from the dissertation point of view. Therefore, this chapter is concluded and the dissertation continues by describing the different transmission channel models used in the Ph.D. analyses.

Chapter 3. Transmission Channels

The received signal is characterized, among other important factors, by the channel through which the signal is propagated. For example, the transmission channel determines the multipath presence and its characteristics over the received signal, specifies which type of noise is found at the receiver antenna input and determines the attenuation applied at the received signal. Therefore, depending on the transmission channel characteristics, it is possible to model and to classify the transmission channels in different categories.

The characterization of the transmission channel is very important because depending on the characteristics of the received signal the demodulation, acquisition, and tacking performance of the signal and the techniques used to improve them differ. Therefore, a mathematical transmission channel model as near as possible to the real transmission channel leads to a very accurate analysis of the demodulation, acquisition, tracking performance of the signal and also leads to the proper choice of the technique maximizing these performances.

The transmission channel is mainly defined by the physical medium that is used to send the signal from the transmitter to the receiver [PROAKISa, 2001]. Besides, in order to simplify the analysis of a communication system, the transmission channel also includes in this dissertation the transmitter antenna, the receiver antenna and the first stages of the receiver.

The transmission channel of a communication system is determined by the environment in which the communication takes place. This means that depending on the landscape -mountains, valleys, etc-, depending on the number of buildings -their size, their distribution, etc- depending on the number of trees -their type, their number of leaves, etc-, etc, the resulting transmission channel varies.

More specifically, each environment has its own individual characteristics, which means that the transmission of a signal through a given environment is different from the transmission of the same signal but made through another environment. The reason is that even the position of a tree, the size of a building, the position of the broadcasting satellite, etc, changes the exact characteristics of the received signal. Therefore, since it is impossible to reproduce for each possible environment the exact properties of the received signal, several statistics mathematical models have been developed and have been associated to the environments with the same general characteristics.

In this dissertation, the transmission channel and its mathematical model are very important, because one of the main objectives of this research work is to optimize the demodulation performance. And the demodulation performance directly depends on the transmission channel. Therefore, if the received signal with all its particularities and properties is not correctly represented in the receiver, the study will not be completely rigorous.

Consequently, in this section, two different channels are presented: the Additive White Gaussian Noise channel and the Mobile Channel. The first channel is the simplest channel model that can be assumed, but it perfectly reproduces the characteristics of some environments, such as rural environments or motorway communications. Besides, this channel can also be used as a simplification or as a first step for more advanced and complicated transmission channels analyses.

3.1. Additive White Gaussian Noise Transmission Channel

The Additive White Gaussian Noise channel is the simplest mathematical model of a transmission channel. In this thesis, this model is used as a simplification for the demodulation performance study of the different GNSS signals although it fairly reproduces the characteristics of an open sky environment.

This section specifies which environments can be modeled by this transmission channel, describes the associated mathematical model, explains its influence over the tracking and demodulation processes and finally describes how this mathematical model has been implemented in the dissertation simulations.

3.1.1. Targeted environments

The targeted environments of the Additive White Gaussian Noise (AWGN) transmission channel mathematical model are the open sky environments. An open sky environment is a line-of-sight (LoS) scenario where the only received ray is the direct ray. This means that one and only one ray is received and no echo -reflected rays, refracted, etc- exists, or if they do, they have a negligible power compared to the direct ray.

The typical open sky environments where this model can be correctly applied are the airborne, rural or in some case the semi-rural environments. These environments were the targeted environments by the US military force when the original GPS system was conceived. Moreover, in these environments, the C/N_0 of the received signals is assumed to be relatively high and thus no demodulation optimization is necessary. Nevertheless, if this model is analyzed with low C/N_0, the results can be used to make a first approximation of more complex transmission channels.

3.1.2. Mathematical Model

The mathematical model of an AWGN channel is simply the addition of a white Gaussian noise to the LOS signal [PROAKISa, 2001]. The white Gaussian noise is modeled as a signal having a flat power density spectrum equal to N_0 or $N_0/2$ (depending on the signal mathematical representation, if the signal is expressed by either its equivalent baseband expression or its real pass-band expression). Figure 3-1 illustrates the AWGN channel mathematical model.

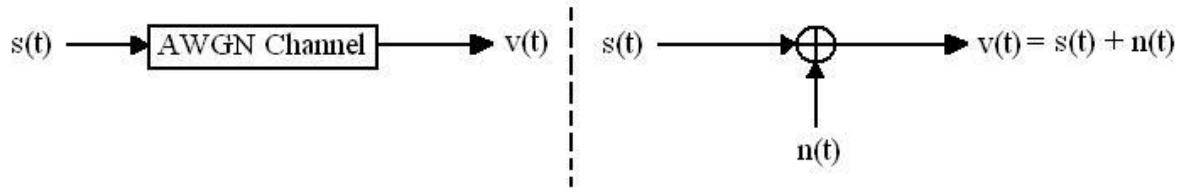

Figure 3-1: Mathematical Model of an AWGN channel

Where:

- s(t): Transmitted signal at transmitter antenna input
- v(t): Received signal at receiver antenna output
- n(t): White Gaussian noise equivalent to all noise contributions

Therefore, since the density power spectrum of a white Gaussian noise is unbounded in frequency, its power is infinite. However, during the reception process, the first action carried out by the receiver is to filter the signal. Therefore, the real noise found during the next signal processing is a narrow-band noise whose density power spectrum is bounded by the useful signal bandwidth. More specifically, this new narrow-band noise results from the filtering of the white Gaussian noise by the RF/IF block and thus its power (P_n) can be approximated by multiplying its constant density spectrum power level by the useful signal bandwidth (B):

$$P_n = N_0 \cdot B \tag{3-1}$$

The impact of the noise can be expressed through its influence on the signal power by the signal-to-noise ratio or SNR figure of merit. This figure of merit is defined in annex B.1.

$$SNR = \frac{C}{P_n} \tag{3-2}$$

Where:

- C: Useful Signal Power at the filter output

Moreover, the SNR can be easily related to the C/N_0 ratio from equation (B-9) as has been done in annex B.3.

$$\frac{C}{N_0} = SNR \cdot B \tag{3-3}$$

3.1.3. Impact of the AWGN channel on the carrier phase tracking process

The impact of the AWGN channel on the carrier phase tracking process refers to the carrier phase estimation error caused by the transmission channel. Therefore, the next subsections explain what a carrier phase estimation error is, describe which are the sources of this error and gives its mathematical model.

Also, if the C/N_0 values set by the channel are too low, the carrier phase tracking process may not be possible. This phenomenon corresponds to the PLL loss of lock [KAPLAN and HEGARTYb, 2006] and changes completely the impact of the AWGN channel on the carrier tracking process. Also, the demodulation process cannot be accomplished if the carrier phase is not estimated. Therefore this PLL loss of lock is also analysed in the following subsections. To sum up, the carrier phase estimation can be ideal, degraded or impossible to achieve.

Finally, before beginning with the explanations, it is reminded that the effects of the FLL have been neglected since for the C/N_0 values analysed in this dissertation, the FLL achieves a lock and the carrier frequency estimation error is negligible [MACABIAU et al, 2003]. Therefore only the impact over the PLL is given.

3.1.3.1. Phase estimation error

The phase estimation error is the difference between the phase of the received carrier and the phase of the carrier locally generated by the PLL due to the existence of different noise sources.

From the demodulation point of view, the PLL carrier phase estimation error is modelled as a cosine factor multiplying the I channel useful signal as it is shown below, and in annex A.2

and A.3.2. The code delay estimation has been assumed ideal. The next equation represent the I channel at the correlator output.

$$r_i(t) = \frac{A}{2} \cdot d(t) \cdot \cos\left(\theta - \hat{\theta}\right) + n_i(t) \tag{3-4}$$

Where:

- $r_i(t)$: signal at the correlator output of I channel
- A: signal amplitude
- d(t): navigation data waveform
- $\hat{\theta}$: estimated signal carrier phase
- θ: received signal carrier phase
- $n_i(t)$: I channel noise

3.1.3.2. Phase estimation error sources

According to [KAPLAN and HEGARTYb, 2006] and [IRSIGLER and EISSFELLER, 2002] there are 4 main sources of error affecting the PLL tracking performance. These 4 sources are: the thermal error noise, the dynamic stress error and the oscillator phase noise which is caused by 2 different effects, the Allan deviation noise and the oscillator vibrations. The expressions of the standard deviation (σ_{tot}) and the bias (θ_{tot}) of the total carrier phase estimation error (ε_θ) caused by the four sources are the following:

$$\theta_{tot} = \theta_e \tag{3-5}$$

$$\sigma_{tot} = \sqrt{\sigma_{th}^{\ 2} + \sigma_{Vib}^{\ 2} + \sigma_{Osc}^{\ 2}} \tag{3-6}$$

Where:

- σ_{th}: Standard deviation thermal noise error
- σ_{Vib}: Standard deviation oscillator vibrations error
- σ_{Osc}: Standard deviation Allan deviation noise error
- θ_e: Dynamic stress error bias

The total carrier phase estimation error, ε_θ, has to be interpreted as a white Gaussian noise filtered by the equivalent PLL filter of bandwidth B_L which has a standard deviation equal to σ_{tot} and which is centered on $\theta_{tot.}$

The error source called thermal noise results from the impossibility of removing all the narrow-band noise existent at the RF/IF output block. The mathematical expression of the variance of the carrier phase estimation error is particularized for each different discriminator. And a discriminator is the element of the PLL responsible for measuring the phase error between the incoming signal carrier phase and the generated local carrier phase replica. The most used discriminators are the Q or coherent discriminator, the Costas Discriminator, the Arctangent discriminator and the Arctangent2 discriminator. These discriminators are described in annex A.3.3.

The error variance for the product discriminator also known as (generic) Costas discriminator is:

$$\sigma_{\varepsilon}^{2} = \frac{B_L}{C/N_0} \cdot \left(1 + \frac{1}{2 \cdot T_I \cdot (C/N_0)}\right) \tag{3-7}$$

Where:

- σ_{ε}^{2}: Variance of the carrier phase estimation error due to thermal noise
- B_L: PLL filter Bandwidth
- T_I: Coherent integration Time

The expression for the Q discriminator is:

$$\sigma_{\varepsilon}^{2} = \frac{B_L}{C/N_0} \tag{3-8}$$

The expressions for the ArcTangent and ArcTangent2 discriminators are very difficult to obtain mathematically; however it has been shown through Monte Carlo simulations that their expression can be approximated by the same expression as the product discriminator [SPILKER and ASHBYd, 1996].

The dynamic stress error is phase jitter caused by the permanent motion of the satellites and the possible receiver motion [IRSIGLER and EISSFELLER, 2002], or in other words, it is phase jitter caused by the signal dynamics.

The mathematical model of the dynamic stress error of a 3rd order PLL when the dynamic stress error is a constant jerk is given below [STEPHENS and THOMAS, 1995].

$$\theta_e = 2\pi \frac{T_I^{\,3}}{K_3} \cdot \frac{m \cdot g}{\lambda} (rad) \tag{3-9}$$

Where:

- θ_e: Dynamic stress error
- m: Constant number of jerks
- g: gravity acceleration
- λ: Wavelength of the carrier signal frequency
- T_I: Integration Time
- K_3: coefficient given by [STEPHENS and THOMAS, 1995], in their description of discrete-update PLL.

In this dissertation, two different types of users with different worst jerks have been considered, a pedestrian and a car. Table 3-1 shows the constant jerk values for each user.

Type of User	*Pedestrian*	*Car*
Jerk (g/s)	1 g/s	0 g/s

Table 3-1: Constant jerk values for different types of users considered in this Ph.D.

The Allan deviation noise or the oscillator frequency noise is the result of the instability of the oscillator central frequency that produces a phase jitter at the local carrier replica level [JULIEN, et al. 2005]. It can be termed system-inherent phase noise and is relevant for both dynamic and static applications [IRSIGLER and EISSFELLER, 2002].

The oscillator vibrations source of error represents the phase noise caused by the external vibrations affecting the oscillator [JULIEN, et al. 2005]. It is called external phase noise and it is a major problem for dynamic applications [IRSIGLER and EISSFELLER, 2002].

A scheme summarizing the different sources of error affecting a PLL is shown below [IRSIGLER and EISSFELLER, 2002].

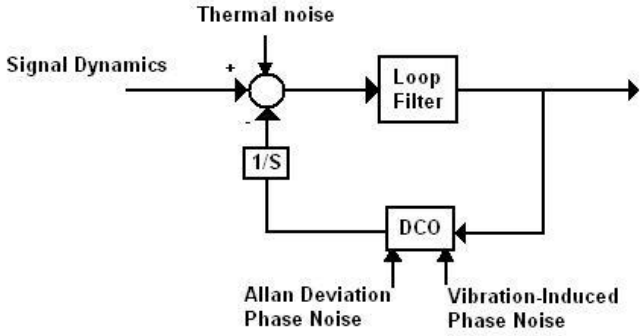

Figure 3-2: Generic track loop diagram with potential error sources [IRSIGLER and EISSFELLER, 2002]

The thermal noise and the dynamic stress error are further explained in annex A.3.3.1 and A.3.3.2.

Finally, the only two sources of error taken into account during this research work are the thermal noise and the dynamic stress error. The reasons are given next. The dynamic stress error only depends on the receiver dynamics and its influence can be the largest of all the sources; therefore this source must be simulated and its influence depends on the type of receiver analyzed, e.g. a pedestrian or a car. The Allan deviation noise and the oscillator vibrations do not depend on the C/N_0 of the received signal, whereas the thermal noise does. Therefore, since for the C/N_0 levels analyzed all along this dissertation, the influence of the thermal noise is larger than the influence of the two other sources, these sources have been neglected in this dissertation: the highest total C/N_0 value for simulations not assuming ideal carrier phase tracking is 32 dB-Hz. The next figure shows the influence of the sources as a function of the C/N_0 [JULIEN, 2009] and justifies why the Allan deviation noise and the oscillator vibrations can be neglected.

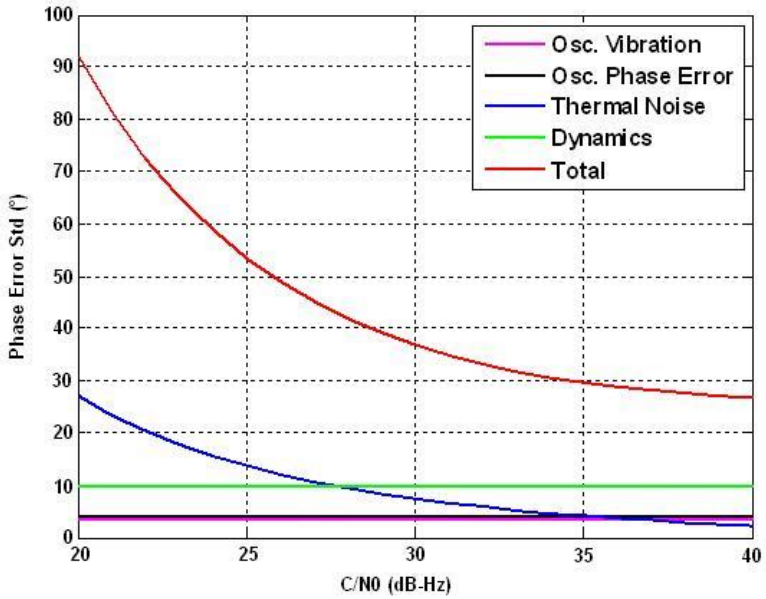

Figure 3-3: Phase Error Standard Deviation of the different phase error sources as a function of the pilot C/N0. Jerk = 0.25 g/s, TCXO, B_L = 10Hz. [JULIEN, 2009]

From Figure 3-3, it can be seen that for values equal or smaller than 32 dB-Hz, the Allan deviation noise and the oscillator vibrations impact are marginal in front of the thermal noise. These curves are plotted for a bandwidth of the PLL filter, B_L, equal to 10Hz which is the same bandwidth used in the simulations and in the analysis of this research work. Note that the C/N_0 is measured on the pilot channel; therefore the total C/N_0 value is bigger. For example, when the C/N_0 of the pilot channel is 32 dB-Hz, the total C/N_0 is 35 dB-Hz for GALILEO E1 and about 33.25 dB-Hz for GPS L1C. Moreover, Figure 3-3 depicts the standard deviation, therefore in terms of power or variance the difference is quite larger.

Finally, Figure 3-3 represents the standard deviation of the phase estimation errors for a TCXO of average quality. This means that the standard deviation values of the phase estimation errors introduced by the Allen deviation noise and the oscillator vibrations sources of error for the lowest caliber TXCOs will be larger than the values represented in Figure 3-3. Therefore, the validity of neglecting these sources of error in this dissertation for low caliber TXCOs and for the inspected values of C/N_0 is not justified. However, a quick reasoning can validate this hypothesis in absence of a more detailed analysis.

For low caliber TXCOs, the internal noise of the oscillator is larger (Allen deviation noise and oscillator vibrations) than for an average caliber TCXOs; therefore, in order to still be able to track the signal carrier phase in the presence of the same signal dynamics, a larger B_L is implemented. This means that the dynamic stress error bias, the Allen deviation noise and the oscillator vibrations standard deviation remain about the same since they are inversely proportional to B_L for average and low calibers TCXO. However, the standard deviation of the phase estimation error due to the thermal noise is increased since it is proportional to B_L. Therefore, if this usual course of action is taken, the C/N_0 values at which the impact of the Allen deviation noise and the oscillator vibrations with respect to the thermal noise are negligible remains about the same. Nevertheless, a specific study should be conducted in order to guarantee this hypothesis.

3.1.3.3. PLL loss of lock threshold

An important characteristic of the carrier tracking process is the PLL loss of lock threshold. This PLL loss of lock threshold indicates the minimum level of C/N_0 at which the PLL still provides a satisfying coherent carrier phase estimation. The definition of the PLL loss of lock threshold is presented in annex A.3.3.3.

In this research work, it has been assumed that for an AWGN channel the PLL never loses its lock. Therefore, in order to guarantee this assumption, we have to calculate the PLL loss of lock thresholds taking into account the specific problems conditions modeled in this dissertation. These threshold values will be compared the C/N_0 values used in all the simulations of this research work in order to verify that the PLL is always locked. Moreover, some realistic threshold values which will serve to validate the values calculated in this section are presented in annex A.3.3.3.

The calculation of the PLL loss of lock thresholds for the dissertation conditions is detailed next. First, the general equation can be recovered from annex A.3.3.3:

$$3\sigma_{\varphi} + \theta_e \leq \frac{L_{\varphi}}{2} \tag{3-10}$$

Where:

- σ_{φ}: Standard deviation of the total carrier phase estimation error
- θ_e: Dynamic stress error

Second, equation (3-10) may be simplified into equation (3-11) since the oscillator noise and the Allan deviation noise are neglected:

$$3\sigma_{th} + \theta_e \leq \frac{L\varphi}{2} \tag{3-11}$$

Where:

- σ_{th}: Thermal noise standard deviation error
- L_{φ}: Two-sided discriminator linear tracking region

Third, the new L_{φ} is determined. The two sided discriminator linear region depends on the type of discriminator and can vary with the pilot channel C/N_0 value as can be observed in Figure A-6. Nevertheless, in this dissertation, the only implemented discriminator is the Q discriminator as justified in annex A.3.3.4, and this discriminator has a constant linear region. The Q discriminator two-sided discriminator linear tracking region is presented in Table 3-2 altogether with the linear region of the product discriminator. These values are taken from [KAPLAN and HEGARTYb, 2006].

	Product discriminator onto a data channel	***Q discriminator onto a pilot (dataless) channel***
L_{φ}	45^{o}	90^{o}

Table 3-2: Two-sided linear tracking region for different channels and discriminators

Nevertheless, this choice of the two-sided discriminator linear region is not unique. This choice varies depending on the accepted degradation of the linear region because this region is not completely linear at its edges. This means that the linear region limit depends on the tolerated degradation between the discriminator curve and the real linear curve (f(x) = x). For

example, for the Table A-1 and Table A-2 values, the maximum degradation between these two curves is a 0.1 factor [JULIEN, et al. 2005]. However, much larger degradation values can be accepted.

Finally, the values of the PLL loss of lock threshold for a Q discriminator are calculated by particularizing equation (3-11) with the Table 3-2 values and with equations (3-8) and (3-9).

$$\sigma_{th} = \frac{L\varphi}{6} - \frac{\theta_e}{3} \tag{3-12}$$

$$\frac{C}{N_0} = \frac{B_L}{\left(\frac{L\varphi}{6} - \frac{2}{3}\pi \frac{T_I^3}{K_3} \cdot \frac{m \cdot g}{\lambda} \right)^2} \tag{3-13}$$

The PLL loss of lock thresholds are obtained by evaluating equation (3-13) with the different problem conditions. These conditions are a PLL 3rd order filter, a PLL bandwidth (B_L) equal to 10Hz and a coherent integration time equal to 20ms. Besides, two cases are inspected for the signal dynamics jerk, 0 and 1 g/s. The results are presented below

	GNSS signal			
Jerk (g/s)	**GPS L2C**	**GPS L5**	**GPS L1C**	**GALILEO E1**
0	18.6	18.6	16.9	18.6
1	22.2	22.0	21.9	23.6

Table 3-3: PLL tracking loss threshold for a Q discriminator. The thresholds represent the total C/N0 of the received signal.

Threshold values presented in this table are quite low in comparison with those of Table A-1 and Table A-2. Nevertheless, this significant difference may be justified by three facts. The first is that two different two-sided linear region values have been applied in each calculation. In this Ph.D. calculation, the $L_\varphi/2$ is equal to 90 degrees or 1.5708 radians, whereas Table A-1 and Table A-2 values are calculated using a $L_\varphi/2$ equal to about 0.8 radians. The second is that different PLL filter bandwidth values have been assumed; 5Hz for the Table A-1 values and 10Hz in this case. Therefore, if the threshold is recalculated with Table A-1 two-sided linear tracking region and with Table A-1 B_L for the GALILEO E1 signal, a new threshold of 21.5 dB-Hz would be obtained. And this number presents a minimal difference with the Table A-1 value which is justified by the third fact. This third fact is that the oscillator vibrations and the Allan deviation noise contributions have been neglected.

In conclusion, the values presented in Table 3-3 are reasonable: they are the PLL loss of lock threshold of the cases simulated in this research work.

3.1.4. Impact of the AWGN channel over the demodulation process

The demodulation process of a digital signal consists in estimating the transmitted symbols from the received signal waveforms or/and received signal waveform amplitudes (see annex C.1).

In this subsection, in order to analyse the impact of the AWGN channel over the demodulation process, a general GNSS received signal at the correlator output is modelled. This model is simplified by assuming that the received signal subcarrier modulation –if the

signal has any- has been perfectly removed and that the code delay estimation is ideal. Note that the subcarrier does not affect the demodulation process if a matched filter of the subcarrier (or equivalent) is implemented in the receiver as it is the case. In fact, the subcarrier function consists in changing the signal autocorrelation function, where the signal autocorrelation function is mainly defined by the PRN code waveform autocorrelation. This model is presented in equation (3-14).

$$r_i[k] = \frac{A}{2} \cdot d[k] \cdot \cos\left(\theta[k] - \hat{\theta}[k]\right) + n_i[k] \tag{3-14}$$

$$r_i[k] = \frac{A}{2} \cdot d[k] \cdot \cos\left(\varepsilon_\theta[k]\right) + n_i[k] \tag{3-15}$$

Where

- $r_i[k]$: Digital correlator output signal of the I channel at epoch k
- $d_m[k]$: Navigation data at epoch k
- $\theta[k]$: Incoming signal carrier phase at epoch k
- $\hat{\theta}[k]$: Estimated signal carrier phase at epoch k
- $\varepsilon_\theta[k]$: Carrier phase estimation error at epoch k
- $n_i[k]$: Impact of the noise introduced by the AWG channel at epoch k at the correlator output

From equation (3-15) and knowing that d[k] is the data value to be estimated, two distortion factors introduced by the AWG channel can be observed. The first one is the noise itself, which appears in the form of $n_i[k]$ and adds a Gaussian variable to the d[k] value. Depending on the magnitude $n_i[k]$ with respect to the magnitude d[k], the noise corrupt the data value in a higher or lower degree.

The second distortion factor is the cosine factor. This cosine factor is different from 1 when the signal carrier phase is not perfectly estimated. And as it has been seen in subsection 3.1.3.2, the perfect carrier phase estimation is not achieved due to the AWG noise among other source of errors. In other words, the AWG noise affects the tracking performance, and the tracking performance affects the demodulation performance by decreasing the amplitude d[k]. This is equivalent to saying that not all the useful signal power at the antenna output is used to demodulate the signal due to the impossibility to achieve a perfect carrier phase estimation.

To sum up, the demodulation performance is affected by the additive white Gaussian noise in two aspects. First, the demodulation process cannot use all the useful signal power at the antenna output since the received signal carrier phase is not ideal. Second and last, the waveforms used for demodulation are the useful signal components plus noise.

3.1.5. AWGN channel simulator

In this subsection, the simulators used to simulate the transmission of a signal through an AWGN channel are presented. Two simulators are presented and both are prepared to support the thermal noise and the dynamic stress error as the source of errors of the carrier phase tracking process. Note that for an AWGN channel the carrier phase tracking process is assumed to be made by a PLL and that the code delay tracking process is assumed ideal. A

more detailed explanation and justification of the following simulators can be found in annex H.3.1.

The first simulator generates the simulator input samples at the receiver correlator output, after the estimation of the incoming signal carrier phase made by the PLL. Therefore, this simulator directly generates the estimation carrier phase error made by the PLL. This simulator is based on equation (3-15). The first simulator scheme is given below.

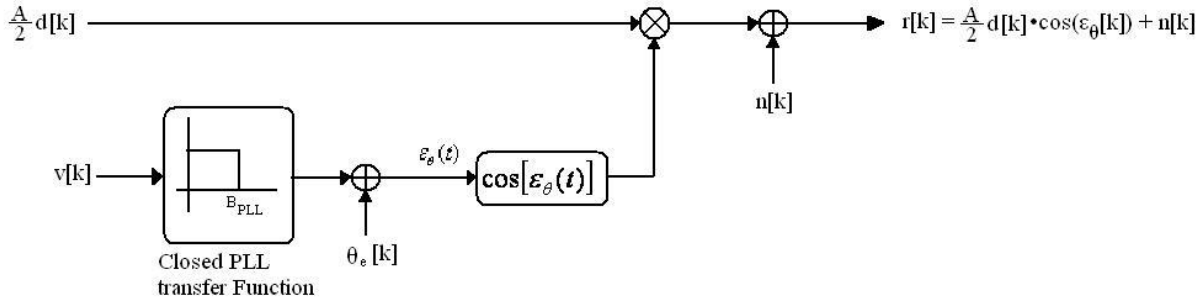

Figure 3-4: AWGN channel with a non-ideal tracking process

Where:

- A: useful signal amplitude
- d[k]: Navigation data at epoch k
- n[k]: discrete white Gaussian noise – $N(0, \sigma^2)$
- v[k]: discrete white Gaussian noise – $N(0, \sigma_\varepsilon^2)$
- $\theta_e[k]$: dynamic stress error bias

The coefficients of the filter representing the closed PLL transfer function are defined in [STEPHENS and THOMAS, 1995]. Nevertheless, these coefficients make that the closed PLL transfer function has a gain different from 1; therefore, in order to keep the desired input noise power, the output noise has to be multiplied by a corrector factor.

The second simulator generates the simulator input samples at the receiver RF/IF block output. Therefore this simulator has to implement a PLL since the estimation of the signal incoming carrier phase is made during the simulation. Besides, this simulator has to generate the original expression of the source of errors of the carrier phase tracking process instead of the error made by the PLL. The second simulator scheme is given below.

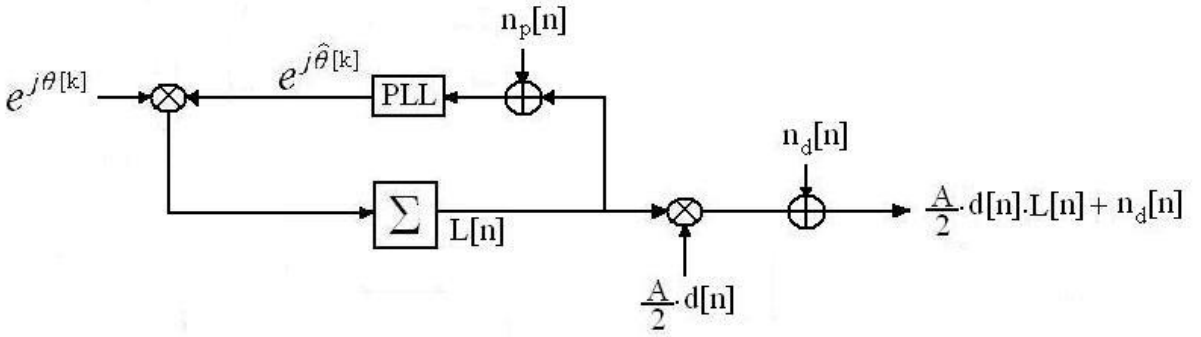

Figure 3-5: Simplified PLL simulator scheme for $\varepsilon_\tau[k] = 0$

Where:

- $\theta[k]$: Incoming signal carrier phase
- $\hat{\theta}[k]$: PLL signal carrier phase estimation
- d[n]: Navigation data at epoch n
- n_p[n]: Pilot channel noise at epoch n.
- n_d[n]: Data channel noise at epoch n.
- k: epoch at chip sampling rate
- n: epoch integration output rate
- L[n]: Complex number at the output of the integration at epoch n

$$L[n] = \frac{1}{M} \sum_{k=0}^{M-1} e^{j\left(\theta[k] - \hat{\theta}[k]\right)} \tag{3-16}$$

- M: Number of samples inside a coherent integration time

3.2. Mobile Transmission Channel

The mobile transmission channel is one of the more complex transmission channel mathematical models. Nevertheless, this model can be simplified if it is used for demodulation performance analysis as it has been done in this dissertation. Therefore, the presented mathematical model should not be used for other practical purposes except for analyzing the signals demodulation performance. In particular, this model is not suited to study the code delay tracking performance.

This section specifies which environments can be modeled by this transmission channel, describes the associated mathematical model, explains its influence over the tracking and demodulation process, and finally describes how this mathematical model has been implemented in the simulations of this research work.

The validity of this mathematical model is given by Perez-Fontan in its papers.

3.2.1. Targeted Environments

The mobile channel mathematical model presented in this section can represent all the environments having the following three principal characteristics.

The first characteristic is a line-of-sight (LOS) signal which can be totally or partially blocked/shadowed during the transmission. The second characteristic is a multipath (or echoes) signal power no longer negligible in front of the LOS signal power. The third and last characteristic is a LOS signal power and a multipath structure which vary with time.

Finally, the types of environment that can be represented by this mathematical model are the mobile environments, or, more specifically, the urban environments or indoor environments. Normally, these environments assume a mobile receiver; however, even if the receiver does not move, the received signal can still vary because the elements of the environment changes, such as the cars, the pedestrians, the leafs of tress, etc. Nevertheless, the main source

responsible for varying the channel is the receiver motion as it is justified in subsection 3.2.2.2.1.

3.2.2. Mathematical model

The mathematical model of a mobile channel has to represent a received signal equal to the sum of the shadowed/blocked LOS signal with the multipath component. Therefore, the received signal is an equivalent signal with a power and a phase which continuously vary. There are two main reasons justifying this variation. The first one is that the addition of the multipath on the LOS signal can have a constructive or destructive effect. The second one is that this addition varies along with the environment variation.

This subsection presents the mathematical model chosen in this research work and used on the simulations. Nevertheless, there are different types of mobile channels and thus, before selecting the model, the type of mobile channel best suiting our needs is identified.

3.2.2.1. Type of mobile channel

The different types of mathematical models for a mobile channel are defined from the channel autocorrelation function. More specifically, the different channel types are defined from the comparison of two main parameters extracted from the channel autocorrelation function with the signal bandwidth, W, and the signal symbol duration, T. These two autocorrelation parameters are the channel coherence bandwidth, $(\Delta f)_c$, and the channel coherence time, $(\Delta t)_c$. Remember that the chip/symbol signal period is approximately the inverse of the signal bandwidth. The mathematical development of the channel autocorrelation function can be found in annex Chapter 17.

The channel coherence bandwidth, $(\Delta f)_c$, determines the minimum separation in frequency which two different sinusoids have to have in order to be affected differently by the channel. Therefore, depending on the signal bandwidth, W, with respect to the channel coherence bandwidth, $(\Delta f)_c$, two types of mobile channel can be defined:

- If $W < (\Delta f)_c$ the channel is frequency non-selective and the entire signal is affected identically by the multipath.
- If $W > (\Delta f)_c$ the channel is frequency-selective and the signal frequency components are distorted differently by the multipath.

The same analysis can be made in the time domain since $(\Delta f)_c$ is the inverse of the multipath spread of the channel, T_m, [PROAKISg, 2001]. In fact, T_m is the channel impulse response length or the spanning of the echoes, and thus, it is quantified by the time offset between the LOS signal and the last echo. Figure 3-6 shows the T_m of a possible received signal:

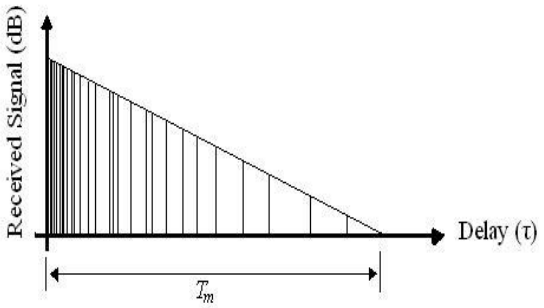

Figure 3-6: Example of spanning of the signal echoes channel impulse response - Multipath spread of the channel

Therefore, it can be seen that depending on the signal symbol duration with respect to T_m, the channel affects differently the signal. In fact, if the signal symbol duration is much larger than the multipath spread, it can be considered that the received symbol is only affected in amplitude and phase since all the echoes arrive mainly during the transmitted LOS symbol duration; however, if the signal symbol duration is smaller, the current transmitted symbol affects the to-be-transmitted symbols and thus an intersymbol interference (ISI) appears. The reason for the ISI is that the echoes arrive mainly after the transmitted LOS symbol duration.

The channel coherence time $(\Delta t)_c$ represents the duration of time in which the channel remains about constant. This variation of the channel is termed *fading* [PROAKISg, 2001]. Therefore, the longer the signal symbol duration is, the more the symbol amplitude and phase vary.

Two mobile channel types can be defined by comparing the signal symbol duration, T, with the channel coherence time.

- If $T < (\Delta t)_c$ the channel is slowly fading and is considered invariant during the symbol duration (T).
- If $T > (\Delta t)_c$ the channel is not slowly fading.

Finally, one important remark has to be done concerning the channel coherence time, $(\Delta t)_c$, the channel coherence bandwidth, $(\Delta f)_c$, the signal bandwidth, W, and the signal symbol duration, T. The comparison between the channel coherence bandwidth, $(\Delta f)_c$, and the signal bandwidth, W, –inverse of T– is always made at the receiver antenna output. This means that the signal symbol duration is equal to the time chip duration, T_c, for a GNSS signal. However, the comparison between the channel coherence time, $(\Delta t)_c$, and the signal symbol duration, T, can be made at the receiver antenna output and at the receiver correlator output. In the first case, the signal symbol duration, T, is equal to the time chip duration, T_c, whereas in the second case, the signal symbol duration, T, is equal to the data symbol duration, T_d. Moreover, note that, in our case, the second comparison is more important. The reason is that if the channel does not vary during the duration of a data symbol, the data and specially the PRN code are not distorted during the reception of a data symbol. Therefore, since a PRN code period is normally equal to the data symbol duration for the modern GNSS signals and since the PRN codes conserve their initial properties, the receiver is still able to separate the pilot and data channels from themselves and from other satellite channels.

3.2.2.2. Determination of Mobile Channel model type

In this subsection, the type of mobile channel model best suited to the characteristics of the propagation channel from the demodulation performance point of view is determined.

3.2.2.2.1. Slowly fading channel

The first characteristic is whether the channel is slowly fading or not. The channel coherence time is compared with different signal symbol durations. In order to do so, first, the signal Doppler spread σ_v, which is the inverse of $(\Delta t)_c$ [PROAKISg, 2001], is computed. The Doppler spread power can be modeled as the sum of three terms [FRIGYES et al., 2001]:

$$\sigma_v^{\,2} = \left(\frac{V_g}{\Lambda_c}\right)^2 + \left(\frac{\Omega_S}{\alpha_c}\right)^2 + \left(\frac{1}{T_{ch}}\right)^2 \tag{3-17}$$

With:

- (V_g/Λ_c): Doppler spread introduced by the mobile motion.
  - V_g = Mobile Speed with respect to the local reference frame.
  - Λ_c = Coherence length which is usually of the order of a signal wavelength (λ).
- (Ω_s/α_c): Doppler spread introduced by the satellite motion
  - Ω_s: Satellite Angular Velocity
  - α_c: Coherence Angle which is determined by the structure of the scenario scatters.
- T_{ch} = Channel self-Doppler Spread

Concerning the Doppler spread introduced by the satellite motion, its typical value for a LEO (Low Earth Orbit) satellite is about 1Hz [FRIGYES et al., 2001], and thus, since the Ω_s is smaller for a MEO (Medium Earth Orbit) satellite, this term can be neglected in front of the Doppler spread introduced by the mobile motion. Moreover, T_{ch}, the channel self-Doppler Spread defined as the Doppler spread of the received signal if neither the satellite nor the ground station moves, is also marginal in comparison with the Doppler spread caused by the mobile motion [FRIGYES et al., 2001]. Therefore, the signal Doppler spread can be reduced to only one term:

$$\sigma_v \approx \frac{V_g}{\lambda} \tag{3-18}$$

From this expression, it is possible to calculate for a given symbol or chip period the maximum speed at which the mobile should travel if the channel is slowly fading. Indeed, if we want:

$$(\Delta t)_c > T \tag{3-19}$$

Then:

$$V_g < \frac{c}{T \cdot f_c} \tag{3-20}$$

With:

- T: signal symbol period
- c: velocity of light in vacuum
- f_c: signal frequency carrier

Nevertheless, note that expression (3-19) is usually $(\Delta t)_c >> T$, with $(\Delta t)_c \sim 100 \cdot T$ [PROAKISg, 2001]. But in our case and for an initial inspection of the demodulation performance of a GNSS signal transmitted through a mobile channel, we have decided to use (3-19).

Table 3-4 gives some examples of the maximum possible speed for different symbol/chip periods and different signals.

		GNSS Signals			
		GALILEO E1 OS	GPS L1C	GPS L2C	GPS L5
Symbol or Chip Period (ms)	20	34.3 km/h	34.3 km/h	44 km/h	45.9 km/h
	10	68.6 km/h	68.6 km/h	88 km/h	91.8 km/h
	4	171.4 km/h	171.4 km/h	219.9 km/h	229.5 km/h

Table 3-4: Maximum allowed speed of a mobile as a function of the symbol/chip period for a slowly fading channel for GPS L1C, L2C and L5, GALILEO E1 OS signals.

To sum up, for GALILEO E1 OS and GPS L1C and for a mobile travelling at speed of 30km/h, the channel is slowly fading at the correlator output even if the signal symbol period was equal to 20ms. More specifically, for GPS L1C, the channel is slowly fading for a mobile traveling at a maximum of 68.6 km/h. For GALILEO E1 OS, the channel is slowly fading for a mobile traveling at a maximum of 171.38 km/h. Therefore, the channel type at the correlator output depends on the mobile velocity.

3.2.2.2.2. Frequency-selective and frequency non-selective channel

The determination of whether the channel is frequency-selective type or channel frequency non-selective type is not done in this dissertation. In fact, the same mathematical model for both channels is applied in this research work. The justification is given next, and it is based on the spreading spectrum signal construction and on the assumption that the channel is always slowly fading at the receiver correlator output.

The mathematical model of a frequency-selective channel is a tapped delay line with tap spacing 1/W and tap weight coefficients named $\{c_n(t)\}$ [PROAKISg, 2001]. Remember that W is the signal bandwidth with T = 1/W, and, in this case, T is equal to the time chip duration, T_c, since the signal is modeled at the receiver antenna output. The $\{c_n(t)\}$ coefficients are generated following a distribution law which depends on the model of the received signal. The different distributions are explained later in subsections 3.2.2.3 and 3.2.2.3.5.

In practice, the number of taps of the delay line is not infinite but truncated at $L = [T_m W] + 1$, where T_m is the inverse of the multipath spread of the channel, or, in other words, the spanning of the echoes. Figure 3-7 is an illustration of such a channel model where $s_l(t)$ is the signal equivalent baseband complex envelope at the transmitter antenna input and $v_l(t)$ is the signal equivalent baseband complex envelope at the receiver antenna output named:

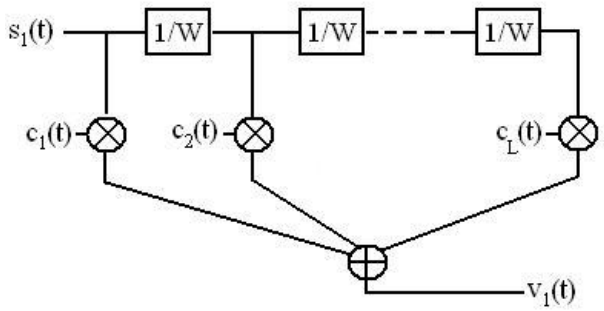

Figure 3-7: Mathematical model for a frequency-selective channel - A truncated tapped delay line [PROAKISg, 2001]

The channel model presented in Figure 3-7 has to be interpreted as it is explained next. The received signal at the receiver antenna output is the transmitted LOS signal -which can be attenuated- followed by echoes of the transmitted signal. The time between the reception of the LOS signal and the reception of the last echo is larger than the symbol duration of the signal because the channel is frequency-selective. The main idea behind the model presented in Figure 3-7 is to model the effect of all the received echoes as the reception of a finite number of main echoes which are weighted by complex coefficients. These main echoes are time separated of each other and of the LOS signal by the signal symbol duration, T, where T is equal to T_c in a general GNSS signal case. Each one of these main echoes gathers all the original echoes received between itself and half the time to the previous main echo and half the time to the next main echo. The element used to represent a gathering of echoes (or tap) is the complex coefficient of each main echo, $c_n(t)$. More specifically, this coefficient, $c_n(t)$ $n \neq 1$, models the contribution to the final received signal carrier phase and to the final received signal amplitude of all the modeled original echoes (half chip time before and half chip time after) around one main echo. And this coefficient also includes the contribution to the final received signal of the main echo. Moreover, the original echoes received between the LOS signal and half the chip time to the first main echo are also modeled by a complex coefficient, $c_1(t)$. Obviously, the $c_1(t)$ coefficient also models the contribution of the LOS signal to the final received signal.

The noiseless signal complex envelope at the receiver antenna output, $v_l(t)$, is equal to:

$$v_l(t) = \sum_{n=1}^{L} c_k(t) s_l\left(t - \frac{n}{W}\right) \tag{3-21}$$

The noiseless signal complex envelope after the RF/IF block, $w_l(t)$, is:

$$w_l(t) = h_l(t) * v_l(t) \tag{3-22}$$

Where:

- $h_l(t)$: RF/IF filter equivalent baseband complex envelope expression And, if the $s_l(t)$ is modeled as it is presented in equation (A-4):

$$s_l(t) = \left[d(t) \cdot c_d(t) + c_p(t)\right] \cdot e^{j\theta(t)} \tag{3-23}$$

The noiseless signal equivalent baseband complex envelope expression after the RF/IF block, $w_l(t)$, can be also derived when the channel is assumed slowly fading at the receiver antenna output:

$$w_l(t) = \sum_{n=1}^{L} c_k(t) s_l'\left(t - \frac{n}{W}\right) \tag{3-24}$$

$$s_l'(t) = \left[d(t) \cdot c_d'(t) + c_p'(t)\right] \cdot e^{j\theta(t)} \tag{3-25}$$

Where:

- c_d'(t): Filtered data channel PRN code waveform
- c_p'(t): Filtered pilot channel PRN code waveform

Therefore, the noiseless received equivalent baseband complex envelope signal at correlator output, $r_l(t)$, can be modeled as:

$$r_l[k] = \frac{1}{T}\int_{(k-1)T}^{kT} w_l(t) \cdot c_{PRN\,i}(t) \cdot e^{j\hat{\theta}(t)} dt \tag{3-26}$$

$$r_l[k] = \frac{1}{T}\sum_{n=1}^{L}\left[\int_{(k-1)T}^{kT} c_k(t) \cdot s_l'\left(t - \frac{n}{W}\right) \cdot c_{PRN\,i}(t) \cdot e^{j\hat{\theta}(t)} dt\right] \tag{3-27}$$

With:

- $c_{PRNi}(t)$: PRN code waveform of the satellite i

Equation (3-27) can be simplified since we are assuming that the channel is slowly fading at the receiver correlator output. That implies that the correlations between the received PRN codes and the generated local replica maintain all their properties.

Moreover, since all the echoes are delayed by at least the signal symbol duration at the receiver antenna output -which is the chip duration, T_c, in this case -, their contribution at the prompt correlator output is about 0 because the PRN codes waveforms are assumed to conserve their ideal properties. This means that at the correlator output the signal can be modeled as:

$$r_l[k] = \int_{(k-1)T}^{kT} c_1(t) \cdot s_l'(t) \cdot c_{PRN\,i}(t) \cdot e^{j\hat{\theta}(t)} dt \tag{3-28}$$

And this expression is exactly the same as the expression for a signal transmitted through a frequency non-selective channel at the receiver correlator output. Figure 3-8 illustrates the mathematical model of a frequency non-selective channel [PROAKISg, 2001]:

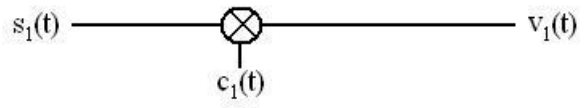

Figure 3-8: Mathematical model for a frequency non-selective channel [PROAKISg, 2001]

The mathematical model of Figure 3-8 has to be interpreted as explained next. This model represents the received LOS signal followed by echoes of the transmitted signal. In the case of a frequency non-selective channel, the time between the reception of the LOS signal and the reception of the last echo is smaller than the signal symbol duration, T, where T is equal to T_c. Therefore, in this case, the contribution of all the echoes and the LOS signal to the received signal amplitude and carrier phase at the receiver antenna output is only modelled by a complex coefficient, $c_1(t)$, which is multiplied by the signal at the transmitter antenna input.

From equation (3-28) and Figure 3-8, it can be observed that both channels have the same equivalent mathematical model if this model is used for demodulation performance analysis. Nevertheless, as it has already been said before, this statement is only true when the channel is slowly fading at the receiver correlator output.

To sum up, if the transmission channel is slowly fading at the receiver correlator output, regardless of the channel coherence bandwidth, the frequency non-selective mathematical model can be correctly selected and implemented in the simulator if this model is used for demodulation performance analysis.

Finally, in the next subsection, a brief discussion about the increment of the signal bandwidth on the modeling of the mathematical model of the transmission channel, its impact on the modeled transmission channel and the length associated to the new code which should be developed is presented.

3.2.2.2.2.1. Signal bandwidth versus code length

We discuss in this section the options for keeping valid the previously presented transmission channel mathematical model while designing a signal with larger bandwidth, where the only option considered to increase the signal bandwidth consists in increasing the signal chip rate; the introduction of higher BOC modulations is not analyzed. We note that whereas a signal with larger bandwidth has its tracking performance fairly improved, a larger signal also leads to a higher possibility of having a frequency-selective channel.

During the presentation of the frequency-selective and frequency non-selective channel, we have shown that one of the fundamental elements determining this channel characteristic is the signal bandwidth. A larger bandwidth increases the probability of the channel of being frequency-selective, and this channel characteristic combined with the no slowly fading characteristic results into a channel which can harm the demodulation. In fact, if the number of main echoes, L, modeling the frequency-selective channel is large enough, we could find inter-symbol interference at the receiver's demodulator. And this phenomenon will result into an unsatisfying demodulation performance. Therefore, we need to keep valid the previously presented transmission channel mathematical model.

Nevertheless, we still want a signal with a larger bandwidth since it is very beneficial from the tracking point of view due to its narrower autocorrelation peak which will allow a more accurate tracking process. Moreover, the increase of the PRN code chip rate will reduce the negative impact of the interferences on the demodulation, acquisition and tracking performance and will also reduce the negative impact of the multipath on the tracking performance. Note that we are talking about the impact of the multipath component on the tracking performance but the mathematical model of the transmission channel presented in this chapter is only valid from demodulation point of view. Therefore, this impact is no longer commented in this dissertation.

From the two previous paragraphs it can be seen that the ideal desired situation in order to keep valid the previously presented transmission channel mathematical model and in order to increase the original signal tracking performance consists in designing a signal with a larger bandwidth but ensuring that the transmission channel is slowly fading (the slowly fading characteristics allows the separation of each of the main echoes of the frequency-selective channel presented in Figure 3-7 and allows its individual processing, if necessary, using a RAKE receiver for example). Therefore, in the next paragraphs, we inspect the slowly fading

characteristics in order to find a signal design solution which will increase the probability of the channel of being slowly fading.

As said in section 3.2.2.1, a transmission channel is slowly fading if its channel coherence time is larger than the symbol duration, where in the GPS L1C or GALILEO E1 OS case the PRN code duration is equal to the data symbol duration. Moreover, the channel coherence time depends on the signal frequency, which cannot be changed, and on the receiver speed, which does not depend on the signal design. Therefore, one possible signal design solution to improve the probability of the channel of being slowly fading could be to keep the same code length in number of chips although the chip rate has been increased (increase of the signal bandwidth). This implies a shorter PRN code period length in time and thus a higher possibility that the channel is slowly fading. Nevertheless, this also implies a shorter data symbol duration leading to the decrease of the available E_b/N_0 at demodulation process (more influence of the noise). One variation of this solution could be to repeat several times the same PRN code inside a data symbol. This action increases the previously shortened coherent integration time to the original value and thus reduces the variance of the noise to the original value as well.

This last proposed solution of increasing the chip rate and repeating the code period in order to keep valid the previously presented transmission channel mathematical model is interesting from the demodulation point of view for the following reason: we increase the probability of the transmission channel being slowly fading for a PRN code period and at the same time we keep the noise influence at the same levels as the values before increasing the signal bandwidth. However, we could also implement another signal design solution where we will have a PRN code with a larger number of chips and thus we will have a larger improvement of the tracking performance because the impact of the interferences and the multipath will be reduced. A possible example of this solution is a signal having a PRN code with the same duration in time than the PRN code before increasing the signal bandwidth (and thus larger number of chips). Nevertheless, this solution also has some drawbacks: first, the degradation of the acquisition time performance and second, that the transmission channel still needs to be slowly fading in order to guarantee satisfying acquisition and tracking processes (note that a longer code lead to a higher possibility of the channel of being no slowly fading). Consequently, we can see that there exists a compromise in the choice of the PRN code period in time.

3.2.2.3. Selection of the mathematical channel model

The mathematical model selected to simulate the transmission channel in a mobile environment and to reproduce reliably the correlator outputs used for bits demodulation purposes is the model defined by Perez-Fontan [PEREZ-FONTAN et al, 2001]. This model has been chosen because it is the most complete model that is fulfilling the dissertation requirements.

The implementation of this model into the frequency-selective and frequency non-selective models previously described is simply the determination of the distribution of complex coefficients $c_n(t)$. This determines how the contribution of the received echoes and the LOS signal to the amplitude and the phase of the received signal are modeled. The distribution of the $c_1(t)$ coefficient, the coefficient associated to the LOS signal, and the distributions of the other $c_n(t)$ coefficients, the coefficients associated to the main echoes of a frequency-selective channel, are not exactly the same. The $c_1(t)$ coefficient contains the LOS signal which can have a very large power with respect to the other echoes whereas the $c_n(t)$ coefficient only

model echoes with about the same power. Therefore, $c_1(t)$ models the statistically sum of a possible large value with relative compared small values and $c_n(t)$ models the statistically sum of about similar values. Another difference between $c_1(t)$ and any $c_n(t)$, and between two coefficients $c_n(t)$ as well, is a correction factor of amplitude. The reason is explained in annex H.2.

In this model, Perez-Fontan divides the distribution of the complex coefficient, $c_1(t)$ into 3 different elements depending on their rate of variation, which is the rate at which they affect the received signal power at the antenna output. These 3 elements are: the very slow variations of the direct signal due to the shadowing or blockage effects, the slow variations of the direct signal due to the different degrees of shadowing of the same obstacle, and the fast variations or fading of the received signal due to the multipath [PEREZ-FONTAN et al, 2001].

In the following subsections the different types of variations, their mathematical expressions as well as the final mathematical model of $c_1(t)$ are presented.

Finally, the $c_n(t)$ coefficients only model the contribution of the multipath. Therefore, the coefficients $c_n(t)$ are represented by a Rayleigh Distribution (see section 3.2.2.3.3). The $c_n(t)$ coefficients are not used to represent the propagation channel mathematical model of this dissertation (see section 3.2.2.2.2).

3.2.2.3.1. LOS very slow variations

The very slow variations of the received signal are due to the very slow variations of the LOS signal. And these variations are generated by different significant shadowing conditions. Indeed, the LOS signal between the satellite and the mobile may be blocked by different types of obstacle, e.g., buildings, trees or nothing at all.

These LOS very slow variations are different significant attenuations of the LOS signal. They are mathematically represented by a system of 3 states in the Perez-Fontan model [PEREZ-FONTAN et al, 2001]. The states are named as:

- S1: LOS conditions
- S2: Moderate shadowing conditions
- S3: Deep shadowing conditions

The 3 states representing the LOS very slow variations determine the distribution of the complex coefficient $c_1(t)$ in a very different way than the LOS slow variations element and the multipath component do. These last two elements define the distribution of these coefficients, the contribution of the LOS signal and the received echoes to the total received signal, whereas the LOS very slow variations element determines the parameters of the distributions defined by the two other elements. In fact, the LOS very slow variations element can be defined as a system or signal state, and depending on which state the signal is, the parameters defining the distribution of the complex coefficient $c_1(t)$ vary. Therefore, despite its name, the LOS very slow variations element also determines the contribution of the multipath to the received signal.

The transitions between the 3 states modeling the LOS very slow variations element are modeled as a first-order Markov chain for which the mobile remains in the same state for a time period equal to the time required by the mobile to cross the state length (L_{Frame}). The state length values are given in [PEREZ-FONTAN et al, 1998]. When the state length has

been crossed, the state transition probabilities are used to determine whether the mobile changes of state or remains in the same one. The transition probabilities are stored into the State Transitions Probability Matrix, [P], where the element P_{ij} indicates the probability of going from the state *i* to the *j* one. Figure 3-9 illustrates the transitions of states:

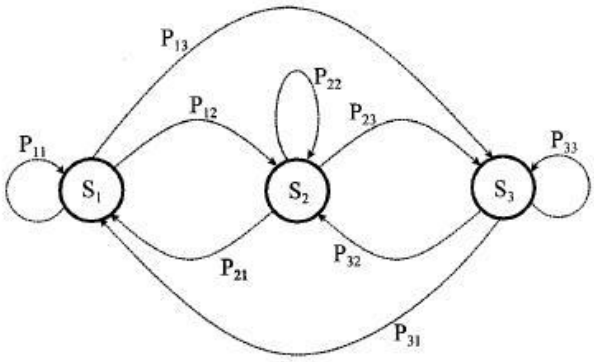

Figure 3-9: First-order Markov chain for a 3 states model representing the LOS very slow variations [PEREZ-FONTAN et al, 1998]

One remark to make about this 3 states model is the transition between states. If the model changes of state, the values of the parameters which distribution is defined by the LOS slow variations and the multipath component vary. Nevertheless, the value of the modeled coefficient $c_1(t)$ does not change abruptly. There is a continuous transition from the values of the previous state of the model to the values of the new state of the model. This transition is called the transition region.

3.2.2.3.2. LOS slow variations

The slow variations of the LOS signal are generated by small-scale changes in the attenuation due to the shadowing of a same blocking obstacle. In other words, the slow variations are the shadow variations suffered by the LOS signal when the mobile travels in the shadow of the same obstacle. One example of this shadow variation is the different shadowings produced by the leaf and branch densities of a group of trees.

The LOS slow variations are mathematically modeled as a log-normal distribution of the signal power. The parameter defining how fast the log-normal distribution varies is the correlation distance (L_{Corr}). This means that the evolution in time of the signal power can be modeled by generating independent log-normal samples separated by the channel correlation distance (L_{Corr}). Nevertheless, since the multipath component has a variation faster than the LOS slow variations component, in order to correctly model the signal transmission through a mobile channel, the simulator needs to generate input signal samples at a rate faster than the rate imposed by the channel correlation distance (L_{Corr}). And these new generated input samples have also to have a log-normal distribution. Moreover, they have to be correlated with the two initial independent log-normal samples in order to model the log-normal evolution of the signal power with time. The correlation distance values are given in [PEREZ-FONTAN et al, 1998] for different types of environments in the S-Band.

To sum up, the contribution of the LOS slow variations on the complex coefficient $c_1(t)$ is to introduce a log-normal distribution on the signal power. And the values of the parameters defining the log-normal distribution depend on the system state {S1, S2, S3} at which the signal is, where the signal state models the LOS very slow variations.

3.2.2.3.3. Multipath

The third and last mathematical model element is the multipath component. The multipath component has already been presented in subsection 3.2.2.1 and is responsible for the fastest variations suffered by the received signal power. This effect is called fading as has been described in the same subsection.

The mathematical model representing this multipath component is a complex number with an amplitude modeled as a Rayleigh distribution and a phase modeled as a uniform distribution [PROAKISg, 2001].

Therefore, the contribution of the multipath component on the complex coefficient $c_1(t)$ (and on the complex coefficients $c_n(t)$ where it is the only contribution), is to introduce a Rayleigh distribution on the signal amplitude and uniform distribution on the signal phase. And the values of the parameters defining the Rayleigh distribution depend on the system state {S1, S2, S3}, where the signal state models the LOS very slow variations.

3.2.2.3.4. Loo Distribution

In subsections 3.2.2.3.2 and 3.2.2.3.3, the LOS slow variations as well as the multipath component have been described, and a mathematical model has been provided for each one. However, the mathematical model of the signal envelope of these two elements can be grouped into one more compact model: a random variable representing the received signal envelope which follows a Loo distribution [LOO and BUTTERWORTH, 1998]. A Loo distribution represents the distribution of the sum between a log-normal variable for the LOS signal and a Rayleigh variable for the multipath component.

The signal envelope probability density function, p(r), of a Loo distribution is shown below.

$$p(r) = \frac{r}{b_0\sqrt{2\pi d_0}} \int_0^\infty \frac{1}{z} \exp\left[-\frac{(\ln z - \mu)}{2d_0} - \frac{r^2 + z^2}{2b_0} \right] \cdot I_0\left(\frac{rz}{b_0}\right) dz \tag{3-29}$$

$$\alpha = 20\log_{10}\left(e^{\mu}\right) \tag{3-30}$$

$$\Psi = 20\log_{10}\left(e^{\sqrt{d_0}}\right) \tag{3-31}$$

$$MP = 10\log_{10}\left(2b_0\right) \tag{3-32}$$

Where:

- $I_0(x)$: modified Bessel function of order 0 evaluated at x
- α: mean of the log-normal distribution (dB)
- Ψ: standard deviation of the log-normal distribution (dB)
- MP: Average Multipath Power (dB) with respect to a LOS signal having a power equal to 0dB

Therefore, α, Ψ and MP values define the signal envelope of a Loo distribution, or in other words, the distribution of $|c_1(t)|$.

Finally, note that the received signal carrier phase of the addition between the LOS slow variations component and the multipath component is not presented since this distribution depends on the phase of the LOS signal. Nevertheless, the received signal carrier phase probability density function has been determined in [LOO and BUTTERWORTH, 1998].

3.2.2.3.5. Final mathematical model – Non-frequency selective channel

The final mathematical model used in this dissertation is based on a frequency non-selective channel mathematical model. Therefore, in our case, the single parameter to adjust or modify is the complex coefficient of the LOS signal, $c_1(t)$, presented in section 3.2.2.2.2. This means that the Perez-Fontan model or any other channel model such as Rayleigh, Lutz, etc are represented only with this coefficient.

In this research work, the random variable modeling the complex coefficient $c_1(t)$ is defined by the Perez-Fontan model [PEREZ-FONTAN et al, 1998]. This variable follows a Loo distribution which models the LOS slow variations and the multipath component. The parameters defining the Loo distribution are determined by the state of the model, which are the LOS very slow variations. Note that despite its name, the LOS very slow variations element also determines the multipath contribution to the received signal. This means that 3 different sets of Loo parameters are defined for each transmission channel.

The main difference between each state is the value of the received LOS signal power. And this difference of received LOS signal power can be represented by a variation of the Loo parameters, α, Ψ and MP. Therefore, each one of the 3 states modeling the LOS very slow variations element on the received signal distribution is represented by generating the Loo random variable with different parameters, α, Ψ and MP. Finally, the state transitions probability matrix altogether with the frame length and the mobile speed is in charge of controlling each Loo random variable state generation since they control the system state. Different values of these parameters can be found in [PEREZ-FONTAN et al, 1998] [PEREZ-FONTAN et al, 2001].

The general mathematical model of an equivalent baseband received signal after propagation through a frequency non-selective channel at the receiver antenna output is obtained from the multiplication of the mathematical expression of a GNSS transmitted signal at the transmitter antenna input (equation (A-4)) with the complex coefficient $c_1(t)$. Note that it has been assumed that the subcarriers are perfectly removed. Remember that GPS L1C and GALILEO E1 are modulated with a TMBOC and a CBOC respectively. The equivalent noiseless baseband complex envelope signal model after propagation through a frequency non-selective channel at the receiver antenna output, $v_l(t)$, is given below:

$$v_l(t) = A \cdot c_1(t) \cdot \left[d(t) \cdot c_d(t) + c_p(t) \right] \cdot e^{j\theta(t)} \tag{3-33}$$

Where:

- A: Incoming signal amplitude at the receiver antenna output
- d(t): Navigation data waveform
- θ(t): Incoming useful signal carrier phase at the receiver antenna output after propagation through an AWGN channel.
- $c_d(t)$: data channel PRN code waveform
- $c_p(t)$: pilot channel PRN code waveform
- $c_1(t)$: Complex random variable representing the frequency non-selective channel

The complex coefficient $c_1(t)$ can be expressed as:

$$c_1(t) = \left| c_1(t) \right| e^{j\theta_{c_1}(t)} \tag{3-34}$$

Where:

- $|c_1(t)|$: Models the contribution of all the received echoes to the LOS signal amplitude at the receiver antenna output
- $\theta_c(t)$: Models the contribution of all the received echoes to the LOS signal carrier phase at the receiver antenna output

Integrating $c_1(t)$ to the signal amplitude and phase, the signal expression can be modelled as:

$$v_l(t) = A \cdot |c_1(t)| \cdot \left[d(t) \cdot c_d(t) + c_p(t) \right] \cdot e^{j(\theta(t)+\theta_c(t))} \quad (3\text{-}35)$$

$$v_l(t) = A'(t) \cdot \left[d(t) \cdot c_d(t) + c_p(t) \right] \cdot e^{j\theta'(t)} \quad (3\text{-}36)$$

Where:

- A'(t): New incoming signal amplitude at the receiver antenna output
- θ'(t): New incoming signal carrier phase at the receiver antenna output

Additionally, the mathematical model of a frequency-selective channel can be found in annex H.2.

3.2.2.4. Additive white Gaussian noise

The mobile channel model implemented in this dissertation also contains the AWGN and, as well as in the AWGN channel model case, its influence is expressed through the C/N_0 figure of merit. Nevertheless, in this case, C does not define exactly the received useful signal power at the antenna output. In this case, the C parameter defines the power of the LOS signal when no obstacle is in its trajectory; no shadowing is affecting the LOS signal. This means that the C/N_0 figure of merit does not take into account the loss of part of the useful signal power due to $c_1(t)$ coefficient which is the coefficient modeling the multipath component and the LOS slow variations.

3.2.3. Impact of the mobile channel over the carrier tracking process

Two of the main characteristics introduced by a mobile channel with respect to an AWGN channel are the variation of the received signal power and the even much faster variation of the received signal carrier phase.

The variation of the received signal power due to the LOS signal blockage and/or shadowing can lead to situations where the power of the pilot channel is not strong enough to allow the PLL to be locked. This means that the carrier phase tracking cannot be always achieved in mobile channels when the LOS signal is significantly attenuated.

The much faster variation of the received signal carrier phase is caused by the additional phase introduced by the mobile channel (the addition of the multipath component to the LOS signal). In fact, as it was observed in equation (3-36), the received signal carrier phase is the sum of the own signal carrier phase, $\theta(t)$, and the phase shift introduced by the mobile channel, $\theta_c(t)$.

The own signal carrier phase is the phase due to the signal dynamics defined in annex A.3.3.2. This phase is the sum of a constant initial term plus some time variable terms, where each time variable term depends on the signal dynamics. Remember that the signal dynamics are reduced here to the difference of motion between the satellite and the receiver (impact of

clock is much lower than noise and motion). Therefore, there is a term related to the velocity, another to the acceleration, jerk, etc. And, as its name indicates, this signal carrier phase, θ(t), only depends on the transmitter motion, on the receiver motion and on the distance between them. Therefore, this phase is the phase of an AWGN channel since this channel does not introduce any additional phase contribution to the received signal carrier phase at the antenna output.

However, the mobile channel does introduce an additional phase. This phase is generated by the addition of the multipath component to the LOS signal. Therefore, this phase is modeled as the sum of a random variable following a uniform distribution within [0, 2π[(this uniform distribution represents the contribution of the addition of all the received echoes) plus a constant term with 0 radians of phase representing the LOS signal (the dynamics of the LOS signal are modeled by θ(t) as was previously said). And the two terms of the addition are weighted by the multipath component envelope to the LOS signal envelope ratio.

Therefore, it can be concluded that the phase introduced by the channel varies depending on the multipath component rate, and this means that the phase variation rate is bounded by the channel coherence time. Moreover, it also means that the PLL phase tracking process, when the power of the pilot channel is high enough, can still be conducted even if it is more difficult in comparison with the case of an AWGN channel.

In fact, the discriminator of a PLL estimates the received signal phase with a time step equal to the coherent integration time T_i seconds. Nevertheless, the PLL has to provide signal phase estimation samples for all the signal samples during all this T_i period. Usually, the PLL generates a series of samples following a linear evolution between the two estimated phases at the period edges. This means that if the signal phase does not follow a linear evolution, the phase tracking is not accurate. Therefore, the solution is to reduce the coherent integration time in order to estimate the signal phase onto periods where the signal phase can be approximated by a line. Figure 3-10 illustrates this phenomenon.

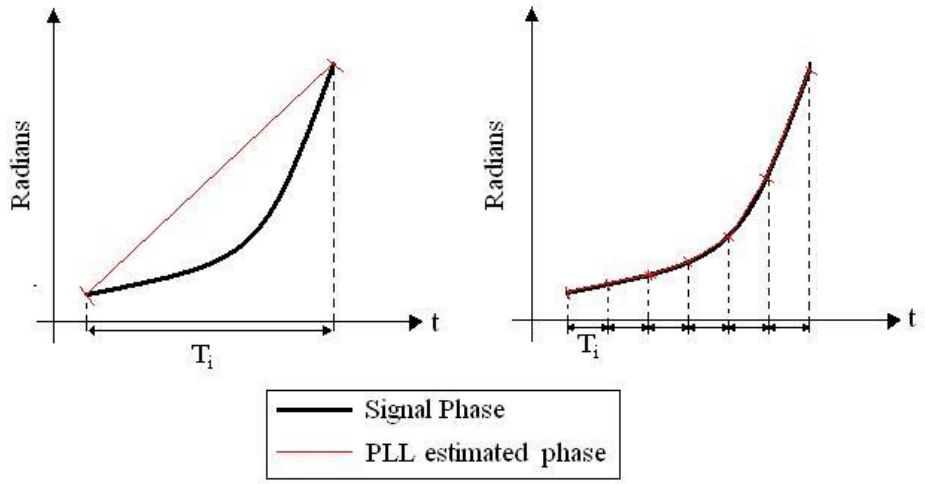

Figure 3-10: PLL phase estimation evolutions for different integration times (T_i)

However, the drawback of decreasing the integration time is the increase of the influence of the thermal noise over the phase estimation process. Therefore, the integration time cannot be indefinitely decreased. This statement is true for all the discriminators explained in annex A.3.3, even for the Q discriminator although the effect of the coherent integration time T_i on its tracking performance is lower than the effect for the other discriminators. In fact, the

mathematical expression of the carrier phase estimation error power due to the thermal noise for the Q discriminator, equation (3-8), does not contain the T_i term and thus the reader must incorrectly assume that the coherent integration does not have any effect on the power of this error. However, this expression is only valid for high C/N_0 values, because for low C/N_0 values the mathematical development used to obtain the expression is not accurate enough. In fact, for low C/N_0 values, it has been tested that the longer the T_i is, the lower the power of the carrier phase estimation error due to the thermal noise is.

For an AWGN channel, the problem is reduced to the impact of the thermal noise since the evolution of the signal phase is not fast enough to require small integration times. Therefore, the PLL can use very long integration times assuming that the signal has a pilot channel. However, for a mobile channel, the signal phase varies a lot and very fast. Therefore, it can happen that the PLL cannot track the signal phase in this channel even when the power of the pilot channel is high enough. In fact, the evolution of the signal phase due to the obstacles is determined by the channel coherence time. Therefore, for channel coherence time values lower than the PLL integration time, the PLL performance is degraded and it may not work.

Finally, since it cannot be assured that the PLL is able to track the signal phase, other alternatives have to be analyzed. In this research work, the performance of the channel estimation method is analyzed since it is the major technique employed in the telecommunication field in order to estimate the phase of the useful signal received through a mobile channel. In fact, this carrier phase estimation method does not require a signal carrier power as strong as required by the PLL. Two different channel estimation methods are analyzed. The first one is the ideal estimation channel which provides the perfect performance bounds which are attainable with the estimation channel technique. The second one is really a carrier phase estimator: the average of the different received carrier phases.

The two next sections explain the PLL and channel estimation methods in mobile channels.

3.2.3.1. Traditional Phase Locked Loop

A quick summary of the PLL characteristics is given in this section. The PLL analyzed is the typical PLL with an integration time of 20ms, an equivalent noise bandwidth, B_L = 10Hz, a 3rd order filter with coefficients defined in [STEPHENS and THOMAS, 1995] and a coherent or Q discriminator. The only source of error affecting the carrier phase tracking process of the PLL are the thermal noise and the dynamic stress error as justified in section 3.1.3.2.

Moreover, a phase lock detector has been implemented in order to know when the PLL is locked. This phase lock detector determines from the I and Q channels if the PLL is still able to track the signal. If the phase lock detector considers that the PLL is no longer tracking the signal, the phase lock detector flags a loss of lock event. Its principle is detailed below.

3.2.3.1.1. Phase Lock Detector

The phase lock detector model applied is defined in [SPILKER and ASHBYd, 1996]. This phase lock detector uses the normalized estimate of the cosine of twice the carrier phase ($C2\phi_k$) given as follows:

$$C2\phi_k = \frac{NBD_k}{NBP_k} \tag{3-37}$$

With:

$$NBD_k = \left(\sum_{i=1}^{M} I_i\right)_k^2 - \left(\sum_{i=1}^{M} Q_i\right)_k^2 \quad (3\text{-}38)$$

And:

$$NBP_k = \left(\sum_{i=1}^{M} I_i\right)_k^2 + \left(\sum_{i=1}^{M} Q_i\right)_k^2 \quad (3\text{-}39)$$

Where:

- I_i: I channel signal sample. One sample for each integration period.
- Q_i: Q channel signal sample. One sample for each integration period.
- M: Number of samples to accumulate before deciding if the PLL has lost its lock

The value of M is chosen as 20 in this analysis.

Finally, the normalized estimate of the cosine of twice the carrier phase is compared to a threshold before deciding if the PLL has lost its lock. If the $C2\phi_k$ is lower than the threshold the loss of lock happens. The threshold value chosen in this study is 0.4.

3.2.3.2. Channel estimation

The objective of a channel estimation method is to allow the coherent demodulation of the received signal. In order to do so, a general channel estimation method implements two actions. The first one is to estimate the phase and the amplitude introduced by the channel; in the previous defined mobile channel mathematical model, this has been denoted as the coefficient $c_1[k]$. The second one is to remove the carrier phase of the received signal in order to coherently demodulate the received signal.

The estimated channel is called $c_{est}[k]$, and it can be modeled as:

$$c_{est}[k] = \left|c_{est}[k]\right| \cdot e^{j\theta_{est}[k]} \quad (3\text{-}40)$$

Therefore, taking the discrete expressions of equations (3-33) and (3-35), and assuming that the channel is frequency non-selective and slowly fading, the ideal mathematical expression of the signal at the RF/IF block output before equalization is:

$$w_l[k] = A \cdot c_1[k] \cdot \left[d_m[k] \cdot c_d'[k] + c_p'[k]\right] \cdot e^{j\theta(t)} \quad (3\text{-}41)$$

And, the estimation of the channel is:

$$c_{est}[k] = c_1[k] \quad \Rightarrow \quad \begin{cases} \left|c_{est}[k]\right| = \left|c_1[k]\right| \\ \theta_{est}[k] = \theta_c[k] \end{cases} \quad (3\text{-}42)$$

Where:

- $d_m[k]$: The navigation data is constant for P data channel PRN code periods, where $P \geq 1$. The sub-index m determines in which group of P data channel PRN code periods the signal is. And thus, sub-index m determines the data value.

Nevertheless, in reality, the channel estimation method assumes that the entire received signal phase is introduced by the channel. Therefore, the initial signal phase and the phase

introduced by the signal dynamics are considered as part of the channel and thus they are also estimated. And the same can be said for the signal amplitude: the estimated channel amplitude includes the transmitted signal amplitude in addition to the amplitude modification introduced by the channel. Therefore, in this dissertation, it is assumed that the signal is transmitted with a phase equal to zero and with an amplitude equal to 1, which means that the received signal phase and the received signal amplitude are completely introduced by the channel.

The equivalent channel complex coefficient, c[k], to be estimated can thus be modelled as:

$$c[k] = A \cdot c_1[k] \cdot e^{j\theta[k]} \tag{3-43}$$

$$c[k] = A \cdot \left|c_1[k]\right| \cdot e^{j(\theta[k]+\theta_c[k])} \tag{3-44}$$

Therefore, the real channel estimation to be made at the receiver RF/IF block output is:

$$c_{est}[k] = c[k] \quad \Rightarrow \quad \begin{cases} \left|c_{est}[k]\right| = A \cdot \left|c_1[k]\right| \\ \theta_{est}[k] = \theta[k] + \theta_c[k] = \theta'[k] \end{cases} \tag{3-45}$$

One last remark about the estimation part is that there are different techniques to estimate the c[k] coefficient, where each technique has its particularities and has its specific performance.

Once the estimation is made, the removal of the received signal carrier phase which allows the coherent demodulation consists in multiplying the received signal with the conjugate of the c[k] estimation [PROAKISg, 2001]. The mathematical expression is given in equation (3-48) but before the influence of the thermal noise is inspected.

Adding thermal noise to equation (3-41), the discrete equivalent baseband complex envelope signal at the RF/IF block output before equalization can be modeled as:

$$w_l[k] = c[k] \cdot s'[k] + n[k] \tag{3-46}$$

$$s'[k] = d_m[k] \cdot c_d'[k] + c_p'[k] \tag{3-47}$$

Where:

- n[k]: Additive colored Gaussian noise after RF/IF filtering

Therefore, the discrete equalized equivalent baseband complex envelope signal at the RF/IF block output, $u_l[k]$, is:

$$u_l[k] = \overline{c_{est}}[k] \cdot w_l[k] = \overline{c_{est}}[k] \cdot \left(c[k] \cdot s'[k] + n[k]\right) \tag{3-48}$$

$$u_l[k] = \left|c[k]\right|^2 \cdot s'[k] + \overline{c_{est}}[k] \cdot n[k] \tag{3-49}$$

For this last expression some remarks can be done.

First, it can be observed that the multiplication of the received signal by the amplitude of the estimated channel does not improve the demodulation performance since the noise and the useful signal are multiplied by the same term. However, if a channel code is implemented, the amplitude equalization improves the decoding performance due to the Maximum-Likelihood (ML) demodulation [PROAKISg, 2001].

Second, as it was previously said, the mathematical expressions of the channel estimation and the removal of the received signal carrier phase which are presented above are the mathematical expressions of an ideal case, where the entire useful signal power can be used to demodulate the signal. However, the actual channel estimation techniques do not provide perfect phase and amplitude estimations due to the presence of thermal noise. Therefore, in

order to reduce the noise influence over the estimation, a channel estimation technique uses several samples to generate the c[k] estimation, e.g. averaging the signal and the noise.

In fact, for the ideal case, it is assumed that the receiver is able to estimate instantaneously the true channel equivalent impulse response with no error for each received signal sample. However, in reality, due to the presence of thermal noise, the receiver uses several signal samples to estimate the channel equivalent impulse response affecting these samples. In other words, the receiver divides the time in periods, uses the samples inside one of these time periods to generate a channel estimation, and uses this generated channel estimation to equalize the samples of this time period (the samples employed to generate this channel estimation). But, note that this estimation will never be perfect because of the thermal noise presence in the channel.

In this research work, two channel estimation techniques have been analyzed. The first one is the ideal channel estimation case and the second one is a very simple channel estimation technique which is in reality a carrier phase estimator for an AWGN channel. These two techniques are presented next.

Nevertheless, before examining the two channel estimation techniques, a more detailed mathematical model of the baseband complex envelope signal at the RF/IF block output, $w_l[k]$, is presented using equation (3-46):

$$w_l[k] = b_d'[k] + b_p'[k] + n[k] \tag{3-50}$$

$$\begin{cases} b_p'[k] = c[k] \cdot c_p'[k] \\ b_d'[k] = c[k] \cdot d_m[k] \cdot c_d'[k] \end{cases} \tag{3-51}$$

Where:

- b_d'[k]: Equivalent baseband complex envelope data channel signal at the RF/IF block output
- b_p'[k]: Equivalent baseband complex envelope pilot channel signal at the RF/IF block output
- c_d'[k]: Filtered data channel PRN code waveform
- c_p'[k]: Filtered pilot channel PRN code waveform
- d_m[k]: Navigation data waveform
- c[k]: Equivalent channel complex coefficient et epoch k
- n[k]: Additive colored Gaussian noise

3.2.3.2.1. Ideal pilot channel estimation

In order to determine the demodulation performance limit or lower bound achieved by estimating the pilot channel, the ideal pilot channel estimation case has to be studied. Therefore, in order to reproduce the pilot channel ideal estimation case, it is assumed that the receiver is able to provide an exact instantaneous estimate of the amplitude and phase introduced by the channel for each received signal sample.

3.2.3.2.2. Samples averaging channel estimation technique

The pilot channel estimation technique proposed in this study is very simple since it is in reality a carrier phase estimator for an AWGN channel. Therefore the BER, WER and EER obtained for this technique could be considered as demodulation performance upper bounds.

The proposed estimation consists in averaging all the pilot samples of a time period in order to obtain an average value which represents the channel impulse response of this period. Nevertheless, the pilot samples can only be used if the pilot channel PRN code has been removed because the PRN code distorts the channel impulse response estimation. Moreover, the data channel has also to be removed from the calculation for the same reason.

The mathematical expression of the proposed channel estimation is given below:

$$c_{est}[n] = c_{avg}[n] = \left|c_{avg}[n]\right| \cdot e^{j\theta_{avg}[n]} \tag{3-52}$$

$$c_{est}[n] = \frac{1}{N} \sum_{k=n\cdot N}^{(n+1)\cdot N-1} w_l[k] \cdot c_p[k] \tag{3-53}$$

Where:

- $c_{est}[n]$: Channel impulse response estimation of period of time n
- $w_l[k]$: Baseband complex envelope signal at the RF/IF block output
- $c_p[k]$: Pilot channel PRN code waveform
- N: Number of samples used to average the instantaneous channel impulse response

If we inspect more closely equation (3-53):

$$c_{est}[n] = \frac{1}{N} \sum_{k=n\cdot N}^{(n+1)\cdot N-1} \left(b_p'[k] + b_d'[k] + n[k]\right) \cdot c_p[k] \tag{3-54}$$

Where:

- b_p‘[k]: Equivalent baseband complex envelope pilot channel signal at the RF/IF block output
- b_d‘[k]: Equivalent baseband complex envelope pilot channel signal at the RF/IF block output
- n[k]: Additive colored Gaussian noise

Equation (3-54) can also be expressed as:

$$c_{est}[n] = \frac{1}{N} \sum_{k=n\cdot N}^{(n+1)\cdot N-1} \left(c_p'[k] \cdot c_p[k] \cdot c[k] + d_m[k] \cdot c_d'[k] \cdot c_p[k] \cdot c[k] + n[k] \cdot c_p[k]\right) \tag{3-55}$$

Where:

- c[k]: Equivalent channel complex coefficient et epoch k
- c_p’[k]: Filtered pilot channel PRN code waveform
- c_d’[k]: Filtered data channel PRN code waveform
- d_m[k]: Navigation data waveform

Assuming that the code delay estimation has been perfectly achieved and that no significant distortion is introduced by the RF/IF filter on the pilot channel PRN waveform, the product $c_p[k]\cdot c_p'[k]$ is equal to 1.

Moreover, if the channel is assumed to be slowly fading during the channel estimation time -c[k] is assumed as constant during all the N samples- and the time used to estimate the channel impulse response is R times the data and/or pilot PRN code period -where R is integer- equation (3-55) can be simplified as it is shown below.

$$c_{est}[n]=\frac{1}{N}\sum_{k=n\cdot N}^{(n+1)\cdot N-1}\left(c[k]+n[k]\cdot c_p[k]\right) \tag{3-56}$$

And from equation 3.43 and 3.44:

$$c_{est}[n]=\frac{1}{N}\sum_{k=n\cdot N}^{(n+1)N-1}\left(A\cdot c_1[k]\cdot e^{j\theta[k]}+n[k]\cdot c_p[k]\right) \tag{3-57}$$

$$c_{est}[n]=\frac{1}{N}\sum_{k=n\cdot N}^{(n+1)N-1}\left(A\cdot\left|c_1[k]\right|\cdot e^{j(\theta[k]+\theta_c[k])}+n[k]\cdot c_p[k]\right) \tag{3-58}$$

Where:

- $|c_1(t)|$: Models the contribution of all the received echoes to the LOS signal amplitude at the receiver antenna output
- $\theta_c(t)$: Models the contribution of all the received echoes to the LOS signal carrier phase at the receiver antenna output
- A: Incoming signal amplitude at the receiver antenna output
- $\theta(t)$: Incoming signal carrier phase at the receiver antenna output

Therefore, the discrete equivalent baseband complex envelope signal at the RF/IF block output equalized by the channel estimation of the period of time n, $u_l[k,n]$, can be modeled as:

$$u_l[k,n]=w_l[k]\cdot\left|c_{avg}[n]\right|\cdot e^{-j\theta_{avg}[n]} \tag{3-59}$$

And, the signal at the data channel correlator output, $ru_l[m,n]$ is:

$$ru_l[m,n]=\frac{1}{N}\sum_{k=m\cdot Q}^{(m+1)Q-1}w_l[k]\cdot c_d[k]\cdot\left|c_{avg}[n]\right|\cdot e^{-j\theta_{avg}[n]} \tag{3-60}$$

$$ru_l[m,n]=\frac{1}{N}\sum_{k=m\cdot Q}^{(m+1)Q-1}\left(b_p'[k]+b_d'[k]+n[k]\right)\cdot c_d[k]\cdot\left|c_{avg}[n]\right|\cdot e^{-j\theta_{avg}[n]} \tag{3-61}$$

Where:

- $ru_l[m,n]$: Equivalent baseband complex envelope data channel at the correlator output at epoch m when the signal has been equalized with channel estimation of time period n.
- Q: Number of samples belonging to the same data symbol period.

Note that the signal data symbol duration and the period of time of the channel estimation do not have to be the same. In fact, the period of time of the channel estimation can be longer than the data symbol duration. Nevertheless, if the period of time of the channel estimate is smaller, the channel estimation, $c_{avg}[n]$, is corrupted by the data channel signal since it has

been assumed that the duration of a data symbol is equal to the duration of R PRN code periods -where R is an integer number equal or bigger than 1.

Finally, equation (3-61) can be simplified since it is assumed that the channel is slowly fading at the correlator output; the channel is assumed to be constant for the signal data symbol duration.

$$ru_l[m,n] = \frac{d_m}{N} \sum_{k=m\cdot Q}^{(m+1)Q-1} A \cdot \left|c_1[k]\right| \cdot \left|c_{avg}[n]\right| \cdot e^{j\left(\theta[k]+\theta_c[k]-\theta_{avg}[n]\right)} + n'[m,n] \tag{3-62}$$

$$n'[m,n] = \frac{1}{N} \sum_{k=m\cdot Q}^{(m+1)\cdot Q-1} n[k] \cdot c_d[k] \cdot \left|c_{avg}[n]\right| \cdot e^{-j\theta_{avg}[n]} \tag{3-63}$$

Where:

- d_m: Value of the symbol data at epoch m

One assumption made during the previous mathematical development is that the channel impulse response does not vary during the channel estimation process. In fact, if the channel impulse response evolves during the estimation time, the proposed pilot channel estimation introduces a lot of errors due to the significant difference between the constant estimated channel impulse response and the instantaneous channel impulse response. Therefore, in order to obtain a satisfactory channel estimation, the channel has to be slowly fading with respect to the time used to estimate the channel impulse response. Note that this restriction is equivalent to the imposition of the maximum symbol or bit period for a slowly fading channel as it has been done in section 3.2.2.2.1. Therefore, the maximal amount of time period used to generate this proposed pilot estimation channel is fixed by the mobile speed as it has been done in section 3.2.2.2.1.

The performance of the channel estimation technique proposed along this section is presented in Figure 3-11. This performance is expressed as the standard deviation in degrees of the phase estimation error of this technique as function of the signal C/N_0. The phase estimation error is calculated for an AWGN channel and for the mobile channel mathematical model presented in section 3.2.2.3.5 when the mobile travels at a speed of either 5 km/h or 30 km/h. Besides, this performance is presented individually for the S1 (LOS conditions) and S2 (Moderate Shadowing conditions) of the mobile channel mathematical model.

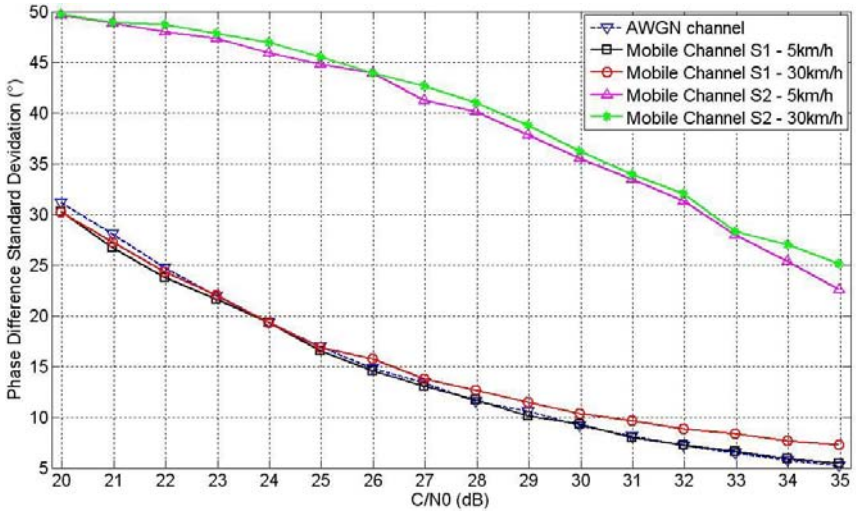

Figure 3-11: Performance of the proposed channel estimation technique

3.2.4. Impact of the mobile channel over the demodulation process

For demodulation performance analysis, the principal difference between a signal received through a mobile channel and a signal received through an AWGN channel is the useful signal amplitude variation since the thermal noise affecting the signal is the same.

Therefore, the main problem introduced by a mobile channel which affects the signal demodulation performance is the C/N_0 decrease. This decrease implies two things. First, the power of the received signal can be so weak that the PLL cannot achieve a lock and thus the signal cannot be demodulated. This is the case for the state S3 (Deep shadowing conditions) of the system, and sometimes, depending on the LOS slow variations, for the state S2 (Moderate Shadowing conditions) of the system. Second, when the PLL is locked, the decrease of C/N_0 implies the obtaining of very noisy channel observations. The definition of a channel observation is given in annex C.1. Obviously, very noisy channel observations are not good for the demodulation process since they hide the received channel information. Nevertheless, if the signal has a channel code implemented and the very noisy channel observations are not consecutive but rather quite separated in time, the channel code should be able to correct the bit errors introduced by these very noisy channel observations. In fact, the decoding of the convolutional codes and the GPS L1C LDPC codes obtain better performance when the bit errors are distributed uniformly with time; or inside a word for a block channel code.

This means that the LOS very slow variations and the LOS slow variations are more harmful than the multipath component since they introduce much more consecutive very noisy channel observations in addition of being responsible for the PLL loss of lock. In fact, the rate of variation of the LOS very slow variations, the LOS slow variations and the multipath component depends on the receiver speed. This means that for a slow speed of the receiver all three components can affect significantly the signal. However, since the evolution of the multipath component is much faster than the other two elements for any receiver speed, the multipath component is less harmful than the other two. This means that the very noisy channel observations introduced by the multipath component are more probable of being randomly distributed inside the signal than the channel observations of the LOS very slow variations component and the LOS slow variations which are more probable to be found consecutively inside the signal.

A solution to improve the demodulation performance is to cancel this correlation between very noisy channel observations introduced by the LOS very slow variations component, by the LOS slow variations component and sometimes by the multipath component. This cancelation of the correlation between consecutives received signal samples is achieved by the introduction of an interleaver block. See section 6.1.4 for a detailed description of an interleaver.

Finally, for transmissions through a mobile channel, it can be concluded that the signals implementing an interleaver will have much better demodulation performance than the signals not having any.

Note that, for all the previous explanations, in this dissertation, it has been assumed that the channel is slowly fading. If this was not the case, the impact of the mobile channel over the demodulation will be more complex. In fact, for a channel which is not slowly fading, the loss of the PRN codes autocorrelation triangular propriety and the loss of PRN codes orthogonal correlation propriety should be studied.

3.2.5. Mobile channel simulator

In this subsection, the simulator implemented to represent the transmission of a GNSS signal through a mobile channel is described. Using the simplified version of the second simulator of an AWGN channel (Figure 3-5), the mobile channel simulator scheme is presented below.

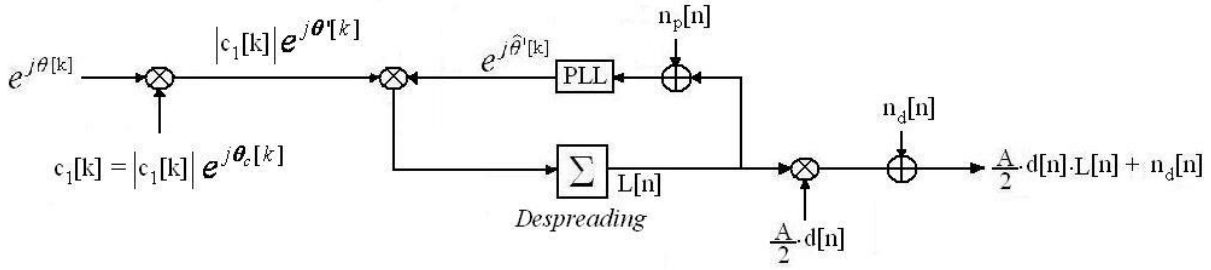

Figure 3-12: Mobile channel simulator scheme

Where:

- $\theta'[k]$: Incoming signal carrier phase
- $\hat{\theta}'[k]$: PLL signal carrier phase estimation
- $c_1[k]$: Loo variable at epoch k
- $n_p[n]$: Pilot channel noise at epoch n.
- $n_d[n]$: Data channel noise at epoch n.
- d[n]: Navigation data at epoch n
- k: epoch at chip sampling rate
- n: epoch integration output rate
- L[n]: Correlator output at epoch n

$$L[n] = \frac{1}{M}\sum_{k=0}^{M-1} e^{j\left(\theta[k]-\hat{\theta}[k]\right)} \tag{3-64}$$

The generation of the $c_1[k]$ coefficient and the generation of the noise is presented in annex H.3.2.

3.2.6. Limitations and validity of the mathematical model of a mobile channel

The main limitation of the presented mathematical model of a mobile channel is that it can only be used to analyze the demodulation process of a transmitted signal. This mathematical model is not valid to be applied for a code delay tracking process. In fact, we are assuming that the estimation of the code delay is perfectly achieved for the values of C/N_0 used in 0 of this dissertation.

Another limitation of the presented mathematical model is the speed at which the receiver can move. We have determined the maximum travel speeds in section 3.2.2.2.1 for each GNSS signal, but it has to be remembered that they are loose bounds useful for a first analysis.

Moreover, this mathematical model of a mobile channel is based on the Perez-Fontan model; more specifically, the $c_1[n]$ coefficient modeling the contribution of the LOS signal and the multipath component to the received signal. The validation of this model is made on the

Perez-Fontan papers where it is literally said "A fairly good agreement between measurements and model outputs has been observed in all cases". Moreover, the Perez-Fontan model is well known and probably is the most used mathematical model of a mobile channel from a demodulation point of view. We have not conducted a field validation of the model for the previously mentioned reasons.

Finally, the equivalence between a non-frequency selective channel and a selective-frequency was based on the implementation of a tapped delay line, where the different taps are delayed by the inverse of the signal bandwidth. However, [PEREZ-FONTAN et al, 2001] also proposes to uses a Poisson distribution to determine the delay of each individual main echo. Therefore, if instead of using the tapped delay line, the proposition made in [PEREZ-FONTAN et al, 2001] is used, the received signal may not only contains the distribution of the coefficient $c_1(t)$. In fact, it could also contain other complex coefficients $c_n(t)$ representing the contribution of the main echoes having a delay smaller than the chip duration.

3.3. Conclusions

In this chapter, two different mathematical models representing two of the most common transmission channels have been presented.

The first mathematical channel model is the Additive White Gaussian Noise channel which models signal transmissions through open sky environments, where an open sky environment is a line-of-sight (LoS) scenario where the only received ray is the direct ray. The mathematical model of an AWGN channel at the receiver antenna output is simply the addition of a white Gaussian noise to the transmitted useful signal. Moreover, this noise is narrow-band after the filtering of the RF/IF block whereas the received useful signal can be assumed as barely affected.

The carrier phase tracking process of a signal transmitted through an AWGN has 4 sources of errors but only the thermal noise and the dynamic stress error are analyzed and taken into account: the other two sources, the Allan deviation noise and the oscillator vibrations, are negligible with respect to the other two sources for the C/N_0 values analyzed in this dissertation. More specifically, the effects of these two sources of error on the tracking performance are given for a PLL implementing a Q discriminator. In fact, the Q discriminator has been chosen as the better discriminator to use in the dissertation tracking and demodulation performance analysis, since the analyzed signals carry a dataless channel and the Q discriminator outperforms the Arctangent2 discriminator for low C/N_0 values.

The PLL loss of lock event has been presented and some preliminary loss of lock threshold values have been calculated taking into account only the previous two sources of error affecting the tracking performance. The PLL loss of lock threshold value for the GPS L1C signal is a C/N_0 of 16.9 dB-Hz when the dynamic stress error is a constant jerk equal to 0 g/s, and the loss of lock threshold value is 21.9 dB-Hz when the constant jerk is equal to 1 g/s. For GALILEO E1 signal, the threshold value is equal to 18.6 dB-Hz for a constant jerk equal to 0 g/s and 23.6 dB-Hz for a constant jerk equal to 1 g/s. For GPS L2C signal, the threshold value is equal to 18.6 dB-Hz for a constant jerk equal to 0 g/s and 22.2 dB-Hz for a constant jerk equal to 1 g/s. For GPS L5 signal, the threshold value is equal to 18.6 dB-Hz for a constant jerk equal to 0 g/s and 22.0 dB-Hz for a constant jerk equal to 1 g/s.

The demodulation process of a signal transmitted through an AWGN channel has two sources of error. The first source is the decrease of the useful signal power due to the tracking

performance. The second source is the corruption of the received signal symbols component due to the thermal noise.

The simulator modeling the transmission of a signal through an AWGN channel has been presented in this chapter, and the generation of all its components has been described.

The second mathematical model is the mobile channel which models the signal transmission through mobile urban environments or indoor environments. The characteristics of a signal received through these types of environments are a LOS signal plus a multipath component, where the total signal power and carrier phase vary with time.

The mathematical model chosen to represent the transmission of a GNSS through a mobile channel is a frequency non-selective channel which is slowly fading at the receiver correlator output. This kind of mathematical model can only be applied for a receiver travelling at speed slower than 171 km/h when the received signal is GALILEO E1 OS and slower than 69 km/h when the received signal is GPS L1C.

The proposed mathematical model consists in multiplying the transmitted signal by a complex coefficient, $c_1(t)$, which models the time evolution of the addition of the LOS signal and the multipath component. The complex coefficient $c_1(t)$ has been modeled by applying the model proposed by Perez-Fontan [PEREZ-FONTAN et al, 1998] [PEREZ-FONTAN et al, 2001]. This model divides the time evolution of $c_1(t)$, the variation of the total signal power and carrier phase, in 3 elements depending on its rate of variation. The LOS slow variations and the multipath component define a signal envelope which follows a Loo distribution [LOO and BUTTERWORTH, 1998], and the LOS very slow variations determines the parameters defining the Loo distribution. The LOS very slow variations are represented as 3 system states, {S1: LOS conditions}, {S2: Moderate shadowing conditions} and {S3: Deep shadowing conditions}, which means that there are 3 different sets of Loo distribution parameters. The final mathematical model adds to the product between the transmitted signal and the complex coefficient c1(t) a white Gaussian noise.

The carrier phase tracking process of a signal transmitted through a mobile channel is more complicated than the carrier tracking process of a signal transmitted through an AWGN channel. In fact, in addition of the AWGN channel sources of errors, the mobile channel introduces a fast non-linear variation of the signal carrier phase. This fast non-linear carrier phase variation limits the PLL coherent integration time T_i since the PLL usually generates a linear variation of the phase between two consecutives estimated phase values. And each estimated phase value linear evolution is provided each coherent integration time T_i seconds.

The channel estimation method has been introduced as an alternative method to the traditional PLL. The channel estimation method consists in estimating the channel impulse response of a period of time and in equalizing the signal with this estimation. One particularity of the application of a channel estimation method on a GNSS signal is that the channel estimation is made on the pilot channel. The two channel estimation methods proposed in this dissertation are the following. The first method is the ideal channel estimation method. This method is assumed to obtain a perfect instantaneous estimate of the channel impulse response. Therefore, this method is a lower bound of the performance obtained by any channel estimation method. The second method consists in averaging the pilot channel samples of a period in order to obtain a constant estimate of the channel impulse response over this period. This method is thus limited by the channel variation rate. Moreover, this method is very simple compared to other more complex methods such as the least-squares (LS) method and minimum mean square error (MMSE) method which provide a better performance; therefore, this method can be considered as an upper bound.

The demodulation process of a signal transmitted through a mobile channel is penalized by the instantaneous decrease of the received useful signal power. This decrease is really a drawback when affecting in several consecutive bits since it complicates the decoding process of the implemented channel code. Therefore, in order to provide a satisfactory demodulation performance, a signal transmitted through a mobile channel requires the implementation of an interleaver.

The simulator modeling the transmission of a signal through a mobile channel has been presented in this chapter, and the generation of all its components has been described.

Chapter 4. Bit Error Rate optimization implementing a navigation message binary prediction

This chapter analyzes different methods of navigation message binary prediction in order to improve the GNSS signals BER at the receiver. More specifically, this chapter tries to improve the GPS L1 C/A signal BER. The reason is that when this study was conducted, the GPS L1 C/A signal was the only available GNSS signal and the only signal with enough broadcasted samples to obtain a representative history. Therefore, the remaining GNSS signals, such as GPS L1C, GALILEO E1 OS, etc, are not analyzed since some of them have not been even yet broadcasted.

The GPS L1 C/A navigation message binary prediction is very interesting, because the successful achievement of this prediction would allow the final user to obtain the satellite ephemeris data in worse reception conditions than the receiver normally could. And thus, this prediction allows the final user to locate himself within environments which were not accessible before, such as mobile or indoor environments.

This chapter presents first the problem targeted by the navigation message binary prediction. After, the chapter details the interest of the binary prediction, gives the definition of the binary prediction performance, presents some cases of study and makes a first analysis of the complexity of the binary prediction. Third, this chapter inspects the binary performance of different predictions methods which can be separated into two main groups, the temporal and the spatial methods. Finally, the extracted conclusions are commented.

4.1. Presentation of the problem

The correct demodulation of all the received satellite navigation messages is fundamental in order to obtain the final user position. This means that one of the actual GPS L1 C/A signal limitations is the high C/N_0 required to correctly demodulate the navigation message transmitted by each satellite. In fact, the required levels of C/N_0 are normally reached in open environments; however, for mobile urban or indoor environments the available C/N_0 decreases considerably in comparison with the levels of an open environment.

One solution to this problem is to try to predict some of the navigation message content, and more specifically the satellite ephemeris data. Therefore, since this predicted information no longer has to be demodulated by the GPS terminal and thus cannot contain any error, the signal BER decreases. Nevertheless, note that when this research work refers to a navigation message prediction, it is referring to a prediction of the bits forming the message. This means that the final objective of the prediction is to search for the value of the bits forming the ephemeris data and not the ephemeris physical magnitude that they represent.

We have chosen to try and to predict the ephemeris data within the navigation data message for 2 reasons. The first one is the vital role that the ephemeris data plays into the final user position algorithm. The second one is the availability of the ephemeris data into the stream of bits forming the GPS L1 C/A message. In fact, each ephemeris set is carried by the GPS L1 C/A navigation message subframes 2 and 3, and it is repeated for several consecutives frames. Therefore, only one set of ephemeris data represents 40% of the message for about two hours.

Moreover, the ephemeris information has been stored for several years and thus allows the construction of a representative history. And from this history the prediction methods can be developed and tested.

4.2. Navigation message binary prediction

In this section, first the interest of the navigation message binary prediction is presented. Second, the different groups of methods which provide a binary prediction of a keplerian parameter are presented. Third, a description of the performance of the navigation message binary prediction is given, and the required level of binary prediction performance is presented. Finally, a justification of why these performance levels are hardly reachable is presented.

4.2.1. Interest of the binary prediction

The main advantage of the navigation message binary prediction is to decrease the level of C/N_0 necessary to achieve a determined BER value at the receiver. Therefore, since some environments limit the level of C/N_0 received by the user, the navigation message binary prediction will allow the correct positioning service in these environments whereas without the binary prediction it would be not possible. The reason is that the GPS L1 C/A receiver will obtain a BER value low enough to allow the ephemeris information demodulation if the binary prediction is implemented.

Moreover, a channel code can improve its performance if part of the received word if known. In fact, the knowledge of some of the bits forming the code word increases the code minimum distance, allows a greater correction and thus improves the channel code decoding performance. See [PROAKISe, 2001] for a channel code minimum distance definition. Besides, the channel code correction capacity is also improved by a binary prediction which provides the probabilities for the transmitted bits to have a certain value. Or in other words, the binary prediction does not have to necessarily provide the bits value, the binary prediction can provide the probability of having a certain bit value rather than the 0 or 1 bit values.

Finally, the binary message prediction also allows performing the data wipe-off technique which improves the tracking performance.

4.2.2. Groups of methods of the binary prediction

There are 2 main groups of methods which provide a binary prediction of a Keplerian parameter, the physical prediction methods and the binary prediction methods. First, the binary prediction methods aim at predicting the individual bits of a Keplerian parameter. Second, the physical prediction methods aim at predicting the Keplerian parameters natural value before converting the predicted natural value into a binary value in order to obtain the binary prediction.

In this dissertation, only the physical prediction methods are analyzed although some commentaries about the first group of methods, the binary prediction methods, can be made. The reason for this choice is the bigger complexity of the binary prediction methods with respect to the complexity of the physical prediction methods and the uncertainty of obtaining better results applying the binary prediction methods.

4.2.3. Binary prediction performance definition

The performance of the binary prediction of the n^{th} bit of a Keplerian parameter is defined as the probability of wrongly predicting the n^{th} bit value, that is the probability of assuming that the n^{th} bit of the given Keplerian field is equal to 1 when its value is really 0, or the probability of assuming that the n^{th} bit of the given Keplerian field is equal to 0 when its value is really 1. Therefore, the set of bit probabilities belonging to a Keplerian parameter is defined as the parameter prediction performance.

Moreover, the performance of the prediction of a bit of a given Keplerian parameter can be related to the performance of the prediction of another bit of the same Keplerian parameter when the Keplerian parameter prediction is made by using a physical prediction method. In fact, the main drawback of using a physical prediction method is that when the value of a bit is not correctly predicted, all the bits having less weight than the erroneous bit are completely random. The reason is that the method has not been able to provide a physical prediction accurate enough to correctly determine the value of the bit. Therefore, the physical prediction is not accurate enough to determine the values of the bits having less weight since they represent smaller quantities of the natural value of the predicted Keplerian parameter.

4.2.4. Required level of binary prediction performance

The performance of the binary prediction of the different bits belonging to the same Keplerian parameter is different for each bit and depends on the bit position inside the Keplerian parameter binary word: the performance level of the binary prediction of a bit depends on the bit weight. This statement is true for the physical predictions methods but it has still to be proven for the binary ones. The reason is given next.

The decimal value represented by each bit of the binary format of the Keplerian parameter depends on the position of the bit inside the parameter, where the MSB represents the larger decimal value and the LSB represents the smaller decimal value. Therefore, the number of bits which can be correctly predicted depends on the precision of the physical prediction. And this means that the precision necessary to correctly predict the values of bits having more weight is lower than the precision necessary to correctly predict the values of bits having less weight. A more detailed explanation is given in section 4.3.

Finally, the level of performance required from the binary prediction is set from the specifications imposed in this dissertation. These specifications can be summarized into a case of study which is explained in the next subsections. Moreover, the assumptions of the case of study and the required level of performance calculated from the case of study are presented.

4.2.4.1. Case of study definition

The case of study is defined from the specifications imposed in this dissertation. These specifications are that the technique should provide correct predictions during all the life span of a GNSS receiver, about 5 years, and that during this time only one not detected error of the final user position is allowed to the binary prediction. This means that the binary prediction cannot provide wrong predictions of the bits value, except once in 5 years. All the other user positions should be correctly provided.

The case of study can thus be defined as follows. The binary prediction is allowed one and only one user position error in 5 years of time, the time life of a GPS receiver. This statement has to be fulfilled for 99% of sets of 5 years.

Moreover, a user position error is defined as a user position which is at a distance equal or bigger than 1 meter from the real user position. Therefore, we chose a distance threshold of 1 meter because the ephemeris data set broadcasted by a satellite introduces a satellite position error which has a standard deviation of about 1 meter [WARREN and RAQUET, 2003]. Consequently, we consider that adding an error bigger than the standard deviation of broadcasted ephemeris error is not admissible.

4.2.4.2. Assumptions

The following assumptions are set in order to consider the worst case scenario:

- The bit prediction is only necessary in indoor or urban environments since the GPS L1 C/A demodulation performance is satisfying in outdoor environments. Therefore, this technique is used approximately 10% of the time (= 6 months).
- An average of 8 satellites pseudo-range measurements are available to calculate the user position.
- A user position error is due to at least one satellite position error.
- A satellite position error is due to at least one Keplerian parameter error. This assumption is proven in annex F.4.
- A Keplerian parameter error is due to the wrong prediction of at least one bit of the Keplerian parameter. This assumption is proven in section annex F.4.
- A RAIM function is implemented in the GPS receiver. This RAIM algorithm incorporates a fault detection function which aims at ensuring the integrity of the used pseudo-range measurements. If a faulty range measurement on one or more satellites is detected, an alarm is raised to warn the user. The algorithm fault detection function is able to detect any position failure equal or larger than 600m with a non detection probability of 10^{-3} for a receiver mono-frequency in a urban environment [ESCHER, 2003].
- The RAIM function is able to exclude the satellite among all the received satellites that has a faulty range when the other satellites have a correct pseudo-range measurement. The faulty pseudo-range measurement of a satellite is caused by the erroneous prediction of its position. If this satellite is removed from the erroneous estimation of the user position, the error of the user position is corrected [ESCHER, 2003].
- The number of bits to be predicted is determined by the number of erroneous predicted bits which cause a satellite position error equal or larger than 600 meters since the RAIM algorithm cannot guarantee the detection of smaller errors. The number of erroneous bits which cause a satellite position error equal or larger than 600 meters has been calculated in annex F.4.
- The methods used to obtain the binary prediction of the Keplerian parameter are the physical methods presented in section 4.3. Therefore, the properties of the binary prediction errors and its distribution depend on the errors of the physical prediction methods. This means that if a bit of a Keplerian parameter is wrong predicted, all the bits of the Keplerian parameter having less weight have a random value.
- Each Keplerian parameter has a bit called first position error bit. This bit is the bit having the biggest weight which cannot be predicted with a probability of error equal to 0%. This means that all the bits having a bigger weight are predicted with a probability of error equal to 0%. Moreover, from the previous assumption, it can be assumed that all the bits

having a smaller weight than the first position error bit are predicted with a probability of error bigger than 0%.

- The binary prediction of the Keplerian parameter X of a satellite I is independent from the next binary prediction of the same Keplerian parameter X of the same satellite I. This means that an error of the binary prediction of Keplerian parameter X does not give any information about whether an error has been made on the next binary prediction of the same Keplerian parameter X.

4.2.4.3. Theoretical calculations

The detailed theoretical calculations conducted to find the probability of error of the most significant bit for which an error can be tolerated are presented in annex F.1. However, the general mathematical model is given next in order to set out the case of study.

The model starts with the requirement of having not more than 1 user position error 99% of the time. This probability is the sum of two conditional probabilities: the probability of not having any user position error during the defined period plus the probability of having 1 user position error during the defined period. Both probabilities are measured after the RAIM application and their sum has to be bigger than 0.99, a percentage defined during the case of study presentation.

$$P(0\ user\ err\ after\ RAIM) + P(1\ user\ err\ after\ RAIM) \geq 99\% \qquad (4\text{-}1)$$

Where

- 0 user err after RAIM: 0 user position errors after the RAIM application during 5 years
- 1 user err after RAIM: 1 user position error after the RAIM application during 5 years

4.2.4.4. Results

In this section, the results obtained from section 4.2.4.3 are presented. The results present the method exposition time, the real time during which the prediction method has been used, and the required error probability of the bit of most weight of a Keplerian parameter which is allowed to be erroneously predicted. In fact, the results are calculated by constraining either the method exposition time or the error probability. If the method exposition time is imposed, the results provide the error probability that the physical prediction method has to provide. If the error probability provided by the physical prediction method is imposed, the results provide the method exposition time. Table 4-1 shows the results.

Method Exposition time	Method application time	Required error probability
≅ 15 years and 4 months	570 days	5e-5
5 years	182 days and 12h	9.5e-5
4 years and 6 months	166 days	1e-4
1 year and 3 months	45 days and 12h	2e-4
2 months and 18 days	7 days and 10h	5e-4
20 days and 20h	2 days and 2h	1e-3

Table 4-1: Physical prediction method requirements for the prediction of all Keplerian parameters bits when only 1 bit error per ephemeris set is allowed.

The required error probability is ideally the error probability of the Keplerian parameter LSB. Nevertheless, the probabilities of Table 4-1 are extremely low to be obtained by physical prediction methods for the Keplerian parameters LSB. The reason is that the decrease of the weight of the bit to predict increases the difficulty of the prediction. Therefore, this probability can also be interpreted as the probability of error of the bit of less weight of the Keplerian parameter which is to be predicted when the bits of more weight are assumed to be perfectly predicted. For example, if the method seeks to predict 10 bits, the first 9 bits have to be perfectly predicted and the 10^{th} bit has to have a bit error probability at most equal to the value specified by the method exposition time.

Moreover, in order to use the RAIM function, the erroneous prediction of the bits has to cause a satellite position error larger than 600 meters. And an error of this magnitude is due to the erroneous prediction of the bits with significant weight of the Keplerian parameters. See annex F.4 for an example.

Finally, from Table 4-1, it can be observed that the error probabilities required for a significant method exposition time, about 1 year, are really low. Even for 2 months of exposition time, the error probability is 4 orders of magnitude smaller than 1. Therefore, the prediction of the bits of the ephemeris data set is very limited by these probabilities.

4.2.5. Theoretical analysis of the relationship between the decimal and binary precision

In this section, a first analysis of the relationship between the decimal precision and the binary precision is presented. This analysis sets the level of accuracy of a decimal value of a physical prediction method which is required in order to predict a given number of bits when using the binary format of the decimal predicted value. This means that if a physical prediction method cannot guarantee the required level of accuracy, the method cannot guarantee the prediction of a given number of bits and thus the specifications imposed in this dissertation are not fulfilled. In this research work, in order to present this accuracy, two different percentages which depend on the position inside the field of the desired bit to be predicted are given. Note that the results are given as percentages, so in relative terms in order to provide a general analysis.

The first percentage is defined using the maximal acceptable decimal distance and the maximal range value. The maximal acceptable decimal distance is the larger distance between the predicted value and the real value which still allows the correct prediction of the desired bit. In other words, the maximal acceptable decimal distance does not guarantee the bit prediction, only guarantees that the prediction is possible: if the distance between the prediction and the real value is larger than the maximal decimal distance, the desired bit cannot be correctly predicted. The maximal range value is the maximal decimal value of the predicted field when all the bits are set to 1. The first percentage is then the ratio between the maximal acceptable decimal distance and the maximal range value.

The second percentage is defined using the minimal acceptable decimal distance and the maximal range value. The minimal acceptable decimal distance is the smaller distance between the predicted value and the real value which does not ensure the correct bit prediction. Indeed, a physical prediction method cannot ensure the correct prediction of a bit if the method provides a prediction separated from the real decimal value by a decimal distance larger than the minimal acceptable decimal distance. The second percentage is then the ratio between the minimal acceptable decimal distance and the maximal range value.

The mathematical development used to calculate the first and second percentage for a given number of bits can be found in annex F.2. The results calculated for a given number of bits which we want to predict are given next.

Table 4-2 summarizes the results of the first percentage or the maximal precision distance. Nevertheless, an approximation is shown in order to generalize the table for any number of bits quantifying the field to predict.

	Bit Position							
	1	**2**	**3**	**4**	**5**	**6**	**10**	**16**
Maximal Precision Distance (%)	≈50	≈25	≈12.5	≈6.25	≈3.125	≈1.5625	≈0.0977	≈0.0015

Table 4-2: Maximal precision distance as a function of the bit position

From Table 4-2, we can observe a fast decrease of the maximal precision distance necessary to have the possibility of predicting a bit. Remember that if the predicted value is farther away than the maximal precision distance, the physical prediction method is unable to predict the n^{th} bit of the parameter. Therefore, the implemented physical prediction method has already to have a very high accuracy in order to allow the prediction of a small number of bits, such as 5 bits

Table 4-3 summarizes the results of the second percentage or the minimal precision distance. This table shows the minimal precision distance as a function of the numbers of bits of the field to predict since the second percentage is the same for any given number of bits which we want to predict.

	Number of bits			
	5	**8**	**10**	**16**
Minimal Precision Distance (%)	1.5625	0.1953	0.0488	7.6294e-004

Table 4-3: Minimal precision distance as a function of number of bits of the field to predict

From Table 4-3, we can observe that even for a small number of bits, the minimal precision distance is very small in comparison with the maximal range value. This means that the implemented physical prediction method has to have an extremely high accuracy in order to ensure the prediction of any bit. For example, for a 10 bit number, the accuracy required in order to ensure the prediction of even the first bit of the number is 0.05%

To sum up, we need a physical prediction method which provides decimal predictions with a very high precision with respect to the real value. In fact, the method has to provide a really accurate prediction in order to have the possibility of predicting certain number of bits, and has to provide an extremely accurate prediction in order to guarantee the prediction of even the first bit of the number/field.

Therefore, we can conclude that a satisfying performance of the binary prediction using physical prediction methods will be very difficult to obtain. Moreover, in addition to this problem concerning the relation between a binary number and a decimal number, another problem associated to the available data can increase even more the complexity of the binary prediction.

4.3. Prediction methods

In this section, the methods applied during this Ph.D. in order to try to predict the future GPS L1 C/A ephemeris data sets are presented. The performance obtained using these methods is quite reduced as it was expected from the analysis of subsection 4.2.3, 4.2.4 and 4.2.5.

The different methods implemented can be divided into two main groups. The first group includes the temporal methods, or, in other words, the methods which analyze the evolution of the Keplerian parameters along the time in order to predict their future samples. The second group includes the spatial methods, or in other words, the methods which use spatial information related to the Keplerian parameters in order to predict the Keplerian parameters.

In this section, first the temporal methods and second the spatial ones are presented.

4.3.1. Temporal methods

This section presents the temporal methods applied during this Ph.D. in order to predict the future GPS L1 C/A ephemeris sets. A temporal method tries to predict the future evolution of a signal after the analysis of its history. And a signal history is the evolution of the signal along the past time. Therefore, in this Ph.D., the satellite ephemeris data of the four last years, 2004 to 2007, from the page web http://igscb.jpl.nasa.gov is used in order to generate a GPS L1 C/A ephemeris history.

In order to reduce the complexity and difficulty of the ephemeris data prediction, a simplification is introduced. This simplification has two main advantages. First, the simplification consists in trying to predict part of the GPS L1 C/A ephemeris sets. Therefore, the simplification reduces the quantity of bits to predict using the temporal methods. Second, the data resulting from the simplification process follows a pattern easier to analyze than the original data. Therefore, the resulting data is easier to predict by the temporal methods.

Therefore, in this section, first the proposed prediction simplification is presented. Second, the data resulting from the simplification is analyzed and the temporal methods more suited to predict the data are chosen. Finally, the temporal methods and their results are presented.

4.3.1.1. Simplification of the binary prediction of the ephemeris data

The binary prediction of the ephemeris data is a very complicated process as shown in sections 4.2.3, 4.2.4 and 4.2.5. In fact, in addition to the high required accuracy of the physical prediction method, other factors such as the irregularity of the stored ephemeris and the errors of the broadcasted ephemeris data makes the binary prediction of the ephemeris data still more difficult to attain. Therefore, we present an intermediate step which simplifies the final ephemeris data binary prediction. This step is applied before the physical prediction method.

This simplification or intermediary step consists in making a first binary prediction of the satellite ephemeris data through either the satellite almanacs data or physical ephemeris predictions which use spatial mechanics models. Therefore, the results of applying this simplification are a first binary prediction of the ephemeris data, the perfect binary prediction of the MSB of the ephemeris data and a decrease of the difficulty of the binary prediction of the remaining bits.

Finally, in the next subsections, the main idea of the simplification process is presented, the simplification gain is defined, the almanac data and long term orbital prediction programs

simplifications are analyzed and a method which allows the use of the best simplification is presented.

4.3.1.1.1. Main idea

Each of the Keplerian parameters of the ephemeris data is represented by a significant number of bits which provide a high decimal resolution of the parameter. Therefore, due to this high decimal resolution, the GPS receiver can obtain very precise satellite position estimates. However, such a large number of bits oblige to obtain a very accurate decimal prediction if we want the physical prediction method to predict a significant amount of bits. Therefore, in order to decrease the level of required accuracy, an intermediate step or simplification is implemented.

The simplification consists in trying to predict the difference between the desired Keplerian parameter and a reference value defined later, rather than the Keplerian parameter itself. Therefore, we try to predict the residual data instead of the satellite ephemeris data set with a physical prediction method. From now on, we call the difference between the ephemeris data and the reference value, the residual data or residual signal. Two reference values have been considered to be subtracted to the ephemeris data; the satellite almanacs data and the satellite ephemeris data provided by long term orbital prediction programs.

The satellite almanac data, as well as the satellite ephemeris data, is a group of Keplerian parameters which describes the satellite orbit in Cartesians coordinates. The accuracy obtained using the almanac data is lower than the accuracy obtained using the ephemeris data. One of the reasons is that the almanac data only includes some of the ephemeris Keplerian parameters. Moreover, the orbit predicted with the almanac data is valid for a period of time equal to 6 days whereas the orbit predicted with the ephemeris data is valid for only 4 hours. Nevertheless, the GPS L1 C/A signal broadcasts one new ephemeris data set each two hours and one new almanac data set each day.

The ephemeris data obtained by a long term orbital prediction program should provide better results than the almanac data. Indeed, the software should predict almost exactly the future ephemeris data in decimal format since it uses spatial mechanics models. Moreover, the number of Keplerian parameters provided by the long term orbital prediction program is the same than the broadcasted GPS L1 C/A ephemeris.

4.3.1.1.2. Bit gain of the simplification

The bit gain is defined as the number of bits which are equal between the satellite ephemeris data and the reference value. Consequently, these common bits do not have to be predicted by the later application of a temporal method. The bit gain can be calculated with two different methods:

- **XOR**: Application of the XOR operator between the broadcasted ephemeris data and the reference value. This operation is made using both elements expressed in binary format. From now on, we call this difference the XOR difference.
- **DIFF**: Binary conversion of the absolute decimal value of the subtraction. The subtraction is the difference between the broadcasted ephemeris data and the reference value expressed in decimal format. From now on, we call this difference the DIFF difference.

Note that the result of these differences is a binary vector of the same length as the Keplerian parameter of the ephemeris data. This means that the first bit of the vector different from 0 is the first bit that cannot be predicted. Therefore, the bit gain level is determined by the last MSB of the vector equal to 0: all the bits equal to 0 can be predicted by the reference value with a probability of error equal to 0%.

The difference and the utility of the bit gain determined by each difference are explained next. The real binary error of prediction between the reference value and the broadcasted ephemeris data is the first difference, the XOR difference. However, the DIFF difference which presents the decimal difference expressed in binary format is lower or equal than the XOR difference. This fact is due to the particularities of the addition and subtraction in the binary domain which means that the addition of a small value X to another value Y can affect a lot of bits of the value Y which have more weight than the bits of the small value X. Table 4-4 shows two examples.

Ephemeris	*Reference Value*	*XOR difference*	*DIFF difference*
0111	0110	0001	0001
1000	0111	1111	0001

Table 4-4: XOR and DIFF difference for different ephemeris and related values

Therefore, it is more interesting to use the DIFF difference instead of XOR difference but the DIFF difference cannot be directly applied. A method allowing the use of the results of the DIFF difference instead of the results of the XOR difference is presented in annex G.1.

4.3.1.1.3. Simplification using the almanac data

In this section, the performance of the process of the simplification using the almanac data is presented. These results are only presented for each Keplerian parameter common to the almanacs and the ephemeris. The exact process used to apply the simplification using the almanac data can be found in annex G.2.

Eight different results are presented for each Keplerian parameter, four for each type of difference. For each difference and each Keplerian parameter, the results show the bit of most weight which is not perfectly estimated and the probability of error of this bit. The probability of error of a bit represents the percentage of times that the bit of the Keplerian parameter estimated using the almanac data value is different from the real broadcasted bit of the Keplerian parameter. Moreover, the results present the bit of most weight which has probability of error equal or bigger than 1% and the exact value of this probability.

The probability of error of the bit on the n^{th} position inside a Keplerian parameter is calculated by dividing the number of difference samples which have the n^{th} bit as the bit of most weight which is different from 0 by the total number of difference samples. Remember that the bit of most weight of the binary vector representing a difference sample which is equal to 0 determines the bit of most weight which has been erroneously predicted. Moreover, remember that when the n^{th} bit of a broadcasted Keplerian parameter has been wrong predicted with a physical prediction method, or, in this case, an almanac data, the values of the bits of the broadcasted Keplerian parameter of less weight are random.

The results of the simplification process using the almanac value for satellite PRN 1 are shown below.

	$\sqrt{A}$ (32 bits)		ω (32 bits)		e (32 bits)		i_0 (32 bits)	
DIFF – Bit Position	18	18	12	13	15	16	17	17
DIFF – Probability of Error (%)	100	100	0.25	18.90	0.09	6.84	7.36	7.36
XOR – Bit Position	16	16	5	9	10	12	10	10
XOR – Probability of Error (%)	13.31	13.31	0.91	2.16	0.17	1.15	1.95	1.95

Table 4-5: Results of the simplification using almanac data for satellite PRN 1 (part 1 of 2)

	Ω_0 (32 bits)		M_0 (32 bits)		$\dot{\Omega}$ (24 bits)	
DIFF - Bit Position	1	6	1	1	14	14
DIFF – Probability of Error (%)	0.35	13.49	24.24	24.24	9.47	9.47
XOR - Bit Position	1	3	1	1	12	13
XOR – Probability of Error (%)	0.58	1.16	50	50	0.53	20.54

Table 4-6: Results of the simplification using almanac data for satellite PRN 1 (part 2 of 2)

The probabilities of error of the bits are calculated by using 677 almanac data sets as estimations of 8124 ephemeris data sets.

From Table 4-5 and Table 4-6, a significant bit gain for 4 parameters, $\sqrt{A}$, e, $\dot{\Omega}$ and i_0 can be observed since the almanac is able to predict almost half of the parameter bits. The bit gain definition was given in section 4.3.1.1.2. Moreover, it can also be observed a bad bit gain for ω, as only third of the bits are well estimated, and no bit gain for the remaining parameters. Besides, the XOR performance is worse than the DIFF performance as it was expected.

Finally, it can be concluded that there is a significant bit gain for some Keplerian parameters while others Keplerian parameters have a very poor improvement. Moreover, all the probabilities of error of the bits of most weight of the Keplerian parameters which cannot be perfectly predicted are too high with respect to any case of study analyzed in section 4.2.4. Therefore, in order to fulfill the requirements of any case of study, the last bit to be predicted has to be the last bit which can always be correctly estimated by the almanacs.

4.3.1.1.4. Simplification using long term prediction ephemeris data

In this section, the performance of the process of the simplification using the long term prediction ephemeris data is presented.

The TAS long term orbital prediction program consists in predicting the orbit of a satellite in Cartesian coordinates (XYZ) and in transforming 4 consecutive hours of these coordinates into a set of Keplerian parameters which define this part of the satellite orbit. This means that the TAS long term orbital prediction program has two main sources of error. The first source of error is the prediction of the satellite position in Cartesians coordinates (XYZ). The second source of error is the transformation of a 4h period of Cartesians coordinates into the Keplerian parameters defining an ephemeris data set. Therefore, it is more convenient to analyze the two sources of errors separately in order to simplify the analysis and the interpretation of the results. However, in this dissertation, we only analyze the second source

of errors since the number of errors introduced by this source alone is already too high to allow a satisfactory performance. Therefore, the results presented only belong to the second source of errors.

In order to analyze the second source of errors, first, we calculated the orbit of a satellite defined by a broadcasted ephemeris data. Second, from the Cartesians coordinates of the orbit, we applied the part of the TAS program responsible for transforming the coordinates in Keplerian parameters. Third and last, we compared the broadcasted ephemeris data to the ephemeris data calculated by the TAS program. The exact process used to apply the simplification using long term prediction ephemeris data taking into account only the second source of errors can be found in annex G.3.

The results of the second source of errors are presented next. Eight different results are presented for each Keplerian parameter, except the t_{oe}, four for each type of difference. For each difference and each Keplerian parameter, the results show the bit of most weight which is not perfectly estimated and the probability of error of this bit. This probability of error has been calculated as explained at the end of the previous section 4.3.1.1.3, and it represents the percentage of times that the bit of the Keplerian parameter estimated using the long term prediction ephemeris data value is different from the real broadcasted bit of the Keplerian parameter. Moreover, the results present the bit of most weight which has probability of error equal or bigger than 1% and the exact value of this probability.

The results of the second source of errors of the simplification process using long term prediction ephemeris data for satellite PRN 1 are shown below.

	$\sqrt{A}$ (32 bits)		ω (32 bits)		e (32 bits)		i (32 bits)	
DIFF – Bit Position	21	21	14	15	18	18	21	21
DIFF – Probability of Error (%)	5.63	5.63	0.61	8.46	2.91	2.91	5.12	5.12
XOR – Bit Position	17	18	5	11	11	15	10	18
XOR – Probability of Error (%)	0.88	1.86	0.09	1.18	0.11	1.13	0.07	1.43

Table 4-7: Results of the second source of errors of the simplification process using long term prediction ephemeris data for satellite PRN 1 (part 1 of 4)

	Ω_0 (32 bits)		M_0 (32 bits)		$\dot{\Omega}$ (24 bits)		*IDOT* (16 bits)	
DIFF - Bit Position	1	17	1	1	15	15	5	6
DIFF - Probability of Error (%)	0.01	49.14	8.14	8.14	10.03	10.03	0.02	7.15
XOR - Bit Position	1	13	1	1	12	13	1	1
XOR - Probability of Error (%)	0.04	1.84	16.71	16.71	0.57	6.47	4.22	4.22

Table 4-8: Results of the second source of errors of the simplification process using long term prediction ephemeris data for satellite PRN 1 (part 2 of 4)

	CUC (16 *bits*)		*CUS* (16 *bits*)		*CIC* (16 *bits*)		*CIS* (16 *bits*)	
DIFF – Bit Position	7	8	7	8	9	9	9	9
DIFF – Probability of Error (%)	0.74	8.96	0.82	9.05	2.13	2.13	1.49	1.49
XOR – Bit Position	1	1	1	4	1	1	1	1
XOR – Probability of Error (%)	1.46	1.46	0.48	1.43	20.62	20.62	23.5	23.5

Table 4-9: Results of the second source of errors of the simplification process using long term prediction ephemeris data for satellite PRN 1 (part 3 of 4)

	CRC (16 *bits*)		*CRS* (16 *bits*)		*Δn* (16 *bits*)	
DIFF - Bit Position	7	7	7	7	7	7
DIFF – Probability of Error (%)	1.39	1.39	1.36	1.36	1.38	1.38
XOR - Bit Position	3	3	1	1	4	5
XOR – Probability of Error (%)	1.19	1.19	1.41	1.41	0.12	4.76

Table 4-10: Results of the second source of errors of the simplification process using long term prediction ephemeris data for satellite PRN 1 (part 4 of 4)

The probabilities of error of the bits are calculated by using 4062 transformed ephemeris data sets as estimations of 8124 ephemeris data sets.

From the previous tables, it can be observed that the probabilities of error of the bit of most weight of each Keplerian parameter which cannot be perfectly predicted are still too high with respect to any case of study presented in section 4.2.4. Therefore, in order to fulfill the requirements of any case of study, the last bit to be predicted has to be the last bit which can always be correctly estimated by the transformation XYZ-ephemeris part of the long term orbital prediction program of TAS.

Finally, the results of the second source of errors of the TAS program are not much better than the results of the simplification using almanac data. This small difference of performance is surprising although a closer examination of the program justifies it. The justification is provided in annex G.4.

4.3.1.2. Selection of temporal methods

In this section, we choose the different temporal methods applied during this work in order to predict the difference signal obtained by the simplification process of the broadcasted ephemeris data. In fact, we have decided to analyze only the difference signal obtained by the simplification process using almanac data since all GPS L1 C/A users can freely access them, whereas the users have to pay the TAS long term orbital prediction program. Moreover, the performance of the simplification process using the TAS long term orbital prediction program was not sufficiently better than the performance of the simplification process using almanac data in order to change this decision.

Therefore, in order to choose different temporal prediction methods that best suit the difference signal, examples of difference signals of different Keplerian parameters are shown below.

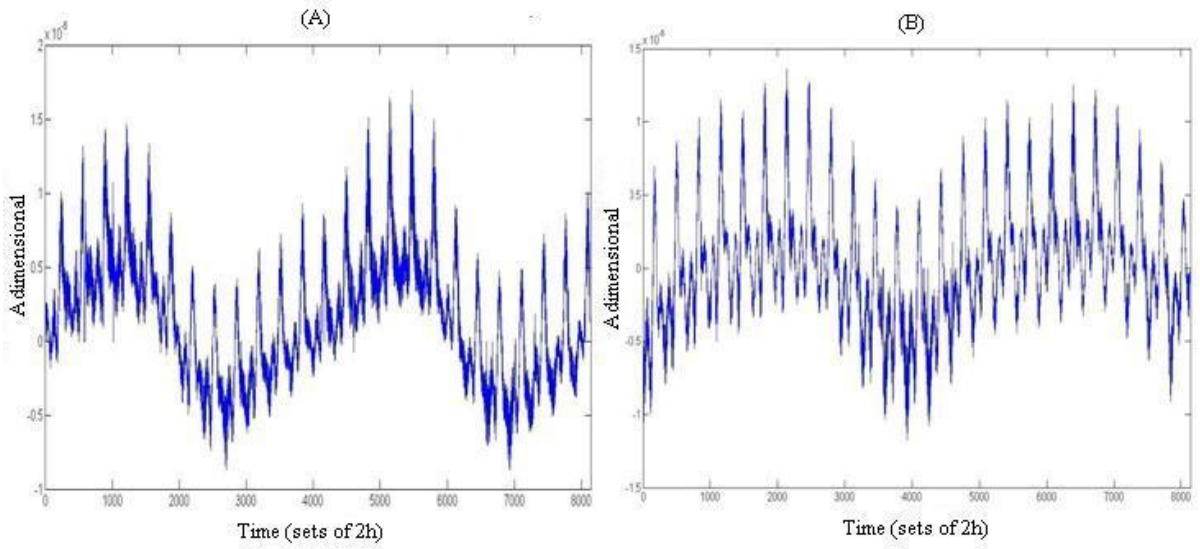

Figure 4-1: Eccentricity residual signal between the broadcasted ephemeris and the almanacs. (A) Satellite 1. (B) Satellite 2.

From Figure 4-1, it can be observed that the residual signal for a single Keplerian parameter depends on the broadcasting satellite. Therefore, even if a general prediction method can be implemented, its parameters should be adapted for each received satellite. This means that we need to create a history for each individual satellite and we need to analyze the history of each satellite. Moreover, this means that a GPS L1 C/A receiver has to store the parameters method for each individual satellite.

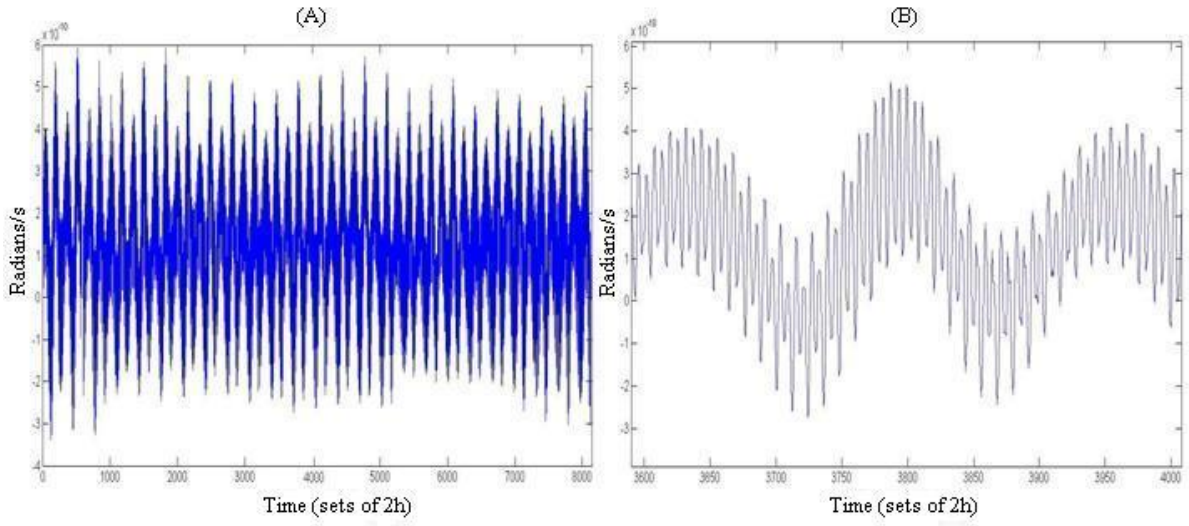

Figure 4-2: $\dot{\Omega}$ residual signal between the broadcasted ephemeris and the almanacs. (A) Satellite 1. (B) Central zoom of figure (A).

From Figure 4-1and Figure 4-2, it can be observed that each different Keplerian parameter has a different residual signal. Therefore, in addition to analyze the history for each different satellite, we have to analyze the history of each different Keplerian parameter.

Moreover, from the observation of Figure 4-1and Figure 4-2, we can determine different prediction/estimation methods. In fact, the residual signals seem to be an addition of sinusoids as can be appreciated on Figure 4-2B. Therefore, we implement specific signal models or

methods which allow the estimation of the sinusoids amplitude, frequency and phase. Therefore, two different types of methods can be implemented: the PRONY model and spectral estimation methods. Both techniques have their drawbacks and their advantages but their principal difference is the input signal length for which they are optimized. Moreover, the PRONY method is parametric and the spectral estimation is not.

- Spectral Estimation: The longer the signal is, the better the resolution is. This method is not parametric.
- PRONY: Optimized for short signals where it has approximately four times [MARPLEd, 1987] the resolution of the Fourier Transform. This method is parametric.

Additionally, since the signal seems to be periodic, we also implement another more complex method, the neural network. This method is more advanced than the previous methods, has a really high accuracy in comparison with the other methods and should thus provide the better performance.

4.3.1.3. Spectral estimation

In this section, the spectral estimation method selected to predict the difference signal of the simplification process using the almanac data is presented.

This section begins by presenting first the main idea of a generic spectral estimation method and second by describing the specific method chosen to conduct the spectral estimation. Afterwards, this section presents the estimation and prediction results.

4.3.1.3.1. Main idea

The main idea of the spectral estimation is to determine the main characteristics of the sinusoids, amplitude, frequency and phase, from the representation of the signal in the frequency domain. The reason is that these characteristics are much easier to identify in the frequency domain than in the temporal domain. In fact, in the frequency domain, the most significant peaks are identified and determined as the spectral components of the difference signal. Therefore, since each peak represents a sinusoid, their total number and their characteristics can be obtained. Finally, once all the sinusoids have been determined, their addition should regenerate the residual signal which is the final objective of the method.

4.3.1.3.2. Blackman-Tuckey method

There are several spectral estimation methods suitable to be implemented. In this case, the method chosen is the Blackman-Tuckey [MARPLEa, 1987] method implemented with the Bartlett window.

This method is a variation of the periodogram [MARPLEa, 1987] and thus searches the signal spectral components from the FFT of the autocorrelation of the signal. Moreover, the main difference of this method with respect to the periodogram consists in the elimination of the extremities of the estimated signal autocorrelation. The reason is that these points have an undesired high variance due to the 'biased' autocorrelation estimator applied to obtain the signal autocorrelation. Therefore, a Bartlett window of half the correlation signal length is applied at the autocorrelation signal center.

The mathematical method and general process can be found in annex G.5.

4.3.1.3.3. Blackman-Tuckey results

The Blackman-Tuckey results are divided into two types. We first analyze how well this method is able to reproduce the analyzed signal history. Then, we analyze how well this method is able to predict the future signal after analyzing the signal history. In both cases, the results are expressed as the bit error probabilities defined in section 4.3.1.1.3. This probability is the probability that the n^{th} bit is the bit of most weight which is different between the broadcasted ephemeris data set and the addition of the almanac data set with the difference signal estimated/predict by the Blackman-Tuckey method. See section 4.3.1.1 for explanation of the simplification process and the difference signal.

This section first shows the estimation results and then presents the prediction results. In the estimation results, we predict the same samples of the difference signal as the samples used to estimate the sinusoids forming the difference signal. In the prediction results, the predicted samples of the difference signal are not the same samples used to estimate the sinusoids forming the difference signal.

4.3.1.3.3.1. Blackman-Tuckey estimation results

The estimation results are only presented for the DIFF difference type of bit gain because they show more clearly than the XOR difference the physical gain introduced by the estimated difference signal. See section 4.3.1.1.2 for a description of the bit gain. Moreover, the results present the bit error probabilities of several consecutive bits in order to better appreciate the gain introduced by the estimated difference signal instead of only showing the bit of most weight which has not been perfectly estimated.

Therefore, the next two tables show the performance level achieved by the simplification process using almanac data plus the estimation of the difference signal using the Blackman-Tuckey method. The results are shown for the eccentricity and ω Keplerian parameters of satellite 1. The results are presented as a function of the number of assumed sinusoids.

Eccentricity (32 bits)		**Bit error probability**								
		15	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*
Number of Sinusoids	*0*	0.09	6.84	30.95	31.93	14.92	7.92	3.66	1.92	0.91
	20	0	0	0.16	2.55	14.67	28.8	24.77	14.15	7.82
	50	0	0	0.06	0.74	3.76	21.14	30.41	20.35	11.51
	100	0	0	0	0.67	0.75	10.39	28.03	27.28	16.31
	200	0	0	0	0.2	0.84	5.14	25.38	29.1	19.13
	500	0	0	0	0	0.37	0.97	11.22	28.76	26.79
	1000	0	0	0	0	0.17	0.39	3.82	23.04	30.37
	10000	0	0	0	0	0	0	0	0	0

Table 4-11: Satellite 1 eccentricity DIFF difference bit error probabilities when applying the simplification process using almanac data plus the estimation of the difference signal using the Blackman-Tuckey method

Table 4-11 is calculated over 8124 broadcasted ephemeris data sets. Moreover, the bit of most weight having a probability of error different from 0 when the difference signal is regenerated

with 10000 sinusoids is the 26th bit. Note that the first row is equal to the results of the simplification process using almanac data.

ω (32 bits)		Probability of having wrongly estimated the bit on the i^th position								
		12	*13*	*14*	*15*	*16*	*17*	*18*	*19*	*20*
Number of Sinusoids	*0*	0.03	18.9	37.2	22.0	10.7	5.61	2.63	1.37	0.8
	20	0	0.01	0.41	5.32	21.9	28.4	20.3	11.9	6.01
	50	0	0	0.39	0.69	8.11	26.2	28.5	17.9	9.24
	100	0	0	0.26	0.32	1.95	14.6	30.0	24.8	14.04
	200	0	0	0	0.5	0.97	8.29	27.4	27.9	16.93
	500	0	0	0	0.1	0.39	1.76	15.6	31.1	23.78
	1000	0	0	0	0	0.21	0.55	7.79	27.0	28.39
	10000	0	0	0	0	0	0	0	0	0

Table 4-12: Satellite 1 ω DIFF difference bit error probabilities when applying the simplification process using almanac data plus the estimation of the difference signal using the Blackman-Tuckey method

Table 4-12 is calculated over 8124 broadcasted ephemeris data sets. Moreover, the bit of most weight having a probability of error different from 0 when the difference signal is regenerated with 10000 sinusoids is the 23rd bit. Note that the first row is equal to the results of the simplification process using almanac data.

From Table 4-11 and Table 4-12, it can be observed that in order to obtain a satisfactory bit gain, the number of sinusoids regenerating the difference signal has to be very large. For example, the bit gain is limited to 2 bits for Keplerian parameter ω when only 50 sinusoids are estimated. Besides, since 1000 sinusoids only provide a 4 bit gain, it can be concluded that at least 10000 spectral components should be generated in order to obtain a satisfactory binary estimation.

Nevertheless, the number of samples representing the difference signal power density function is about 16000. This means that when we create 10000 sinusoids in order to generate the difference signal, we are almost reproducing the entire difference signal spectrum. Therefore, it is quite possible that in addition to recreate the difference signal main spectral components, we are also reproducing the signal noise. And this noise consists in not periodic samples and in ephemeris errors. Therefore, this recreation of the difference signal spectrum with a very large number of sinusoids, 10000 for example, is useful when we estimate the difference signal; however the large number of sinus cannot help the prediction of the signal difference.

4.3.1.3.3.2. Blackman-Tuckey prediction results

The prediction results are only presented for the DIFF difference due to the same reason given for the estimation results. Nevertheless, in this case, the format of the results is not the bit error probabilities but rather the position of bit of most weight which is not correctly predicted. In fact, the prediction results are presented individually for each predicted ephemeris data set. In other words, for each broadcasted ephemeris data set, only the bit of most weight of the DIFF difference value which is different from 0 is shown. Remember that

the DIFF difference value is calculated as a subtraction between the broadcasted ephemeris data set and the broadcasted almanac data set plus the predicted difference signal using the Blackman-Tuckey method. Moreover, in order to better observe the gain introduced by the Blackman-Tuckey method, the individual performance of each ephemeris data set is shown when only the simplification process using almanac data is applied.

Therefore, in order to present the prediction results, we have divided the past broadcasted ephemeris data into two groups. The first group is formed by the first 8124 broadcasted ephemeris data sets and is used to estimate the desired number of sinusoids. The second group is formed by the last 2000 broadcasted ephemeris data and is used as the ephemeris data sets to be predicted. Therefore, once the sinusoids have been estimated, we generate a sum of sinusoids signal to be added to the almanac data sets in order to predict the second group of ephemeris data sets.

This section only presents the prediction results for a reduced number of ephemeris data sets. And the first ephemeris of this group is the ephemeris broadcasted immediately after the last ephemeris data set of the history, made of the previous 8124 broadcasted ephemeris data sets. The reason is that the prediction performance depends on the proximity between the predicted ephemeris and the ephemeris data history. Therefore, in order to show the best prediction performance, we show the results of the prediction of the ephemeris data sets which are the closest to the ephemeris data history.

More specifically, we present the prediction results for the eccentricity Keplerian parameter when only the simplification process using almanac data is applied, and when the simplification process using almanac data plus the difference signal generated with 10000 sinusoids is applied.

Eccentricity (32 bits)														
Sample Position*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*
Almanac simplification	17	17	18	17	19	18	18	18	19	18	21	20	18	18
Spectral Prediction	22	21	22	22	22	22	21	21	20	20	21	21	20	20
Sample Position*	*15*	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*	*24*	*25*	*26*	*27*	*28*
Almanac simplification	18	18	19	21	19	19	20	19	21	20	20	19	19	19
Spectral Prediction	20	20	20	21	20	20	19	22	20	20	21	20	19	20
Sample Position*	*29*	*30*	*31*	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*	*40*		
Almanac simplification	20	23	20	20	20	22	21	21	21	19	19	20		
Spectral Prediction	20	25	23	23	20	19	19	23	19	19	20	20		

Table 4-13: Satellite 1 eccentricity Keplerian parameter prediction results – Bit of most weight different from 0 of the DIFF difference value resulting from simplification process using almanac plus the difference signal generated with 10000 sinusoids

***Sample Position:** The sample position field marks the position of the predicted sample after the last sample of the difference signal used to estimate the difference signal

From Table 4-13, it can be observed that the number of bits predicted by the non-parametrical spectral technique is much smaller than the number of bits estimated by the non-parametrical spectral technique. For example, the bit of most weight different from 0 of the DIFF

difference value of the estimation results is the 26th bit and the bit of most weight different from 0 of the DIFF difference value of the prediction results is the 19th bit.

Moreover, from Table 4-13, it can also be observed that there are some samples which are better predicted by applying only the simplification process using almanac data instead of the simplification process using almanac data plus the predicted difference signal. Therefore, this fact implies that the 10000 sinusoids not only represent the difference residual signal but also the signal noise which cannot be predicted as explained for the estimation results (section 4.3.1.3.3.1).

Therefore, we can conclude that the spectral estimation technique does not provide a satisfactory bit gain compared to the desired bit gain, taking into account the costs of the method in terms of memory space and power consumption.

4.3.1.4. PRONY model

In this section, the PRONY model is used to predict the difference signal of the simplification process using the almanac data and the results of applying the method are presented. More specifically, in this section, first a definition of the method is provided and second the results are presented.

The exact mathematical expressions of the PRONY model as well as the method used to determine the number of exponentials composing of the signal is presented in annex G.6.

4.3.1.4.1. PRONY model definition

The PRONY method [MARPLEd, 1987] is a parametric estimation method applied in order to model sampled data as a linear combination of exponentials. This technique has a close relationship with the AR and ARMA parameter estimation methods [MARPLEb, 1987] but whereas these last techniques seek to fit in a general model with the second-order data statistics, the PRONY method searches to fit a deterministic exponential model to the data.

This technique has three basic steps [MARPLEd, 1987]:

1. Determine the linear prediction parameters that fit the available data.
2. Identify the roots of a polynomial formed from the linear prediction coefficients that determine the attenuation coefficient and the frequencies of the exponential terms.
3. Estimate the amplitude and sinusoidal phase of the exponentials.

Nevertheless, there are different methods for identifying the linear prediction coefficients, and we have decided to implement the modified covariance method because it is known as the most efficient [MARPLEd, 1987].

One important parameter of the PRONY model which is taken for granted during its application is the number of exponentials p composing of the signal or, in other words, the PRONY model order. However, this parameter has to be previously estimated. In this dissertation, the method used to estimate the number of sinusoids is the SVD method (Singular Value Decomposition) and is detailed in annex G.6.2. The performance of this method depends on the number of inspected samples of the signal q or, in other words, on the length of the input vector.

4.3.1.4.2. PRONY model results

The results of the PRONY model are presented as were presented the Blackman-Tuckey results. First, the estimation results are given and second the prediction results are shown. Moreover, the two types of results are presented with the same format as the Blackman-Tuckey prediction results. That means that the results are presented as an individual comparisons between the broadcasted ephemeris data sets and the almanac data sets plus either the estimation or the prediction of the difference signal using the PRONY model. Therefore, only the bit of most weight being different from 0 of the DIFF difference value is displayed. Moreover, for the estimation results, several figures plotting the real difference signal and the estimated difference signal using the PRONY model are shown.

Finally, two remarks have to be made before presenting the PRONY model results. The first remark concerns the number of samples used to characterize the signal with the PRONY model, N. In fact, as commented in section 4.3.1.2, the PRONY model was conceived to be applied on sequences with a small number of available samples. And the reason is that, for a small number of samples, the resolution achieved by the PRONY model is quite higher than the resolution obtained with non-parametrical spectral estimation methods. Therefore, since the number of available samples in the history is really large, we have to choose the amount of samples employed for the PRONY model.

We chose a small amount of samples because the application of sequences with large number of samples leads to MATLAB calculation numerical errors. Therefore, we used 1 or 2 weeks of samples for estimating the signal and the signal prediction was made over a period of 1 week.

The second remark is about the number q used to determine the PRONY model order, see annex G.6.2 for a description of the method. In fact, an inadequate value q can lead to either a too complex model or an underestimated model order. On one hand, if q is too small, the model order is underestimated and the signal is not fully reproduced. On the other hand, if q is too big, the model order is overestimated and the PRONY model provides exponentials which do not represent the signal. Additionally, note that the range of validity of q depends on the signal length, N.

4.3.1.4.2.1. PRONY model estimation results

The estimation results of the PRONY model are presented for different parameters of N and q. More specifically, for N equal to one week, 84 samples, the selected q values are 46, 50 and 62. These values have been chosen because they provide the better estimation results when one week is used to estimate the difference signal. And for N equal to two weeks, 168 samples, the selected q value is equal to 120. This value has been chosen because they provide the better estimation results when two weeks are used to estimate the difference signal. Additionally, note that the results were unsatisfactory for N equal to 3 weeks.

We present the same type of table as for the Blackman-Tuckey prediction performance. In this case, Table 4-14, Table 4-15 and Table 4-16 present the bit of most weight different from 0 of the DIFF difference value which results from the subtraction between the eccentricity of the broadcasted ephemeris data of the satellite 1 and the eccentricity of the almanac data plus the estimation of the difference eccentricity signal using the PRONY model.

Eccentricity (32 bits)														
Sample Position*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*
Almanac simplification	21	21	23	20	21	21	20	20	20	21	22	21	19	20
PRONY (N=84, q=46)	23	19	25	19	20	22	22	19	19	20	24	21	21	23
PRONY (N=84, q=50)	22	20	20	24	19	19	20	22	23	21	24	25	22	24
PRONY (N=84, q=62)	19	19	20	21	21	21	19	19	20	21	20	21	22	20
PRONY(N=168,q=120)	19	19	22	19	20	24	21	20	22	21	21	21	20	21

Table 4-14: Satellite 1 eccentricity Keplerian parameter PRONY model estimation results (part 1 of 3)

Eccentricity (32 bits)														
Sample Position*	*15*	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*	*24*	*25*	*26*	*27*	*28*
Almanac simplification	20	23	21	20	19	20	20	21	23	21	19	20	20	21
PRONY (N=84, q=46)	20	22	21	21	21	20	22	23	23	21	21	19	23	20
PRONY (N=84, q=50)	21	20	19	20	20	20	19	20	22	21	21	21	22	20
PRONY (N=84, q=62)	20	19	20	20	20	19	20	20	20	20	20	19	19	20
PRONY(N=168,q=120)	22	20	20	23	20	25	21	20	22	20	20	22	22	23

Table 4-15: Satellite 1 eccentricity Keplerian parameter PRONY model estimation results (part 2 of 3)

Eccentricity (32 bits)												
Sample Position*	*29*	*30*	*31*	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*	*40*
Almanac simplification	22	22	19	21	20	21	20	21	23	20	20	20
PRONY (N=84, q=46)	25	22	21	22	19	18	19	19	25	19	20	21
PRONY (N=84, q=50)	19	20	21	21	19	19	22	20	24	20	22	20
PRONY (N=84, q=62)	20	20	20	19	20	24	20	20	20	18	19	20
PRONY(N=168,q=120)	20	20	20	21	21	24	23	21	20	20	21	19

Table 4-16: Satellite 1 eccentricity Keplerian parameter PRONY model estimation results (part 3 of 3)

***Sample Position:** The sample position field marks the position of the estimated sample among the samples of the difference signal used to estimate the difference signal

From Table 4-14, Table 4-15 and Table 4-16, we can see that the performance of the binary estimation using the PRONY model is very low. In fact, from these tables, it can be observed that there is not always a bit gain using the addition of the estimated difference signal using the PRONY model to the almanac data. And, if there is some gain, it is not significant because the gain is never bigger than 5 bits.

4.3.1.4.2.2. PRONY model prediction results

The prediction results of the PRONY model are presented for the same N and q parameters as the estimation results. More specifically, for N equal to one week, 84 samples, the q values are equal to 46, 50 and 62. And for N equal to two weeks, 168 samples, the q value is equal to 120. These values are chosen because they provide the best estimation results.

The next tables show the bit of most weight different from 0 of the DIFF difference value which results from the subtraction between the eccentricity of the broadcasted ephemeris data of the satellite 1 and the eccentricity of the almanac data plus the prediction of the difference eccentricity signal using the PRONY model. The prediction of the difference signal means that the parameters characterizing the PRONY model were estimated from an earlier part of the difference signal.

Eccentricity (32 bits)														
Sample Position*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*
Almanac simplification	16	16	16	17	17	16	16	17	17	17	17	17	16	16
PRONY (N=84, q=46)	16	17	16	16	16	16	16	16	16	17	17	17	16	16
PRONY (N=84, q=50)	16	16	16	17	17	17	17	16	17	17	17	17	16	16
PRONY (N=84, q=62)	16	17	16	17	17	16	16	16	17	17	17	17	16	16

Table 4-17: Satellite 1 eccentricity Keplerian parameter PRONY model prediction results (part 1 of 6)

Eccentricity (32 bits)														
Sample Position*	*15*	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*	*24*	*25*	*26*	*27*	*28*
Almanac simplification	16	17	17	17	17	17	17	17	18	18	16	17	17	17
PRONY (N=84, q=46)	16	17	17	17	16	17	17	17	18	18	16	16	17	17
PRONY (N=84, q=50)	16	17	17	17	16	17	17	17	18	18	16	17	17	17
PRONY (N=84, q=62)	16	16	17	17	16	17	17	17	18	17	16	16	16	16

Table 4-18: Satellite 1 eccentricity Keplerian parameter PRONY model prediction results (part 2 of 6)

Eccentricity (32 bits)												
Sample Position*	*29*	*30*	*31*	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*	*40*
Almanac simplification	17	17	17	17	18	17	19	19	17	17	18	17
PRONY (N=84, q=46)	17	17	17	17	18	18	20	21	17	17	18	18
PRONY (N=84, q=50)	18	18	17	17	18	18	19	19	17	17	18	18
PRONY (N=84, q=62)	17	17	17	17	17	17	19	19	17	17	17	17

Table 4-19: Satellite 1 eccentricity Keplerian parameter PRONY model prediction results (part 3 of 6)

Eccentricity (32 bits)														
Sample Position*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*
Almanac simplification	19	20	19	21	20	20	19	25	20	21	20	20	19	24
PRONY(N=168,q=120)	18	22	19	20	19	19	19	23	20	20	21	20	20	20

Table 4-20: Satellite 1 eccentricity Keplerian parameter PRONY model prediction results (part 4 of 6)

Eccentricity (32 bits)														
Sample Position*	*15*	*16*	*17*	*18*	*19*	*20*	*21*	*22*	*23*	*24*	*25*	*26*	*27*	*28*
Almanac simplification	20	21	21	20	19	20	27	21	22	20	19	20	21	20
PRONY(N=168,q=120)	19	21	19	19	19	21	19	21	23	21	19	23	22	21

Table 4-21: Satellite 1 eccentricity Keplerian parameter PRONY model prediction results (part 5 of 6)

Eccentricity (32 bits)												
Sample Position*	*29*	*30*	*31*	*32*	*33*	*34*	*35*	*36*	*37*	*38*	*39*	*40*
Almanac simplification	21	21	19	19	19	20	21	22	19	19	19	19
PRONY(N=168,q=120)	24	20	20	21	23	22	19	20	18	18	18	18

Table 4-22: Satellite 1 eccentricity Keplerian parameter PRONY model prediction results (part 6 of 6)

***Sample Position:** The sample position field marks the position of the predicted sample after the last sample of the difference signal used to estimate the difference signal

From Table 4-17 to Table 4-22, it can be observed that the PRONY model prediction performance is as bad as the PRONY model estimation performance. In fact, the PRONY prediction is hardly able to gain more than 2 bits. Nevertheless, the low PRONY prediction performance level could be expected since the prediction process is much harder to achieve than the estimation process. Therefore, it can be concluded that the PRONY model cannot serve as a prediction technique for the difference signal resulting from the application of the simplification process using almanac data. Moreover, this bad performance implies that the signal is not really formed by exponentials and/or the signal is too noised.

4.3.1.5. Artificial neural network

In this section, we present the last method implemented in order to try to predict the difference signal resulting from the application of the simplification process using almanac data. This method is the artificial neural network. The artificial neural network method is the most advanced and the most complex technique analyzed in this study, and thus it should a priori provide better performance than the spectral estimation and PRONY model methods. Moreover, one important advantage of the neural network method compared to the other methods is that this method does not specifically target the spectral components or the exponentials forming the signal. In fact, an artificial neural network tries to predict the signal from several past signal samples and some internal components called neurons. Therefore, if the reason responsible of the unsatisfactory spectral estimation and PRONY model results is that the signal is not formed by a sum of sinusoids, the neural network method should overcome this impediment.

In this section, first the definition of the neural network is presented. Second, the definition of the fractal dimension is given and the fractal dimension of the difference signal resulting from the application of the simplification process using almanac data is calculated. Last, the prediction results obtained with the neural network method application are presented. The structure of the neural network and its learning phase are not described in this section but a detailed description can be found in annex G.7.

This method was implemented with the help of two ENAC undergraduate students, V. BARREAU and X. BOULANGER.

4.3.1.5.1. Neural Network definition

An artificial neural network (ANN) or neural network (NN), is a mathematical model or computational model that tries to simulate the functions and the structure of biological neural networks. An artificial neural network consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. In most cases an ANN is an adaptive system that changes its structure based on external or internal information that runs through the network during the learning phase. Neural networks are non-linear statistical data modeling tools. They can be used to model complex relationships between inputs and outputs or to find patterns in data [BHADESHIA, 1999] [YEGNANARAYANAa, 2006].

A neural network scheme is illustrated below. It can be observed that a neural network is an interconnected group of nodes, similar to the vast network of neurons in the human brain [BHADESHIA, 1999].

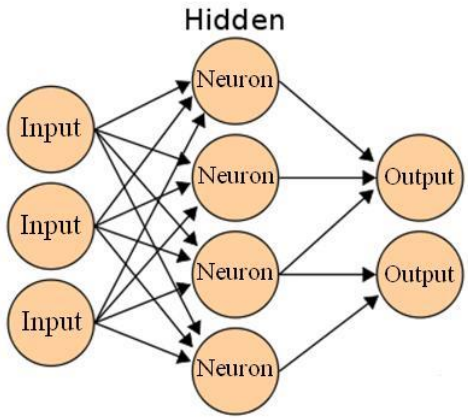

Figure 4-3: Neural network scheme

Figure 4-3 presents a non recurrent neural network scheme, where the neuron outputs only depend on external inputs, which are the signal delayed samples. However, if some or all the neuron outputs are used as a neuron inputs, the neural network is called recurrent neural network. Nevertheless, we only implement non recurrent neural networks.

In our case, we use a neural network which uses as inputs the known samples of the difference signal in order to predict the future samples of the difference signal which are unknown. More specifically, the known samples of the signal used as inputs are the last known samples of the difference signal.

A description of the structure of a neural network and of its learning phase is given in annex G.7.

4.3.1.5.2. Fractal dimension

In annex G.7, the typical structure of a neural network is given and its main parameters are defined. Besides, the method responsible for obtaining the value of these parameters, known as learning phase, is also presented. However, two parameter values still remain to be determined. These parameters are the total number of neurons and the number of neural network inputs, or, in other words, the number of delayed samples used to predict the future signal sample. Remember that the neural network that we are trying to implement uses the last known samples of the difference signal in order to predict the first unknown sample of the difference signal.

The number of neurons is arbitrarily chosen but it is always larger than the number of delays. The number of delays has to be carefully chosen because it is the most important neural network parameter. In fact, if the number of delays is underestimated the neural network is not able to correctly predict the signal; however, the larger the number of delays is, the more complex the neural network becomes and the more difficult its implementation is. Besides, a neural network with more than 10 delays is too complex and is considered impossible to implement [HAYKIN, 1994].

Therefore, in order to determine the number of input delays, we search for the fractal dimension of the difference signal resulting from the application of the simplification process using almanac data. The fractal dimension, *D*, is a statistical quantity that gives an indication of how completely a fractal appears to fill space, as one zooms down to finer and finer scales [GRASSBERGER and PROCACCIA, 1983]. And, from the fractal dimension, we can find the number of delayed inputs.

$$Number\ Delays = 2*(Fractal\ Dimension) + 1 \tag{4-2}$$

From equation (4-2), it can be seen that the fractal dimension determines the number of delays necessary to implement a neural network which can predict the desired signal, or temporal series in our case. This means that the temporal series to be predicted has to have a low fractal dimension in order to allow its prediction. Nevertheless, note that a low fractal dimension implies the possibility that the temporal series can be predicted but it does not guarantee its predictability.

We use a specific definition of the fractal dimension widely applied for the temporal series in order to determine the number of required delays. This definition is known as the correlation dimension, D_{cor}, and can be calculated by [GRASSBERGER and PROCACCIA, 1983]:

$$D_{cor}(\varepsilon) \approx \frac{\ln\left(C(\varepsilon)\right)}{\ln\left(\varepsilon\right)} \tag{4-3}$$

$$C(\varepsilon) = \lim_{N\to\infty} \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \theta\left(\varepsilon - \left\| x_i - x_j \right\|\right) \tag{4-4}$$

$$\theta(x) = \begin{cases} 0 & x \le 0 \\ 1 & x > 0 \end{cases} \tag{4-5}$$

$$x_i = \{x[i], x[i-1], \ldots, x[i-d+1]\} \tag{4-6}$$

Where:

- N: number of signal samples
- x[i] signal sample at instant i

- d: plunging dimension

Therefore, we need to calculate the $D_{cor}(\varepsilon)$ for all possible ε values when d is fixed. And we need to calculate the linear regression of all the resulting $D_{cor}(\varepsilon)$ values when d is fixed in order to find the fractal dimension, D, for a given d value [GRASSBERGER and PROCACCIA, 1983]. The linear regression expression is:

$$D = \frac{\mathrm{cov}\left(\ln\left(C(\varepsilon)\right), \ln(\varepsilon)\right)}{\mathrm{var}\left(\ln(\varepsilon)\right)} \tag{4-7}$$

This process is repeated for different d values in order to remove the influence of the plunging dimension on the fractal dimension calculation.

A table showing the fractal dimension of the difference signal history of the satellite 1 eccentricity resulting from the application of the simplification process using almanac data is shown below.

Eccentricity fractal dimension (D)		**Plunging dimension (d)**				
		2	*3*	*4*	*5*	*6*
N	*1000*	0.68	0.52	0.46	0.42	0.42
	2000	0.62	0.47	0.42	0.38	0.37

Table 4-23: Fractal dimension of the difference signal history of the satellite 1 eccentricity resulting from the application of the simplification process using almanac data

From Table 4-23, and approximating the fractal dimension by its immediate superior number, it can be calculated from equation (4-2) that our neural network requires at least a minimum number of delayed inputs equal to 3. This means that we can try to predict the difference signal of the satellite 1 eccentricity resulting from the application of the simplification process using almanac data because its fractal dimension is small. Nevertheless, remember the fractal dimension only gives an indication; the difference signal can still be not predictable.

4.3.1.5.3. Results

In this section, the prediction results obtained by the application of the neural network method should be presented. However, even trying different combinations between the number of input delays and the number of cells, this study has not been able to find a neural network able to predict the difference signal. In fact, for the different neural network configurations the error power never converges and thus the neuron weight values, the activation thresholds values, etc, cannot be determined.

This fact can be seen in Figure 4-4 and Figure 4-5, which represent the MLP neural network prediction error when the network tries to predict the difference signal of the satellite 1 eccentricity resulting from the simplification process using almanac data. In these figures, the learning rate (μ) is always equal to 0.0001, the X axis shows the first iteration of each consecutive group of 100 iterations, and the Y axis represents the absolute normalized error value (*ANE*).

$$ANE[n] = \left| \frac{x[n] - \hat{x}[n]}{x[n]} \right| \tag{4-8}$$

Where:

- $x[n]$: Sample to be predicted
- $\hat{x}[n]$: Predicted sample

Figure 4-4 and Figure 4-5 represent the different MLP neural network configurations. The Y axis is limited to 60.

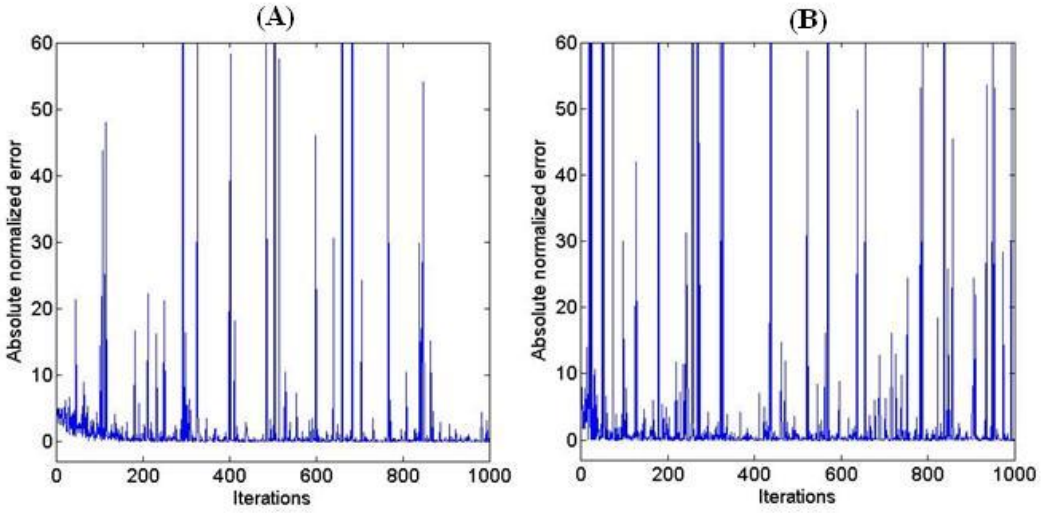

Figure 4-4: *ANE* between the neural network prediction and the difference signal of the satellite 1 eccentricity resulting from the simplification process using almanac data. (A) Delays = 3, Neurons = 10. (B) Delays = 5, Neurons = 15.

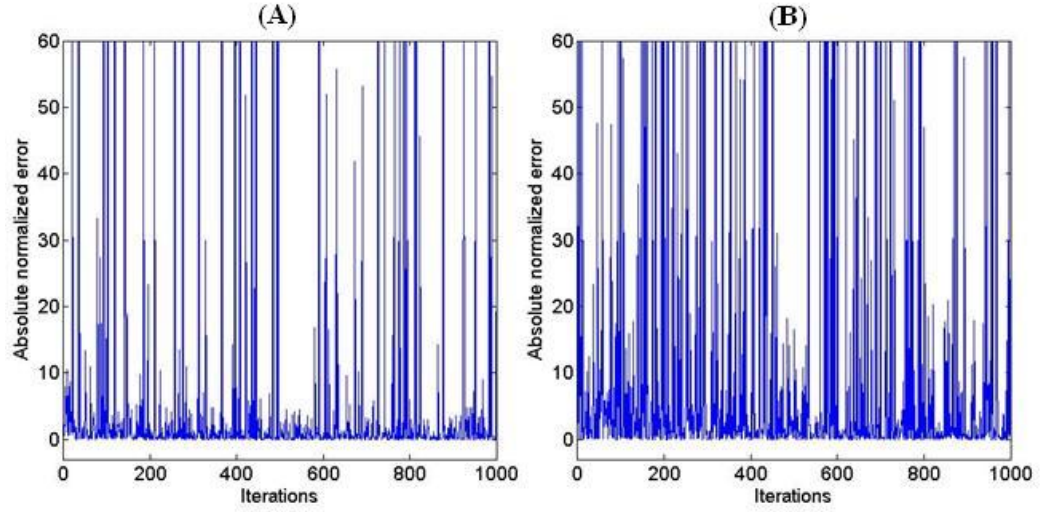

Figure 4-5: *ANE* between the neural network prediction and the difference signal of the satellite 1 eccentricity resulting from the simplification process using almanac data. (A) Delays = 7, Neurons = 30. (B) Delays = 10, Neurons = 50.

From Figure 4-4 and Figure 4-5, it can be observed that the *ANE* does not converge but rather exhibits several huge errors. In fact, the number of wrong predictions increases along with the number of assumed delays in the neural network. Therefore, it can be suspected that the error is due to bad inputs, which are obviously more numerous for a larger number of delayed input configurations. Consequently, we tried to run the learning phase without varying the neural

networks parameters when the ANE between the predicted sample, $\hat{x}[n]$, value and the known value used to train the neural network, $x[n]$, is larger than 1. Figure 4-6 shows the results.

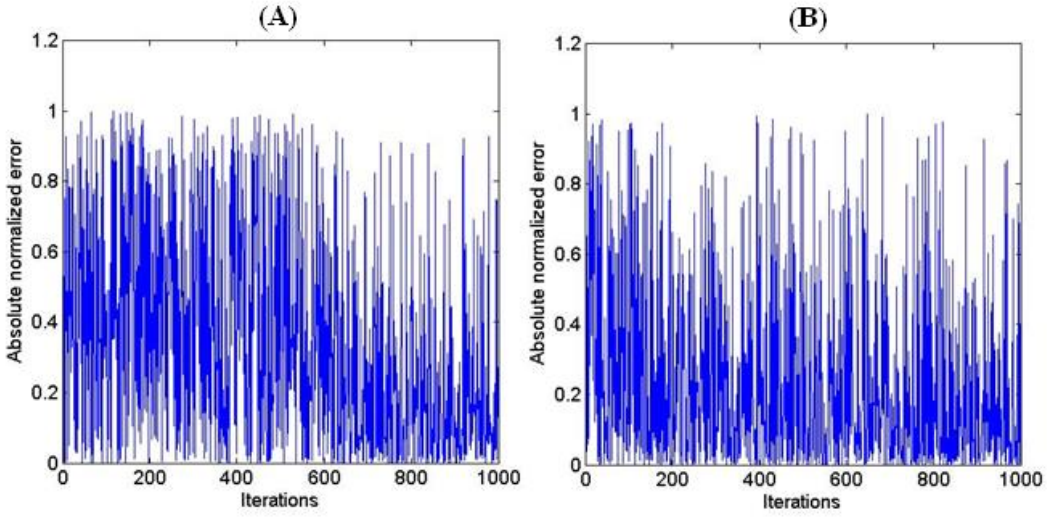

Figure 4-6: ANE between the neural network prediction and the difference signal of the satellite 1 eccentricity resulting from the simplification process using almanac data when the prediction error is limited to 1. (A) Delays = 3, Neurons = 10. (B) Delays = 5, Neurons = 15.

From Figure 4-6, it can be observed that although we eliminated from the learning phase all the predicted samples $\hat{x}[n]$ having an *ANE* larger than 1, the neural network still does not converge. Therefore, we can conclude that either the neural network cannot predict the difference signal or the history contains too many errors. And the history may contain too many errors because the ephemeris data sets are not always broadcasted every two hours and they can sometimes be badly calculated by the GPS control segment.

Moreover, we also tried a RBF neural network, obtaining the same results as for the MLP neural network. The absolute normalized error does not converge even when the bad inputs were excluded from the learning phase. And the same conclusions have been found for the other Keplerian parameters.

4.3.2. Spatial methods

In this section, the spatial methods used in order to predict the GPS L1 C/A ephemeris data sets are presented. The main idea of the spatial methods is to use the spatial correlation of the GPS satellites orbits. Or, in other words, the spatial methods seek to exploit the fact that some GPS satellites share a priori the same orbit. Therefore, in this section, we use the broadcasted ephemeris data of a satellite in order to predict the future ephemeris data of another satellite which follows the same orbit but with a position delayed in time. The GPS satellite constellation as of 15-12-2009 is given in Table 4-24, where for each orbit plane and orbit position the satellite PRN number is presented.

However, this type of methods has not been fully inspected because the preliminary analyses have not been promising. Moreover, in addition to these first studies, the extremely high accuracy required by the binary prediction has finally leaded us to determine that the spatial methods cannot provide a satisfactory binary prediction performance.

		Satellite Orbit Plane					
		A	B	C	D	E	F
Satellite Orbit Position	1	09	16	29	02	20	14
	2	31	30 / 01	03	11	22	15
	3	08	28	19	21	10	13
	4	27	12	17	04	18	23
	5	25	--	06	24	32	26
	6	07	--	--	--	05	--

Table 4-24: GPS satellite constellation

The first method analyzed is to use a satellite broadcasted ephemeris data set as the future ephemeris data set of another satellite having the same orbit but with its position delayed in time. Therefore, the first step is to calculate the delay between the satellites. Once this delay is found, the corresponding ephemeris data sets of both satellites are compared. The results are not good mainly for two reasons. The first reason is that the delay between satellites does not correspond to the time difference between broadcasted ephemeris data sets. Therefore, the method can compare two ephemeris data sets which have an application difference time of 4h when the satellites are really delayed by only 3h30. The second reason is that the broadcasted Keplerian parameters values have to represent the satellite orbit over a period of time. Therefore, the broadcasted Keplerian parameter value may be the result of an average or a linear interpolation, and thus the value varies depending on the represented orbit interval. And since the delay between satellites is not the same as the difference between the time of application of the broadcasted orbits, the orbit interval represented is not the same. Therefore, the first method is directly discarded.

From the conclusions of the first method, a second method can be considered. This method consists in generating from the ephemeris data of the satellite advanced in time the ephemeris data of the satellite delayed in time which defines the satellite orbit over the desired period of time. Therefore, since the broadcasted ephemeris data of the satellite delayed in time and the predicted ephemeris data define the same part of the satellite orbit, the binary prediction should be better. A detailed explanation is given below.

The 2nd method consists in generating from the broadcasted ephemeris data of the satellite advanced in time, the XYZ points of the orbit of the satellite delayed in time over the validity period of time of the ephemeris data to be predicted. Once the orbit has been defined over this validity period of time, the second method calculates the instantaneous ephemeris data of this period, where an instantaneous ephemeris data is an ephemeris data set which defines one and only one XYZ point of the satellite orbit. This calculation is done by applying a perfect transformation between a XYZ coordinates set of the satellite orbit and the Keplerian parameters which define an ideal orbit passing by this defined XYZ point. An ideal orbit is an orbit which is not disturbed by any external influences, such as the moon and sun attraction, solar wind, flattening earth effect, etc. This means that an ideal orbit is defined by the Keplerian parameters of the almanac data and that not all the Keplerian parameters of the ephemeris data can be predicted, only the parameters which are common to the almanac data. Finally, since the Keplerian parameters of the broadcasted ephemeris data are a kind of average of the real instantaneous ephemeris data (the 15 Keplerian parameters of the ephemeris data), the second method calculates either an average or a linear interpolation from

the previous calculated ideal instantaneous ephemeris data (Keplerian parameters of the almanac data).

Nevertheless, before applying this method to two satellites having the same orbit, a preliminary study has been conducted. This study consists in applying the second method in order to predict the n^{th} satellite broadcasted ephemeris data set from the n^{th} satellite broadcasted ephemeris data set. Or, in other words, this preliminary study analyzes the transformation performance. Therefore, if the method is not able to perfectly recover the broadcasted Keplerian parameters of a satellite from the same broadcasted Keplerian parameters of this satellite, we can guarantee that the second method cannot predict a future ephemeris data set.

Unfortunately, the method has not succeeded in recovering the ephemeris data and one possible reason of the failure is that we do not know the exact transformation applied by the GPS control segment. Moreover, it is possible that the transformation takes into account the first order correction terms such as C_{uc} or C_{rc}. Therefore, the method should try to use these parameters; however, as seen in section 4.3.1.1.4, the TAS transformation program does not provide the desired ephemeris data.

Finally, one last reason for why the preliminary spatial methods tests have failed and thus their further inspection is not advised is that all satellites orbits differ too much. Theoretically, the GPS system is designed to have groups of satellites sharing the same orbit. However, in reality, they orbit is different enough to make all the attempts fail to exploit the assumed satellite spatial correlation. Some figures illustrating this statement are presented below.

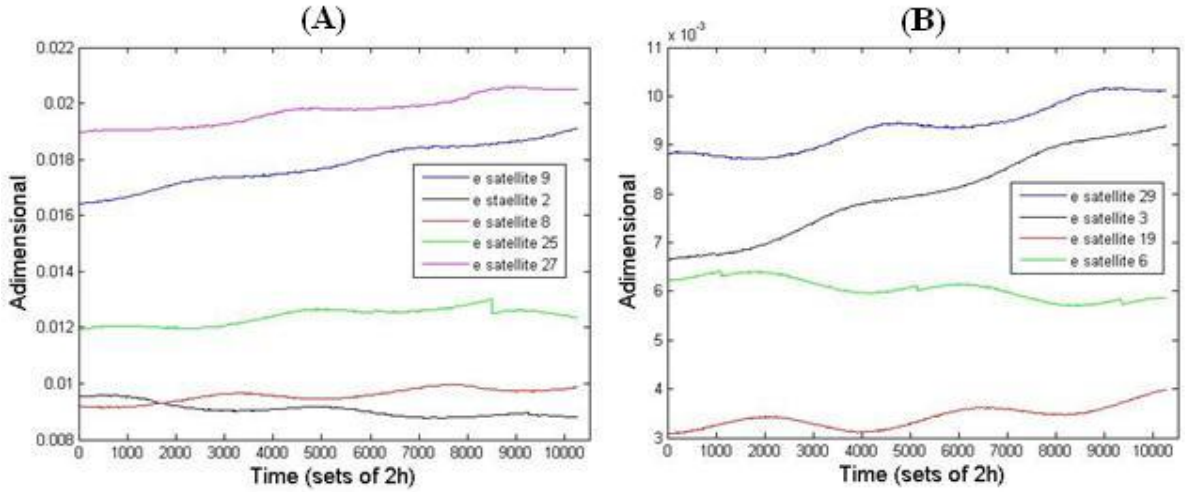

Figure 4-7: GPS satellites eccentricity. (A) Satellites plane A. (B) Satellites plane C

From Figure 4-7, it can be observed that although the satellites are in the same orbit plane and thus should have the same eccentricity, their eccentricity is different. In fact, it can be observed that even the time evolution is different. However, there are other Keplerian fields' values, such as the inclination, that are not so different among satellites of the same plane. Figure 4-8 shows an example.

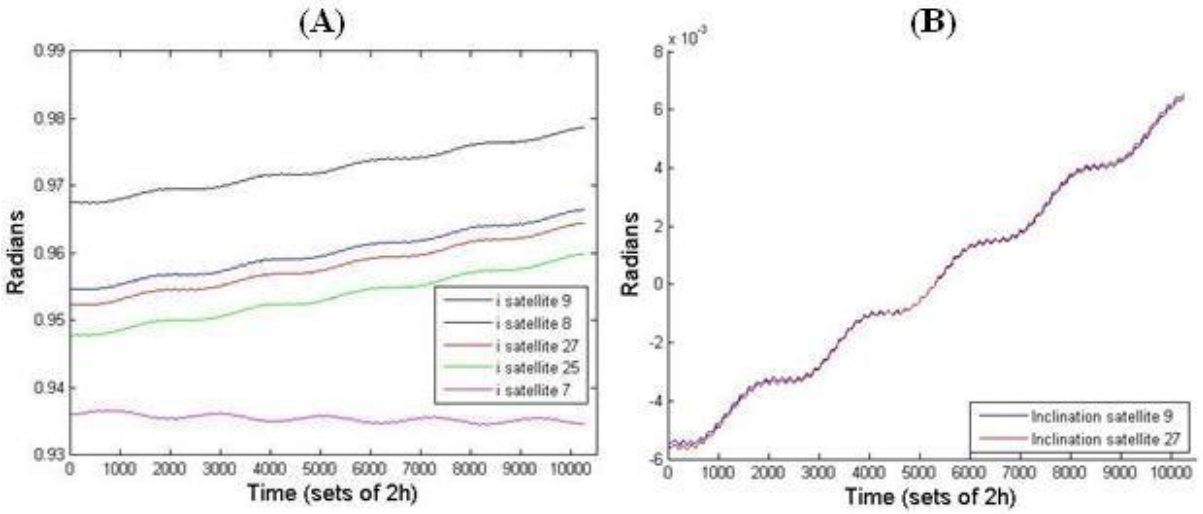

Figure 4-8: GPS satellites inclination. (A) Satellites plane A. (B) Satellites 9 and 27 without mean value

From Figure 4-8, we can see that several satellites from plane A appear to have the same inclination evolution in time. Therefore, we could consider implementing a method which will translate one inclination parameter to the other since it seems to be only an offset. Nevertheless, we first need to verify this similitude. Therefore, in order to verify the time evolution, we have picked two satellites, 9 and 27, and we have subtracted the mean value to their broadcasted inclination. The result can be seen in Figure 4-8B. From this figure, we can observe that the evolution is not exactly the same because satellite 27 begins with a lower inclination value but finishes with a larger one. We also make a zoom in order to closely inspect the evolutions of the two inclinations.

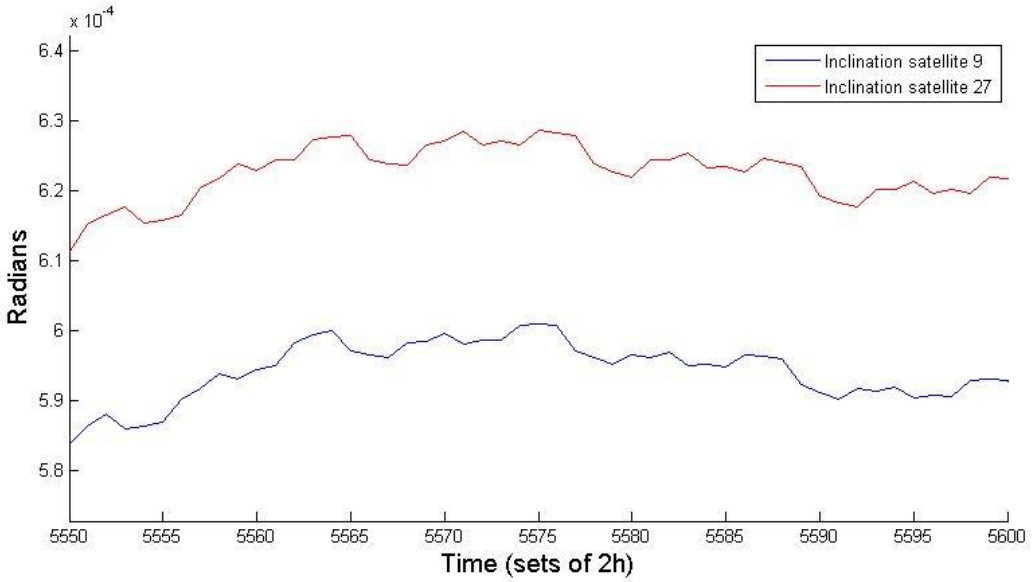

Figure 4-9: Temporal evolution of the inclination of satellites 9 and 27

From Figure 4-9, we can see that the time evolution between the two inclination values is similar but not identical. Therefore, since the binary prediction requires an extremely high accuracy, we cannot use one Keplerian parameter value to predict the same parameter for another satellite into the same plane.

To sum up, due to the difference between broadcasted Keplerian parameters of satellites sharing a priori the same orbit, the difference between satellite delays and their broadcasted ephemeris validity interval, and the bad performance of the instant XYZ orbit coordinates transformation to perfect Keplerian parameters, the spatial methods are not suited to provide a satisfactory binary prediction performance.

4.4. Conclusions

The main conclusion that can be extracted about this chapter is that the binary prediction is very hard to achieve. In fact, from the different options and methods analysed during this chapter, the broadcasted ephemeris binary prediction can only be achieved by using external aid, such as the broadcasted almanacs or long term predicted ephemeris. And, in any case, a good binary prediction performance is limited to about half or less the binary word information. Different reasons explain this great difficulty in obtaining a satisfactory prediction.

The first reason is the accepted probability of error. From section 4.2.4, the probability of wrong predicting a bit has been calculated and justified. This probability is very low due to the necessity of well predicting all the ephemeris bits in order not to commit a user position error larger than 1m. Remember that this threshold has been fixed in order to make an error smaller than the standard deviation of the ephemeris data error. Therefore, the prediction methods must have a very good performance in order to fulfil the requirement of the case of study. Nevertheless, a more realistic approach is that only the bit predictions having 100% chances of success are employed. Therefore, the case of study limits the binary prediction to the bits which are always perfectly predicted, which means a very low number of predicted bits.

The second reason is the special relationship between the binary and the decimal domain. This relation or influence can be separated into two components. The first component is the different law of growth in each domain. In fact, the linear growth of the value in the binary domain, for example one more bit to represent the value, is equivalent to an exponential growth in the decimal domain. Therefore, the additional prediction of one bit results into a required decimal prediction two times more accurate than before. Consequently, the prediction of a significant number of bits requires a really high accuracy. The second component is the fact that even the wrong prediction of the least significant bit can modify the value of all the bits of the number. Therefore, the binary format in itself presents two intrinsic problems which cannot be solved. And these problems make the binary prediction really hard to achieve and, what is worse, not reliable.

The third reason is the poor performance provided by the different methods examined during this chapter. The spatial methods have shown to be not really exploitable because the satellites do not share the same orbit and the programs transforming the XYZ orbit coordinates to the ephemeris data are not aimed at the binary prediction. The spectral estimation methods and the PRONY model have presented gains of 1 or 2 bits which is not very surprising if we consider the level of accuracy required to predict more than 4 bits. However, the neural network has provided disappointing performance despite of being the most general method. In fact, this method is not able to find a neural network structure which predicts the signal and thus it can be thought that this signal is not predictable. The reason can be a bad history where the sampling period is not constant and the fact that the samples present too many errors.

Therefore, the only reliable manner of predicting some of the broadcasted ephemeris data bits is to employ external information such as the almanac data or the long term predicted ephemeris. This predicted ephemeris data has to be calculated by an external computer since the implemented program needs a very powerful CPU to run. Moreover, the predicted ephemeris data provided by the TAS program has resulted to be a different data set from the GPS L1 C/A broadcasted ephemeris. However, this fact does not ensure that other long term orbital prediction programs are not capable of providing the GPS L1 C/A broadcasted ephemeris although that seems highly improbable. And, in any case, these programs face the problems presented as reasons 1 and 2 for failure. Additionally, we presented in this chapter a method which allows the exploitation of the decimal value, called DIFF difference, instead of the exploitation of the binary value, called XOR difference.

Finally, since the exact procedure used by the GPS L1 C/A control segment to calculate the ephemeris data is unknown and thus it is impossible to calculate an ephemeris data set equivalent to the GPS L1 C/A ephemeris data set for the XYZ positions of the orbit of a satellite, another solution has to be proposed. This solution would be to transmit in advance the future ephemeris data sets. In fact, the satellite ephemeris data sets are calculated one day in advance because each satellite receives an ephemeris data update each day. Therefore, since the ephemeris data is known in advance, it could be transmitted before through the satellite or using other platforms such as mobile telephone structures.

Chapter 5. Improved methods to decode GPS L2C and GPS L5 signals

The GPS L2C and GPS L5 signals obtain a good demodulation performance in open environments where the received C/N_0 is high. However, in this dissertation, we are rather interested in the signals demodulation performance in urban or indoor environments where the available C/N_0 is low. In these conditions, the BER value obtained by traditional demodulation and/or decoding techniques may not fulfill the application requirements. Therefore, the objective of this chapter is to improve the GPS L2C and GPS L5 decoding schemes. Or, in other words, our aim is to decrease the level of C/N_0 required to obtain a given BER after the GPS L2C or GPS L5 navigation message decoding process. In this chapter, two different methods are presented in order to improve the GPS L2C and GPS L5 navigation message decoding process.

Nevertheless, note that the GPS L5 signal is targeted to civil aviation purposes where the available C/N_0 is always high. Therefore, the GPS L5 navigation message inclusion in this chapter can a priori appear unnecessary. However, there is one step during an airplane flight where these methods can be applied. This airplane flight step is the parking part where the hangars and other airplanes can decrease the received C/N_0 level. Moreover, these methods were originally conceived to improve the GPS L2C signal BER in urban or indoor environments, but, since the GPS L5 signal has the same navigation message structure as the GPS L2C mode CNAV data message, this former signal can take advantage of one of the proposed methods.

The BER improvements introduced by these new two methods are analyzed in different situations. The methods are tested into the presence of an AWGN channel with no tracking error sources, an AWGN channel with thermal noise tracking error and an AWGN channel with thermal noise and dynamic stress tracking errors.

In this chapter, first the problem to be solved is presented. Second, two different methods are described and their performance calculated through simulations is presented.

5.1. Presentation of the problem

The objective of this chapter is to decrease the C/N_0 level required to obtain a given BER. The particularity here is that the signal structure and the navigation message structure are already established. The signal power, the symbol transmission rate, the modulation, the channel code, etc, have been already designed and we do not have any possibility of modifying them. Therefore, the real challenge is to improve the decoding process with the existent elements.

In this research work, two options are found in order to decrease the C/N_0 level required to obtain a given BER. The first option is to use external information. The second option is to use the navigation message structure components in a different way than the planned one but without modifying their final function.

5.2. Proposed methods

In this section, the two decoding methods proposed by this dissertation in order to improve the GPS L2C and GPS L5 BER are presented.

The first method uses external aid in order to improve the decoding process. This external aid is the navigation message ephemeris statistics which provide a priori bit probabilities. Therefore, the first method is simply the traditional convolutional decoding algorithms explained in annex E.3.2 but including the a priori ephemeris bit probabilities.

The second method modifies the functioning of the GPS L2C and GPS L5 channel codes. This method changes the typical decoding strategy and implements a new one. Nevertheless, this method is transparent from the point of view of the receiver functions before and after the channel code.

Another difference between the two proposed methods is the targeted navigation message. The first method only improves the decoding performance of the GPS L2C mode NAV with FEC data message, whereas the second method improves the decoding performance of this former navigation message as well as the GPS L2C mode CNAV data message and GPS L5.

The decoding performance of both proposed methods is presented at the end of the chapter and is compared with the decoding performance of the Viterbi algorithm.

5.2.1. Inner code decoding with a priori bit probabilities

In this section, the method improving the decoding performance of the GPS L2C mode NAV with FEC data message is presented. This proposed method consists of using the a priori bit probabilities. The two traditional convolutional decoding techniques can be employed.

First, we present the modified Viterbi method, or, in other words, the modifications introduced to the Viterbi algorithm [VITERBI, 1967] [VITERBI and OMURA, 1979] in order to take advantage of the a priori bit probabilities. Second, the BCJR [BAHL et al., 1974] algorithm, which already includes the possibility of using the a priori bit probabilities, is also presented in order to inspect which method is farther improved by the a priori bit probabilities utilization.

In this section, first the main idea of the general method is given and second, the data from which the a priori bit probabilities are calculated is presented. Third, the modified Viterbi is described and fourth, the exact BCJR algorithm part where the probabilities have to be used is given.

5.2.1.1. Main idea

The main idea of method is to introduce the a priori bit probabilities from the known message in order to help the convolutional decoding algorithm. The introduction of these a priori probabilities makes the decoder to be more prone to decode certain bit values than others ones.

As the use of a priori known bit probabilities is part of the optimal ML demodulation expression, see annex C.1 and equation (C-4), it is normal that the a priori bit probabilities introduction improves the Viterbi and the BCJR algorithms.

5.2.1.2. A priori bit probabilities source

In our case, the employed a priori bit probabilities are the probabilities of the bits representing the ephemeris of the GPS L1 C/A signal. Therefore the only message which can make use of the first method is the GPS L2C mode NAV with FEC data message.

These probabilities have been calculated from the same history employed in section 4.3.1. The method used to calculate the probabilities is explained in annex F.3 altogether with the table summarizing these probabilities.

5.2.1.3. Modified Viterbi

The modifications introduced in the Viterbi algorithm in order to take into account the a priori bit probabilities are found after reviewing the fundamental expression of the Viterbi algorithm. The Viterbi algorithm has been described in section E.3.2.2.

The introduction of the a-prori bit probabilities implies the addition of an extra term to the transition distance between states of the decoding trellis. The exact calculation of this term has been developed in annex E.3.2.4. The additional term is equal to:

$$Additional\ transition\ distance\ term = -2\sigma^2 \log\left(P\left(s_k^{(m)}\right)\right) \tag{5-1}$$

Where:

- m : Number of symbols of the source X alphabet
- $s^{(m)}$: Information sequence of K elements from source X alphabet
- $s_k^{(m)}$: k^{th} information symbol of the $s^{(m)}$ sequence
- σ^2 : AWGN variance

Note that this additional term depends on the noise power and on the a priori bit probability at instant k. Therefore, the new transition distance is:

$$Modified\ Transition\ Distance = \sum_i \left|r_{ki} - c_{ki}\right|^2 - 2\sigma^2 \log\left(P\left(s_k^{(m)}\right)\right) \tag{5-2}$$

Where:

- r_k: k^{th} element or estimated coded symbol of the estimated coded sequence
- c_{ki}: i^{th} component of the k^{th} estimated coded symbol
- c_k: k^{th} coded symbol of the coded sequence
- c_{ki}: i^{th} component of the k^{th} coded symbol

To sum up, the modified Viterbi algorithm is exactly the same as the original one but using equation (5-2) in order to calculate the distance transition between states.

5.2.1.4. Modified BCJR

The BCJR algorithm is already suited to the use of a prori bit probabilities. Therefore, in this section, only the part of the algorithm where the a priori probabilities have to be applied on is explained.

Recovering the BCJR algorithm published on [BAHL et al., 1974], the calculation of the gamma coefficient $\gamma_t(m',m)$ is the following.

$$\gamma_t(m', m) = \sum_X p_t\left(m|m'\right) \cdot q_t\left(X|m', m\right) \cdot R\left(Y_t, X\right) \tag{5-3}$$

Where, following the original paper notation:

- m: destination state
- m': initial state
- X: coded symbol
- Y_t: received symbol at the time t
- $R(Y_t, X)$: Probability of receiving Y_t at time t if X was transmitted.
- $q_t(X|m',m)$: Probability of generating X when the state transition is from m' to m at time t.
- $p_t(m|m')$: Probability of transiting from state m' to state m at time t.

Therefore, from equation (5-3) and its notation, it is clear that the $p_t(m|m')$ term represents the a priori symbol probability.

5.2.2. Combination of the inner and the outer channel codes

In this section, the second proposed method in order to decrease the C/N_0 required to obtain a given BER is presented. This method consists in using the navigation message inner and outer channel codes in a different way than usual. Or, in other words, the proposed method changes the traditional decoding strategy followed by GNNS receivers.

This method aims at improving the decoding performance of the GPS L2C signal, although it can only be employed on GPS L2C mode NAV with FEC data message and GPS L2C mode CNAV data message. The reason is that this new proposed method exploits the inner channel code Viterbi properties. This method can also be applied for the GPS L5 signal and the provisional defined GALILEO signal [ESA, 2008].

Since this proposed method changes the traditional decoding strategy, we present first this traditional strategy before defining the new idea of the proposed method. Then, this section continues by giving the method advantages and drawbacks, the pre-existing work, the algorithm basis, the algorithm main steps and the algorithm variations.

5.2.2.1. Traditional decoding strategy of a GNSS receiver

The traditional decoding strategy of a GNSS receiver can be divided into 3 mains steps:

1. Applying the inner code in order to correct the errors introduced by the message transmission: process known as Forward Error Correction (FEC).
2. Applying the outer code in order to detect which words/subframes have been wrongly decoded.
3. The words/subframes failing the outer channel code verification are discarded whereas the other words are considered free of errors and are saved.

Figure 5-1 summarizes the decoding strategy:

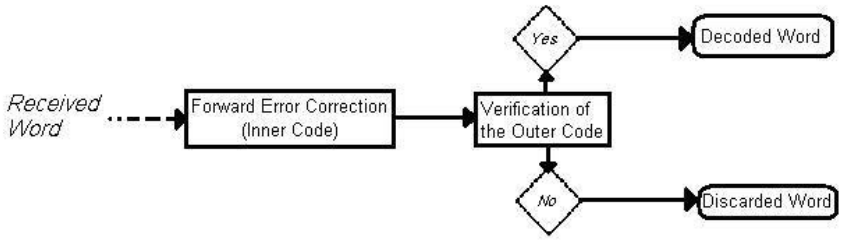

Figure 5-1: Traditional GNSS receivers decoding strategy

5.2.2.2. Main idea

The main idea of the proposed method consists in combining the inner and outer channel decoders in order to obtain a better decoding method.

This combination consists in using the property of the Viterbi algorithm of finding the most probable emitted sequence and of exploiting the outer channel code detection capacity. Therefore, this technique forces the receiver to decode the most probable coded sequence that succeeds the verification of the outer channel code.

Figure 5-2 summarizes the new decoding strategy.

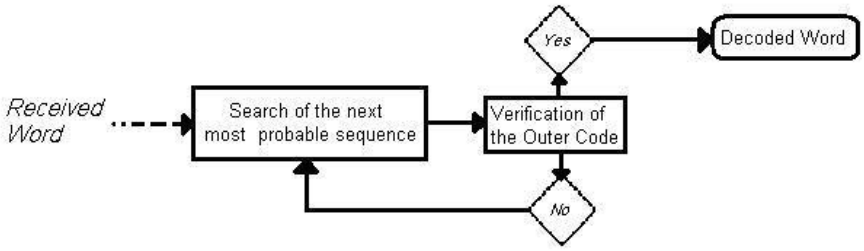

Figure 5-2: Traditional GNSS receivers decoding strategy

The new decoding strategy is slightly different from the strategy presented in Figure 5-2. The difference consists in setting a maximal number of sequences which are inspected before determining that the received sequence cannot be correctly decoded and before passing to decode the next received sequence. The reason for this variation is that depending on the C/N_0 level, the emitted sequence can be far from the most probable sequence determined by the Viterbi algorithm. Therefore, the receiver could take too much time and too many resources to search for the transmitted sequence.

5.2.2.3. Advantages and drawbacks

The main improvement obtained by this technique is the increase of the correctly decoded words. The reason of this increment is the continuous sequence search implemented by the receiver until one sequence verifies the outer channel code, instead of simply stopping the process and discarding the most probable sequence when this sequence fails the outer channel code verification.

This method introduces a drawback which impact depends on the final application requirements. This disadvantage is the method incapacity of discarding wrong sequences which succeed the outer channel code verification due to the continuous search of the most

probable sequence. In other words, the proposed method always provides an information sequence although this sequence can be wrong decoded, and thus, the number of wrong decoded words being used is incremented. Note that the impact of this drawback is decreased by the inclusion of a maximal number of sequences which are inspected before passing to decode the next received sequence.

A closer inspection of this disadvantage reveals that this drawback is introduced by the outer channel code limitations in terms of error detection capacity. Nevertheless, this limitation is shared by both decoding strategies since their implemented outer channel code is the same. Therefore, this fact could lead to the false assumption that the traditional strategy and the new strategy accept the same number of wrong decoded words. However, the difference between them is that the new technique is always forced to provide a decoded word, or to inspect much more sequences, whereas the traditional one is not, or only inspects one sequence. Therefore, the probability of accepting a wrong word is increased. Note that the probability of accepting a wrongly decoded sequence is much higher for extended Hamming code than for the CRC-24Q code.

5.2.2.4. Pre-existent Work

Although this technique was not extracted from any paper or publication, its main idea, *The List Viterbi with CRC*, was already studied and published by Seshadri [SESHADRI and SUNDBERG, 1994] and complemented by Leanderson [LEANDERSON and SUNDBERG, 2005]. Nevertheless, it has to be stated that whereas the method objective or main idea is the same, the algorithm implemented is different despite following the same fundamentals.

Moreover, there are some differences between the contexts where the methods are applied. In the Seshadri's case, the initial and final Viterbi states are known, e.g. the states are forced by tail bits, whereas the states are unknown in the GPS L2C navigation message case. And this context difference adds some new important complications to the problem. Nevertheless, all the new complications are commented and solved during the next subsections.

Finally, other works which analyzed proposed different decoding techniques to this type of concatenated channel codes and continued the work of [SESHADRI and SUNDBERG, 1994] are [NILL and SUNDBERG, 1995], [WEI, 2002] and [WANG et al, 2008].

5.2.2.5. Algorithm basis

The algorithm basis is to sort out the sequences into an ascending order with respect to their final accumulated distance beginning with the smallest accumulated distance sequence. Therefore, the sequence ordination begins with the most probable sequence, and continues, if necessary, until a sorted sequence succeeds the outer channel code verification.

The path with the k^{th} smallest accumulated distance is called the k^{th} candidate. Therefore, the algorithm continues to sort out the candidates until the last ordered candidate succeeds the outer channel code verification.

5.2.2.6. Algorithm main steps

The algorithm can be divided into 3 main steps being repeated until the 3^{rd} step fulfils a determined requirement. This requirement is that the sorted sequence succeeds the outer code channel verification. Therefore, considering that the accomplishment of this requirement can

take too much time, the algorithm can include the possibility of limiting the iteration number. In this case, the algorithm ends when the 3rd step is verified or when the maximal number of iterations is reached. A first general scheme of the algorithm main steps is given below.

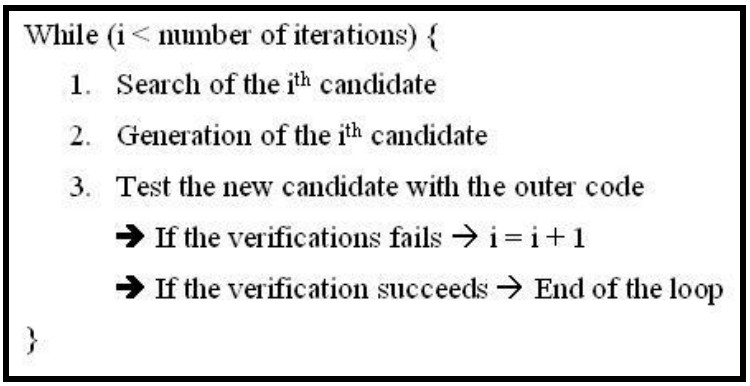

Figure 5-3: Algorithm main steps

The first two steps are explained in annex E.4. The third step is explained in annex E.1 and E.2.

5.2.2.7. Algorithm variations

The proposed algorithm accepts different variations that privilege some decoding aspects with respect to others. More specifically, 6 variations are commented in this research work.

The first variation consists in introducing the use of the a priori bit probabilities in the traditional Viterbi decoding process as explained in section 5.2.1.3. This variation is further commented in annex E.4.5.1.

The second variation is called sliding window and it can significantly increases the number of correct decoded words. However, this variation slows down the decoding process. This variation consists in decoding more than one code word at the same time by means of using a candidate which spans on several code words. This variation is further commented in annex E.4.5.2.

The third variation consists in imposing the initial state of the all the inspected candidates. This variation increases the number of times that the transmitted sequence can be found since it is more probable to impose the correct state than to impose an incorrect one. This variation is further commented in annex E.4.5.3.

The last 3 variations speed up the decoding process but, at the same time, can decrease the number of correct decoded words. These variations consists in discarding some sequences before verifying if they pass the outer channel code verification since the probability that they are the transmitted sequence is very low. These variations are further commented in annex E.4.5.4.

Finally, the inclusion of the algorithm variation for each analyzed signal is discussed at the end of each description of the algorithm variation.

5.2.2.8. Complexity of the decoder

The complexity of the decoder required by the decoding method proposed in this section has not been analyzed in this dissertation. However, some remarks can already be made. First, the

required decoder is more complex than a traditional Viterbi algorithm since it needs to execute several times the Viterbi algorithm onto the same sequence but imposing certain path to survive and it has to execute several times the verification of the outer channel code.

Moreover, the reader must remark that the complexity of a generic decoder for this proposed decoding method is significantly higher than the complexity of a LDPC decoder in terms of number of operations to execute. In fact, this proposed decoding method has been chosen because the structure of the GPS L2C and GPS L5 navigation messages were already defined. But given the opportunity to define a message, we will select a LDPC code or a turbo code instead of channel code scheme used in these navigation messages.

5.2.3. Performance of the proposed algorithms

In this section, the results of the Viterbi algorithm with a priori probabilities and the results of the method base on the inner and outer channel codes combination are presented. These results are presented in terms of BER, WER and EER as function of the signal E_b/N_0 or C/N_0 at the correlator output.

These results have been evaluated by simulations which characterize the transmission of the GPS L2C and GPS L5 signals through an AWGN channel. The chosen simulator is the first simulator presented in section 3.1.5, which means that the simulator directly generates the signal samples at the receiver correlator output.

Before presenting the results of both methods, the assumptions and conditions under which the simulations have been conducted are presented. Afterwards, first the results of the Viterbi algorithm with a priori probabilities and second the results of the method based on the inner and outer channel codes combination are shown.

5.2.3.1. Assumptions of the analysis

The assumptions taken into account for this analysis vary depending on the searched result. Concerning the signal carrier phase tracking, the assumption varies from an ideal phase tracking process to a non-ideal phase tracking process due to thermal noise and dynamic stress error. Moreover, all the simulations have been made assuming a seamless code tracking process and a seamless synchro-frame hypothesis. A further description of these last assumptions is given below.

5.2.3.1.1. Seamless code delay tracking process hypothesis

This assumption means that the DLL have obtained a perfect estimation of the code delay value. Therefore, the contribution of the code delay tracking process on the signal at the correlator output is removed. The I channel of the simulated signal at the correlator output can be modeled as:

$$r_i[k] = \frac{A}{2} \cdot d[k] \cdot \cos\left(\varepsilon_\theta[k]\right) + n_i[k] \tag{5-4}$$

Where:

- $r_i[k]$: Numeric correlator output signal of the I channel at epoch k
- $d_m[k]$: Navigation data message at epoch k

- $\varepsilon_\theta[k]$: Phase tracking error due to thermal noise and the dynamic stress error at epoch k
- $n_i[k]$: Impact of the noise introduced by the AWG channel at epoch k at the correlator output of the I channel
- A: Amplitude of the signal at the receiver antenna output

Note that the $\varepsilon_\theta[k]$ value depends on the chosen assumptions of the signal carrier phase tracking process.

The assumption of a seamless code delay tracking process is very realistic. The reason is that the levels of C/N_0 for which the code delay estimation is accurate enough to remove the influence of the code delay tracking on the received signal at the correlator output are lower than all the levels of C/N_0 analyzed in this chapter and than all the levels of C/N_0 for which the PLL is locked.

5.2.3.1.2. Seamless synchro-frame process hypothesis

The second assumption made by this analysis is that the navigation message synchro-frame process has already been achieved.

The synchro-frame process consists in searching the beginning of a page/word/subframe in the continuous stream of received bits. This means that at any moment of the transmission, the receiver knows exactly which bit inside the frame is being decoded and at which bit the next frame begins. In other words, the receiver is perfectly able to locate all the fields, words, frames, etc, inside the stream of bits.

The main consequence of this assumption is that no bit is lost because the receiver does not know at which field, word, subframe this bit belongs to. Therefore, since no bit is lost, all the pages or subframes are decoded with the proposed decoding techniques; no subframe is lost due to the lack of bits.

Finally, note that this assumption corresponds to the case where the user passes from a zone with satellite LOS signals, which have been acquired, tracked and the synchro-frame process achieved, to a zone with satellite LOS signals of low power. However, in this new zone, the receiver is still able to track the LOS signals of low power and thus there is no need of searching the synchro-frame again. Therefore, this means that the main change from the demodulation point of view is a decreasing of the C/N_0 value.

5.2.3.2. Performance of the Viterbi algorithm with a priori bit probabilities

In this section, the demodulation performance in terms of BER as a function of the signal E_b/N_0 is presented. The demodulation performance is calculated for the GPS L2C signal which carries the navigation message named as GPS L2C mode NAV with FEC data message in this dissertation.

In this case, an ideal carrier phase tracking process has been assumed; therefore the only source of error is the AWG noise.

The a priori probabilities employed by the proposed method only belong to the bits representing the ephemeris data, and have been calculated as explained in section 5.2.1.2. The probabilities associated to the other bits are considered equal to 0.5. Therefore, the following figure presents three curves. The first curve plots the traditional Viterbi decoding method. The second curve plots the BER obtained over the entire navigation message when it is decoded

by the Viterbi algorithm with a priori bit probabilities. And the third curve plots the BER obtained over the ephemeris data of the navigation message when the ephemeris data is decoded by the Viterbi algorithm with a priori bit probabilities.

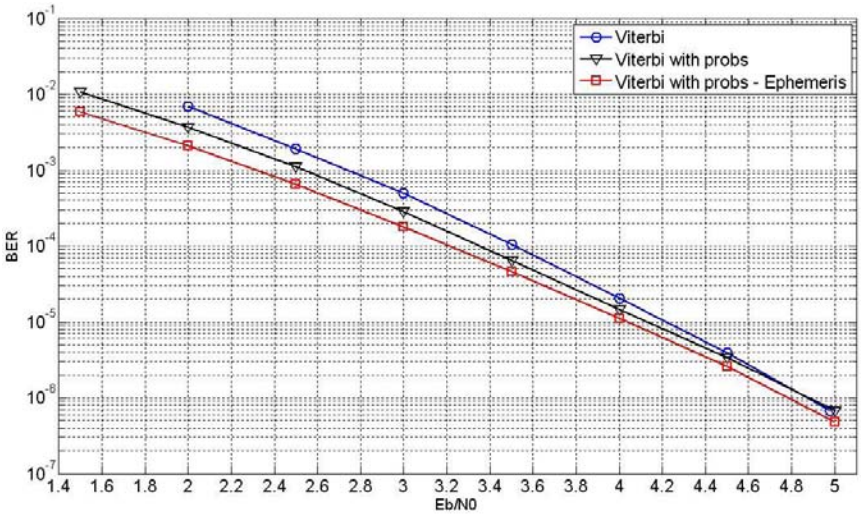

Figure 5-4: BER vs Eb/N0 for GPS L2C mode NAV data with FEC signal when transmitted through an AWGN channel using the Viterbi algorithm with a priori bit probabilities

The first observation that can be made from Figure 5-4 is that the gain in terms of E_b/N_0 introduced by the proposed method decreases when the BER also decreases. For example, the gain is about 0.2 dB for the entire message and about 0.5 dB for the ephemeris when the obtained BER is equal to $4 \cdot 10^{-3}$, but it is only of 0.1 dB and of 0.2 dB when the BER is 10^{-5}.

The second observation is that the gain over the entire navigation message when only the ephemeris probabilities are employed tends to 0 when the BER is very low. And, moreover, the gain over the ephemeris appears to follow the same behavior.

Finally, from Figure 5-4, it can also be seen that the gain is never very large; therefore from this observation and the two previous ones, it can be concluded that this technique does not provide a significant again and thus, it will be more important as a complement of the method based on the inner and outer channel codes combination than as a main decoding technique.

5.2.3.3. Performance of the method based on the inner and outer channel codes combination

The demodulation performance in terms of BER, WER and EER is evaluated for the GPS L2C mode NAV with FEC signal, the GPS L2C mode CNAV signal and GPS L5 signal. Additionally, for the GPS L2C mode NAV with FEC signal, the method based on the inner and outer channel codes combination is complemented by using the a priori probabilities of the bits of the navigation message ephemeris data.

In this case, three different carrier phase tracking processes are assumed. First, the PLL perfectly tracks the signal carrier phase. Second, the carrier phase tracking process is not ideal due to the existence of thermal noise. Third and last, the carrier phase tracking process is not ideal due to the existence of thermal noise and a dynamic stress error. The PLL configuration used to model the carrier phase tracking error has been presented in section 3.2.3.1.

The parameters of the method based on the inner and outer channel codes combination employed during these simulations are presented next. First, there are three options for the number of candidates, 50, 100 and 4000. Second, the initial state is always imposed when the

previous word has been successfully decoded. Third, the number of words forming a candidate depends on the signal: two words form a candidate for GPS L2C mode NAV with FEC signal, and only one word form a candidate for GPS L2C mode CNAV signal and GPS L5 signal. This choice has been made regarding the outer channel code; the code in charge of the error detection. The outer channel code of the GPS L2C mode NAV with FEC signal is the extended Hamming code which is not very powerful. Therefore, the inclusion of a 2^{nd} word increases the decoding performance of the method. The improvement of a third word is less significant and slows the decoding process; therefore, for these simulations the number of words has been set to two. The outer channel code of the GPS L2C mode CNAV signal and GPS L5 signal is the CRC-24Q code which is very powerful. Therefore it is not necessary to extend the number of words forming a candidate.

The calculation of the BER of the proposed decoding method, combination of the inner and of the outer channel codes, and the calculation of the BER of the traditional decoding method, Viterbi algorithm, are described next. First, the BER of the Viterbi algorithm is calculated by dividing the number of bit errors of all the transmitted words between the total number of transmitted bits. This means that the all the bits of the transmitted words, regardless of the outcome of the outer channel code verification, are used on the BER calculation. In other words, the erasures -the bits belonging to a word which has failed the outer channel code verification and thus whose values are uncertain- are used in the BER calculation instead of being set apart. Second, the BER of the proposed decoding method is also calculated by dividing the number of bit errors of all the transmitted words between the total number of transmitted bits. In this case, this means that the bits of all the words, the words passing the outer channel code verification and the words which are known to be wrong because no suitable candidate has been found during the Viterbi search, are used to calculate the BER. Again this means, that the calculation of the BER of the proposed contains erasures, bits whose value is uncertain.

Finally, the configuration of the different speed improvements for each signal was presented in annex E.4.5.4.4.

5.2.3.3.1. Performance when assuming an ideal carrier phase tracking process

For the ideal carrier phase tracking case, only two figures are shown. The first figure displays BER of the GPS L2C mode NAV with FEC signal. The second figure represents the BER of the GPS L2C mode CNAV signal and GPS L5 signal. Note that the second figure can represent both signals at the same time since the carrier phase tracking have been assumed ideal and the BER is expressed as function of the E_b/N_0.

The BER of the GPS L2C mode NAV with FEC signal is presented below for the method based on the inner and outer channel codes combination when using 4000 candidates. The decoding performance obtained by the traditional Viterbi decoding technique is also plotted.

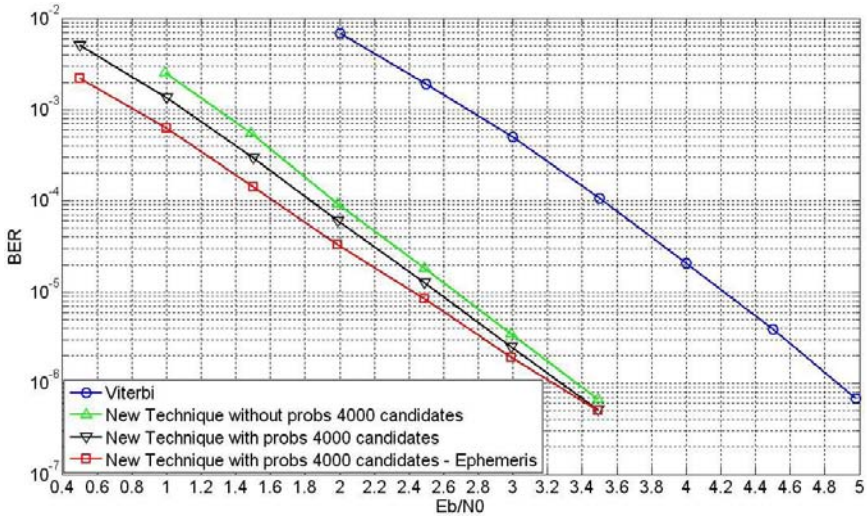

Figure 5-5: BER vs Eb/N0 for GPS L2C mode NAV data with FEC signal when transmitted through an AWGN channel using the method based on the inner and outer channel code combination

The first observation that can be made from Figure 5-5 is that the different options of the method have different gains in terms of E_b/N_0 as it is expected. Nevertheless, the gain difference among them is reduced when the BER decreases. The difference gain rages from 0.4 dB for a BER equal to $2 \cdot 10^{-3}$ to 0 or 0.05 dB for a BER equal to $6 \cdot 10^{-7}$. Note that this convergence could be expected since the only difference between the presented options is the implementation of the a priori bit probabilities of the ephemeris data. And remember that these probabilities induced a convergence between the Viterbi algorithm with a priori bit probabilities and the traditional Viterbi algorithm.

The second observation that can be made from Figure 5-5 is that the plotted gain between the proposed technique with any option and the traditional Viterbi decoding is always larger than 1.4 dB. This is a significant value which increases when the BER decreases. More specifically, for a BER equal to 10^{-6}, a gain of 1.7 dB is achieved over the ephemeris data for 4000 candidates using a priori bit probabilities, a gain of 1.6 dB is achieved over the entire navigation message for 4000 candidates using a priori bit probabilities and a gain of 1.5 dB is achieved over the entire message for 4000 candidates not using a priori bit probabilities.

The results for the GPS L2C mode CNAV signal and GPS L5 signal are presented below for the decoding method based on the inner and outer channel codes combination using 50, 100 and 4000 candidates. The demodulation performance obtained by the traditional Viterbi decoding technique is also plotted.

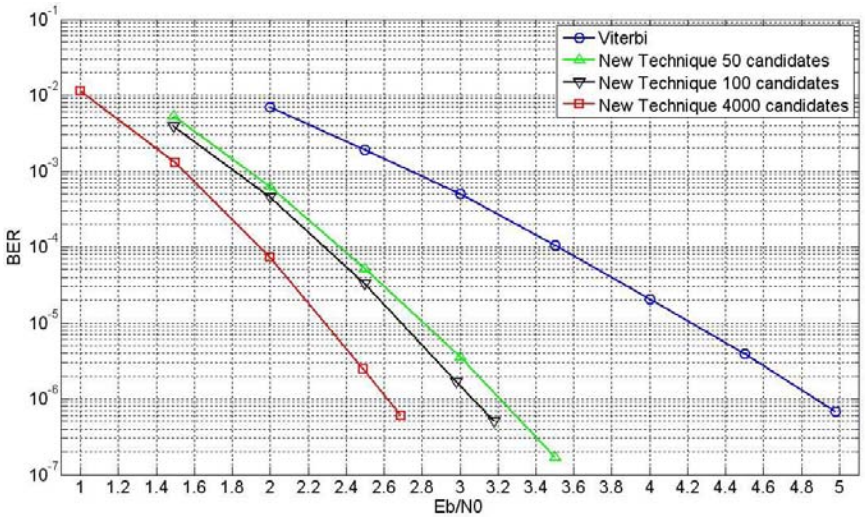

Figure 5-6: BER vs Eb/N0 for GPS L2C mode CNAV data signal and GPS L5 signal when transmitted through an AWGN channel using the method based on the inner and outer channel code combination

The first observation that can be made from Figure 5-6 is that the gain depends on the number of candidates. More specifically, it can be seen that the difference of gain between the different numbers of candidates grows with the decrease of BER. The plotted difference of gain between 50 and 100 candidates ranges from almost 0 dB to 0.1 dB, whereas the plotted difference gain between 100 and 4000 candidates ranges from 0.2 dB to 0.5 dB.

Note that the difference between the GPS L2C mode CNAV data signal and GPS L2C mode NAV data with FEC signal is mainly the outer channel code. In this case, since the CRC-24Q detects any word containing errors with a high probability, the only possibility of not correcting the received sequence is that the proposed algorithm reaches the maximum number of candidates without finding the transmitted sequence. And this means that the larger the number of candidates is, the larger the possibility of correcting the sequence is. Moreover, when the E_b/N_0 increases, it is more probable to find the transmitted sequence among the K inspected candidates, and thus the probability of correcting the received sequence is also increased. And the higher the number of candidates K is, the more this probability is improved.

The second observation that can be made from Figure 5-6 is that the plotted gain between the proposed technique with any number of candidates and the traditional Viterbi decoding is always larger than 0.5 dB. And although this value is smaller than the gain obtained in the previous case, the gain for a significant BER as 10^{-5} and 10^{-6} is equal or larger for the current signals. More specifically, for a BER equal to 10^{-5}, a gain of 1.4 dB is found for 50 candidates, 1.5 dB for 100 candidates and 1.9 dB for 4000 candidates.

5.2.3.3.2. Carrier phase tracking with thermal noise results

For the case assuming a carrier phase tracking process affected only by thermal noise, 11 figures are shown. These figures can be separated into two groups since they show different properties of the new proposed decoding method. The first group is composed of nine figures, three for each signal, and they display the signal BER, WER or EER as a function of the signal C/N_0 at the receiver antenna output. The second group is composed of the remaining two graphics and they show for the GPS L2C mode NAV data with FEC signal the percentage of wrong words detected by the new proposed technique.

Additionally, note that in this case the GPS L2C mode CNAV signal and GPS L5 signal cannot be plotted together since they are transmitted at different symbol information rates and the carrier phase tracking process is not assumed ideal.

5.2.3.3.2.1. GPS L2C mode NAV data with FEC signal

In this subsection, the results of the GPS L2C mode NAV data with FEC signal when the method based on the inner and outer channel codes combination is used and when the carrier phase tracking process is only disturbed by the presence of thermal noise are presented.

The following results are always presented for a proposed method employing 100 and 4000 candidates. Moreover, these results are compared with the demodulation performance of the Viterbi algorithm with a priori bit probabilities when the carrier phase tracking process is disturbed by the presence of thermal noise.

The BER as a function of the signal C/N_0 is presented below.

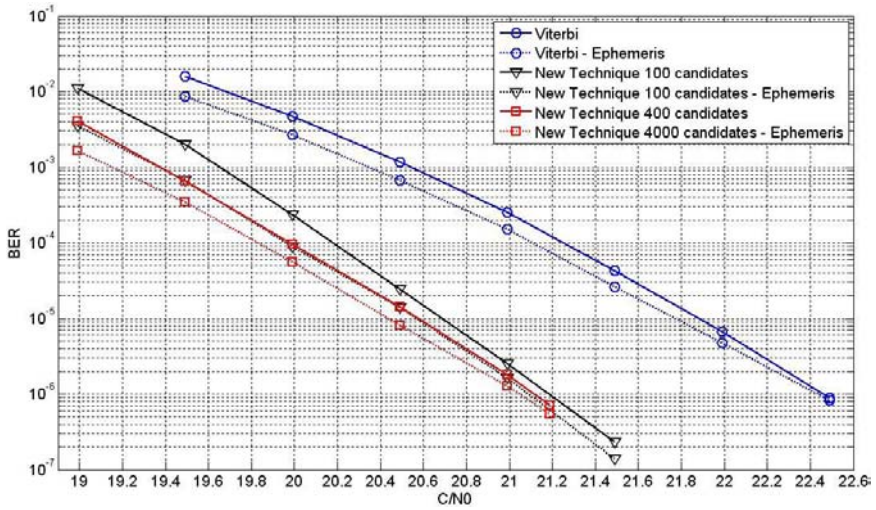

Figure 5-7: BER vs C/N_0 for GPS L2C mode NAV with FEC signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise

From Figure 5-7, all the observations made from Figure 5-5 can be verified. Nevertheless, since the demodulation performance of the Viterbi algorithm with a priori probabilities has been plotted instead of the demodulation performance of the traditional Viterbi, the observed gain is expected to be smaller than the gain obtained in Figure 5-5. In fact, for a BER equal to 10^{-5}, a gain of 1.2 dB is obtained for 100 candidates and a gain of 1.35 dB for 4000 candidates. The same gains are obtained over the ephemeris data.

Moreover, one new important observation that can be made from Figure 5-5 is that the demodulation performance curves of the proposed method with different number of candidates converge. This convergence seems a priori unexpected because the curves obtained for the GPS L5 signal diverge. But the different channel codes implemented by each signal explain this difference. More specifically, it is explained by the Hamming code low error detection capacity compared to the CRC-24Q code. A detailed explanation is given below.

For the GPS L2C mode NAV with FEC data signal, the proposed method has two sources of errors and not only one as it was on the GPS L5 signal case. The first source of errors is the same as the GPS L5 signal case: the method reaches its maximum number of inspected

candidates without finding the transmitted sequence. The second source of errors is introduced by the hamming code limitation: the channel code accepts a wrong word as a correct one. Nevertheless, these two sources of errors have a different behavior when the C/N_0 increases; and it is this different behavior which explains the different evolution of the performance curves. The increase of C/N_0 results into a decrease of the number of candidates inspected before finding the correct sequence. This means that the number of times that the maximum number of candidates is reached without finding the transmitted sequence is highly reduced. However, the Hamming code incapacity of detecting all the wrong words is not altered although the number of words which have to pass the verification is reduced. This means that the number of wrong words passing the outer channel code verification is decreased but not as much as the number of wrong words accepted because the method has reached the maximum number of inspected candidates. Therefore, the second source of errors has more presence in the final number of errors when the C/N_0 increases. And, since the wrong words accepted by the method using 100 candidates are also accepted by the method using 4000 candidates, it seems obvious that gains of the different numbers of candidates converge.

The following figure presents the WER as a function of the signal C/N_0, and for GPS L2C mode NAV data with FEC signal, the words refers to the 5 subframes forming the message.

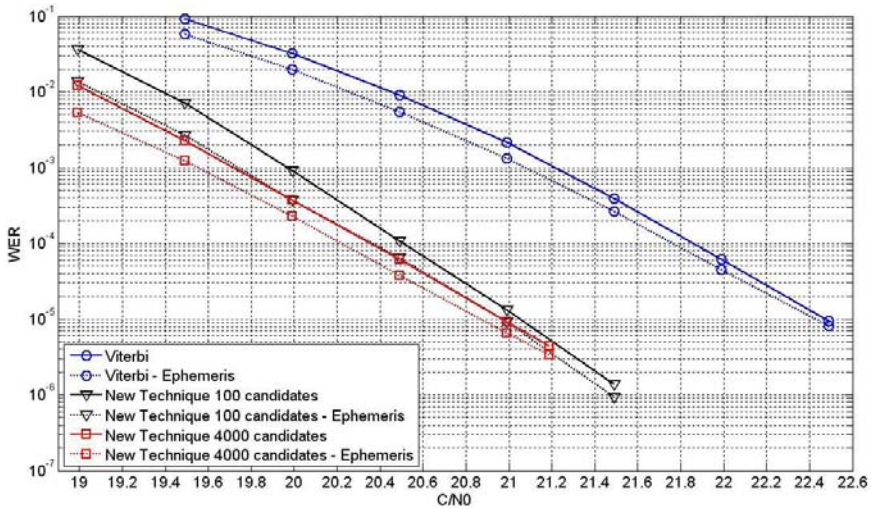

Figure 5-8: WER vs C/N0 for GPS L2C mode NAV with FEC signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise

The more interesting WER values that can be observed from Figure 5-8 are a WER equal to 10^{-3} and a WER equal to 10^{-4}. This figure shows for a WER equal to 10^{-3} a gain of 1.2 dB over the entire message and a gain of 1.55 dB over the ephemeris data when the method uses 100 candidates, and it shows a gain of 1.55 dB over the entire message and a gain of 1.6 dB over the ephemeris data when the method uses 4000 candidates. Moreover, this figure shows for a WER equal to 10^{-4} a gain of 1.35 dB over the entire message and a gain of 1.5 dB over the ephemeris data when the method use 100 candidates, and it shows a gain of 1.5 dB over the entire message and a gain of 1.7 dB over the ephemeris data when the method uses 4000 candidates.

The next figure presents the EER as a function of the signal C/N_0. For GPS L2C mode NAV data with FEC signal, a wrong ephemeris is defined as the wrong decoding of subframe 2 and/or subframe 3 where the two subframes are transmitted inside the same frame.

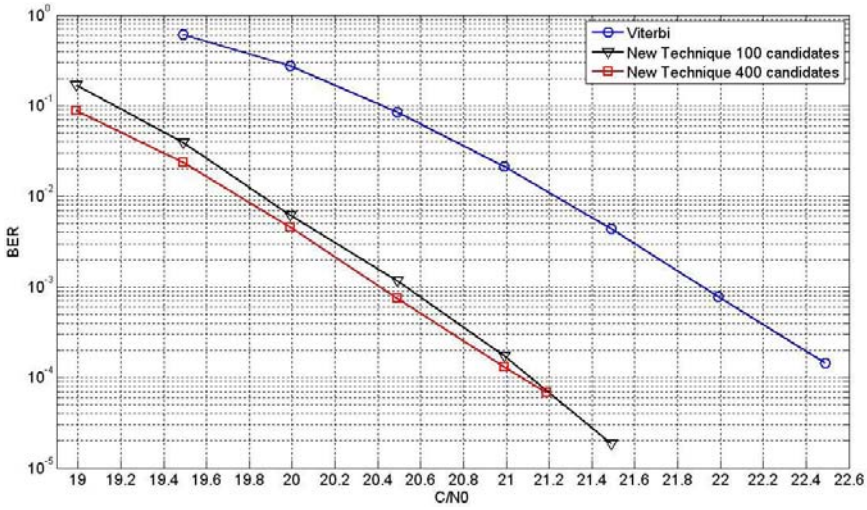

Figure 5-9: EER vs C/N0 for GPS L2C mode NAV with FEC signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise

The more interesting EER values that can be observed from Figure 5-9 are an EER equal to 10^{-2} and an EER equal to 10^{-3}. This figure shows for an EER equal to 10^{-2} a gain of 1.35 dB when the method uses 100 candidates, and it shows a gain of 1.45 dB when the method uses 4000 candidates. Moreover, this figure shows for an EER equal to 10^{-3} a gain of 1.4 dB when the method use 100 candidates, and it shows a gain of 1.5 dB when the method uses 4000 candidates. Therefore, since the ephemeris decoding is the most interesting figure of merit from the final user point of view, it can be concluded that the decoding method based on the inner and the outer channel codes combination when 4000 candidates are used provides a gain of 1.5 dB in terms of C/N_0 compared to the Viterbi algorithm with a priori bit probabilities for a EER equal to 10^{-3}.

The last figure presents the percentage of wrong words and wrong ephemeris detected by the proposed method as a function of the signal C/N_0. Note that the C/N_0 values are equal to C/N_0 values used for the previous three figures in order to allow a rapid association among all the graphics.

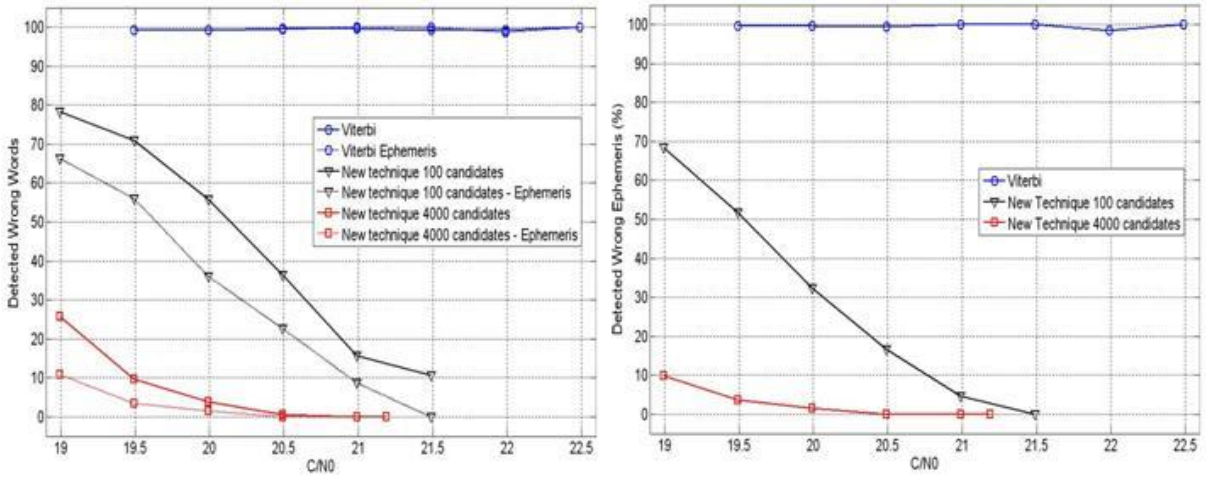

Figure 5-10: Percentage of detected wrong words and wrong epehemeris for carrier phase tracking process noised by thermal noise

From Figure 5-10, it can be observed that the number of detected wrong words and wrong ephemeris decreases with C/N_0. Remember that the total quantity of wrong words is formed

by the wrong words accepted because the maximum number of candidates was reached and by the words wrongly accepted by the extended Hamming channel code. Therefore, this figure verifies the hypothesis that the higher the C/N_0 value is, the more influence over the total number of words has the second source of errors. In fact, the curves tendency shows that for low levels of WER and EER, the second source of errors almost provide the 100% of total number errors.

Moreover, it is also verified that although the BER, WER and EER are improved with the application of the proposed decoding method, the number of wrong words accepted by the proposed method is higher than the number of words accepted by the traditional use of the channel codes.

Another observation that can be made from Figure 5-10 is that, as it was expected, the percentage of detected wrong words and wrong ephemeris is much lower for the proposed method when 4000 candidates are used than for the proposed method when 100 candidates are used. In fact, since the method with 4000 candidates inspects much more candidates, it has a much higher probability of accepting a wrong decoded word.

Nevertheless, note that for the WER values equal to 10^{-3} and to 10^{-4}, the percentages of detected wrong words for the proposed method when 100 candidates are used are about 56% and 35% respectively. And for EER values equal to 10^{-2} and to 10^{-3}, the percentages of detected wrong ephemeris sets for the proposed method when 100 candidates are used are about 35% and 19% respectively. Therefore, this method is still able to detect a reasonable amount of words or ephemeris. Moreover, if the proposed method is used but with 50 candidates this amount should be improve.

Finally, note that all the C/N_0 values used on the previous three figures are always higher than the PLL lock threshold calculated in section 3.1.3.3 Table 3-3. Nevertheless, the lower C/N_0 values are too close to the PLL lock threshold to assure that the PLL is always locked.

5.2.3.3.2.2. GPS L2C mode CNAV data signal

In this subsection, the results of the GPS L2C mode CNAV data signal when the method based on the inner and outer channel codes combination is used and when the carrier phase tracking process is only disturbed by the presence of thermal noise are presented.

The following results are always presented for a proposed method employing 100 and 4000 candidates. Moreover, these results are compared to the demodulation performance of the traditional Viterbi algorithm when the carrier phase tracking process is disturbed by the presence of thermal noise.

The BER as a function of the signal C/N_0 is presented below.

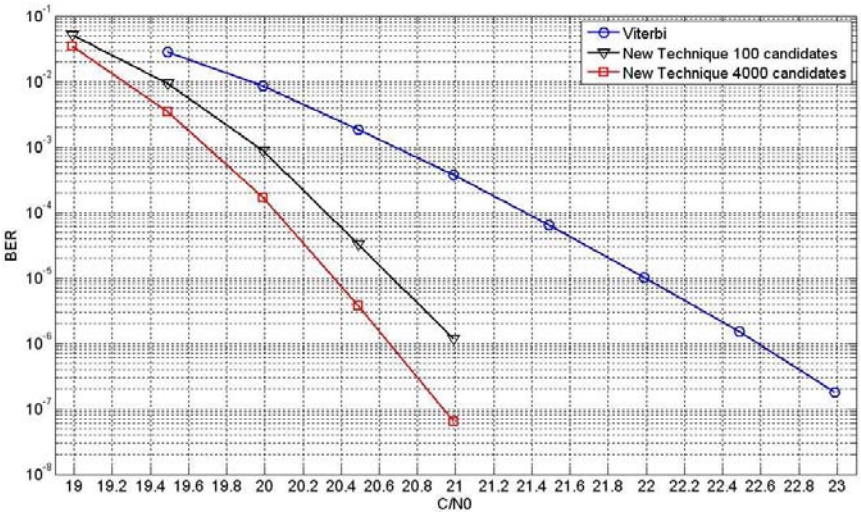

Figure 5-11: BER vs C/N0 for GPS L2C mode CNAV data signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise

From Figure 5-11, all the observations and conclusions extracted from Figure 5-6 are verified. In Figure 5-11, it can be seen that for a BER equal to 10^{-5}, the proposed method obtain a gain of 1.35 dB when 100 candidates are used and a gain of 1.65 dB when 4000 candidates are used. Moreover, for a BER equal to 10^{-6} the gains are of 1.6 dB when 100 candidates are used and of 1.95 dB when 4000 candidates are used.

The following figure presents the WER as a function of the signal C/N_0, and for GPS L2C mode CNAV data signal, the words refers to the any subframes inside the stream of bits.

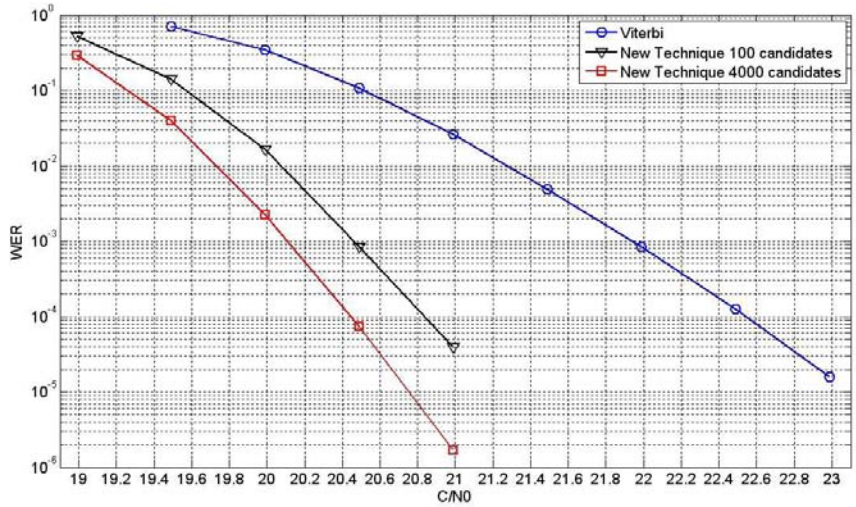

Figure 5-12: WER vs C/N0 for GPS L2C mode CNAV data signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise

The more interesting WER values that can be observed from Figure 5-12 are a WER equal to 10^{-3} and a WER equal to 10^{-4}. This figure shows for a WER equal to 10^{-3} a gain of 1.5 dB when the method uses 100 candidates, and it shows a gain of 1.85 dB when the method uses 4000 candidates. Moreover, this figure shows for a WER equal to 10^{-4} a gain of 1.7 dB when the method use 100 candidates, and it shows a gain of 2.1 dB when the method uses 4000 candidates.

The last figure presents the EER as a function of the signal C/N_0. Nevertheless, for GPS L2C mode CNAV data signal, the defined frame has not a rigid structure [ARINC, 2004];

therefore, the calculation made on this research work assumes that among 5 transmitted subframes, 2 of them carry the ephemeris. In other words, the same calculation used for GPS L2C mode NAV data with FEC signal has been implemented.

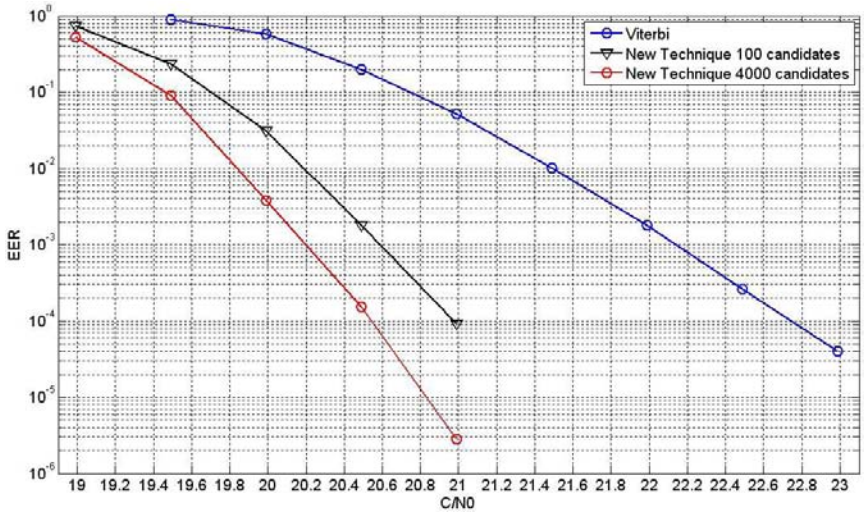

Figure 5-13: EER vs C/N0 for GPS L2C mode CNAV data signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise

The more interesting EER values that can be observed from Figure 5-13 are an EER equal to 10^{-2} and an EER equal to 10^{-3}. This figure shows for an EER equal to 10^{-2} a gain of 1.3 dB when the method uses 100 candidates, and it shows a gain of 1.65 dB when the method uses 4000 candidates. Moreover, this figure shows for an EER equal to 10^{-3} a gain of 1.55 dB when the method use 100 candidates, and it shows a gain of 1.95 dB when the method uses 4000 candidates.

Finally, note that all the C/N_0 values used on the previous three figures are always higher than the PLL lock threshold calculated in section 3.1.3.3 Table 3-3. Nevertheless, the lower C/N_0 values are too close to the PLL lock threshold to guarantee that the PLL is always locked.

5.2.3.3.2.3. GPS L5 signal

In this subsection, the results of the GPS L5 data signal when the method based on the inner and outer channel codes combination is used and when the carrier phase tracking process is only disturbed by the presence of thermal noise are presented.

The following results are always presented for a proposed method employing 100 and 4000 candidates. Moreover, these results are compared to the demodulation performance of the traditional Viterbi algorithm when the carrier phase tracking process is disturbed by the presence of thermal noise.

The BER as a function of the signal C/N_0 is presented below.

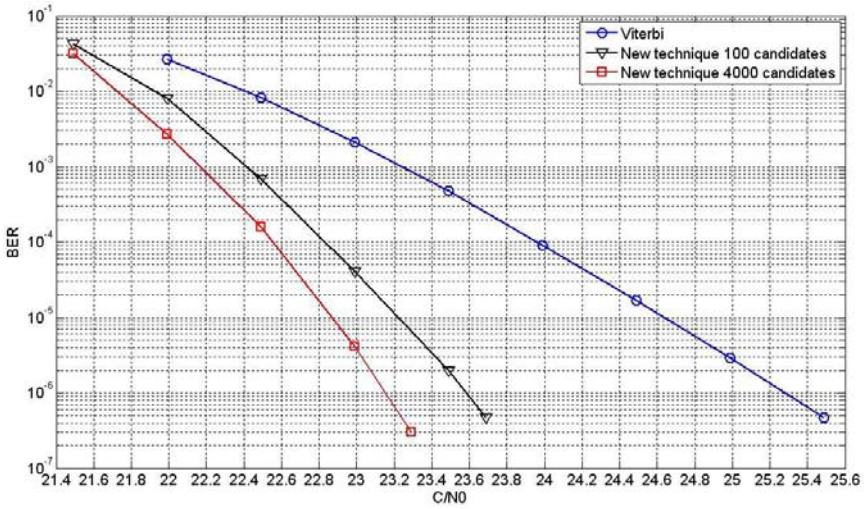

Figure 5-14: BER vs C/N0 for GPS L5 signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise

In Figure 5-14, it can be seen that for a BER equal to 10^{-5}, the proposed method obtains a gain of 1.4 dB when 100 candidates are used and a gain of 1.75 dB when 4000 candidates are used. Moreover, for a BER equal to 10^{-6} the gains are of 1.7 dB when 100 candidates are used and of 2.15 dB when 4000 candidates are used.

The following figure presents the WER as a function of the signal C/N_0, and for GPS L5 signal, the words refers to the any subframes inside the stream of bits.

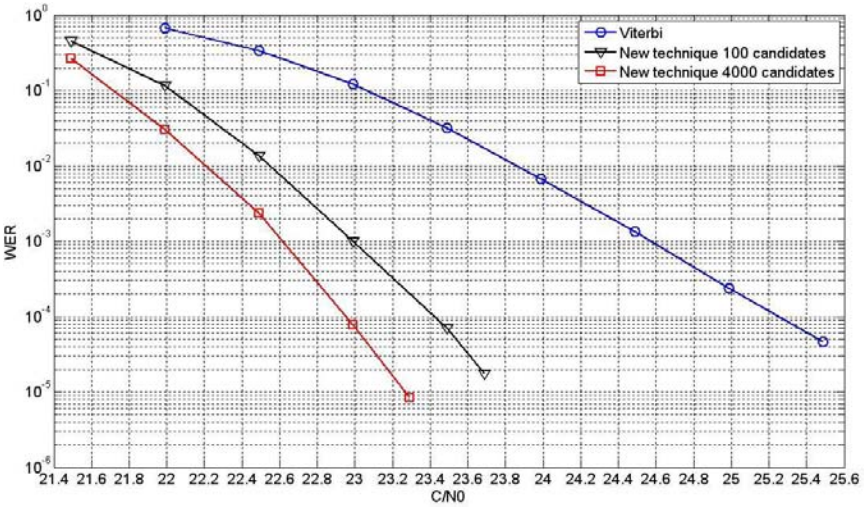

Figure 5-15: WER vs C/N0 for GPS L5 signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise

The more interesting WER values that can be observed from Figure 5-15 are a WER equal to 10^{-3} and a WER equal to 10^{-4}. This figure shows for a WER equal to 10^{-3} a gain of 1.55 dB when the method uses 100 candidates, and it shows a gain of 1.95 dB when the method uses 4000 candidates. Moreover, this figure shows for a WER equal to 10^{-4} a gain of 1.85 dB when the method use 100 candidates, and it shows a gain of 2.3 dB when the method uses 4000 candidates.

The last figure presents the EER as a function of the signal C/N_0. And the EER calculation for the GPS L5 signal has been done as the calculation of GPS L2C mode CNAV data signal since they have the same message structure.

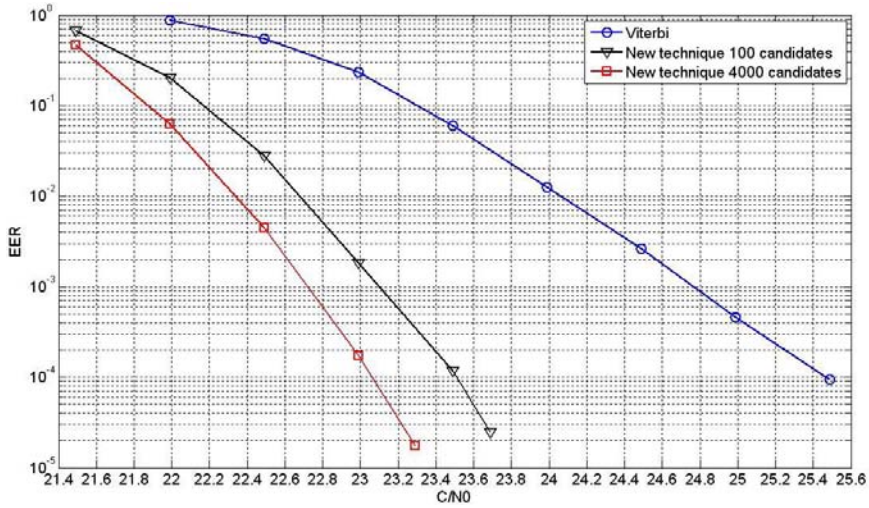

Figure 5-16: EER vs C/N0 for GPS L5 signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise

The more interesting EER values that can be observed from Figure 5-16 are an EER equal to 10^{-2} and an EER equal to 10^{-3}. This figure shows for an EER equal to 10^{-2} a gain of 1.4 dB when the method uses 100 candidates, and it shows a gain of 1.7 dB when the method uses 4000 candidates. Moreover, this figure shows for an EER equal to 10^{-3} a gain of 1.65 dB when the method use 100 candidates, and it shows a gain of 2.05 dB when the method uses 4000 candidates.

One important observation made from Figure 5-11 to figure 5-16 is that although both signals GPS L5 and GPS L2C mode CNAV data have the same navigation message structure, the difference in symbol transmission rate makes the proposed method slightly more effective when applied on the GPS L5 signal.

Finally, note that all the C/N_0 values used on the previous three figures are always higher than the PLL lock threshold calculated in section 3.1.3.3 Table 3-3.

5.2.3.3.3. Carrier phase tracking with thermal noise and dynamic stress error result

For the case assuming a carrier phase tracking process affected by thermal noise and by dynamic stress error, the same figures as for the case assuming a carrier phase tracking process affected only by thermal noise are presented. Nine figures display the BER, the WER or the EER of each signal as a function of its C/N_0 at the receiver antenna output. The two remaining figures display the percentage of wrong words detected by the proposed method when applied over the GPS L2C mode NAV with FEC data signal.

5.2.3.3.3.1. GPS L2C mode NAV data with FEC signal

In this subsection, the results of the GPS L2C mode NAV data with FEC signal when the method based on the inner and outer channel codes combination is used and when the carrier phase tracking process is disturbed by the presence of thermal noise and a dynamic stress error of 1 jerk are presented.

The following results are always presented for a proposed method employing 100 and 4000 candidates. Moreover, these results are compared with the demodulation performance of the

Viterbi algorithm with a priori bit probabilities when the carrier phase tracking process is disturbed by the presence of thermal noise.

The BER as a function of the signal C/N_0 is presented below.

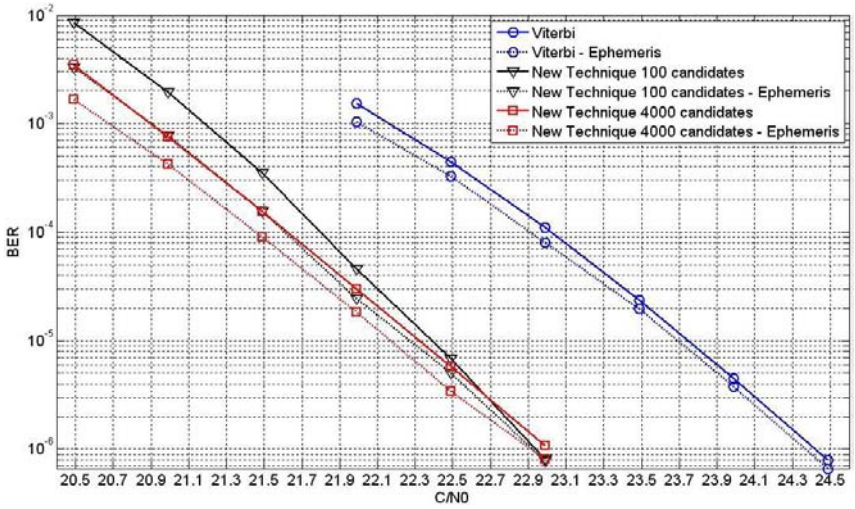

Figure 5-17: BER vs C/N0 for GPS L2C mode NAV with FEC signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise and 1 jerk of dynamic stress

The first observation that can be made from figure Figure 5-17 is that the curve of the proposed method when 100 candidates are used and the curve of the proposed method when 4000 candidates are used converge and even cross. That results into a better demodulation performance of the 100 candidates option for large C/N_0 values. More specifically, it can be seen that the proposed method when 100 candidates are used obtain a better demodulation performance than the proposed method when 4000 candidates are used for C/N_0 values larger than 25.7 or 26 dB-Hz. The element responsible of this behavior is again the limitation of the outer channel code: a wrong word if accepted as a correct one (second source of errors).

As has been said before and has been verified by Figure 5-10, for high C/N_0 values the second source of errors has more influence over the final number of errors. Therefore, the different behavior of this source for the proposed method when 4000 candidates are used and the proposed method when 100 candidates are used has to be analyzed in order to understand the demodulation performance curve evolution.

Assuming that the transmitted sequence is the X candidate where X is larger than 100 but lower than 4000, the proposed method when 100 candidates are used cannot find the transmitted sequence whereas the proposed method when 4000 candidates are used can. However, it is possible that before reaching the X candidate, the Hamming code allows a wrong word to be accepted as the transmitted sequence. Therefore, if this occurs, two different situations can happen. First, the accepted wrong word is the Y candidate where $Y < 100$. In this case, both method options decode the same wrong word and thus the number of errors obtained by each method option is the same. Second, the accepted wrong word is the Y candidate where $Y > 100$. In this case, the proposed method when 100 candidates are used provides the 1st candidate as decoded word, whereas the proposed method when 4000 candidates are used provides the Y candidate. Therefore, the number of errors provided by each method option varies. And, since the 1st candidate average number of errors should be lower than the average number of errors of any wrong accepted sequence, the proposed method when 100 candidates are used should slightly outperform the proposed method when

4000 candidates are used. Obviously, there will be cases where the 4000 candidates option will find the transmitted word whereas the 100 candidates option will be unable. Nevertheless, this last case appears to be much less frequent than the previous commented situation for large C/N_0 values. Moreover, this fact can be worsened by the discard of the transmitted sequence before the beginning of the candidate search due to the implementation of speed improvements.

To finish with Figure 5-17, it can be observed that for a BER equal to 10^{-5}, the proposed method obtain a gain of 1.35 dB over the entire message when 100 candidates are used and a gain of 1.45 dB when 4000 candidates are used. And for a BER equal to 10-^{6}, the proposed method obtain a gain of 1.5 dB over the entire message when 100 candidates are used and a gain of 1.45 dB over the entire message when 4000 candidates are used. Moreover, about the same gains are obtained over the ephemeris.

The following figure presents the WER as a function of the signal C/N_0, and the words are defined as the message subframes 1 to 5.

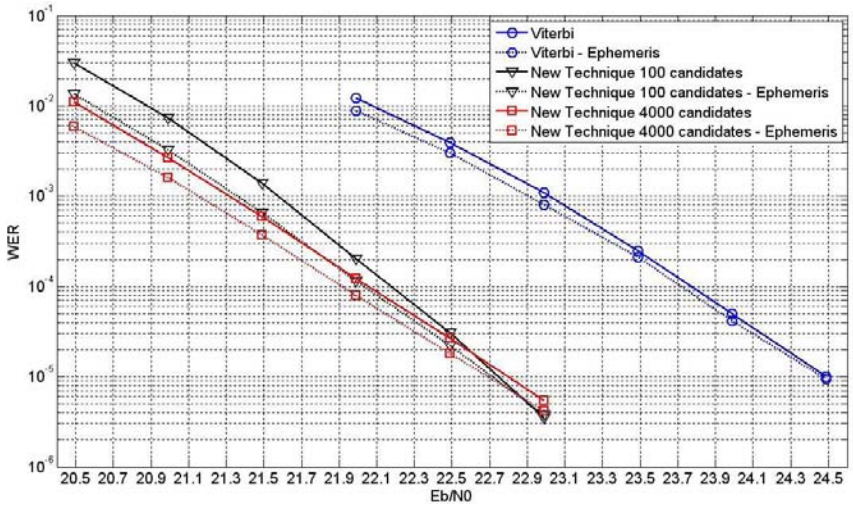

Figure 5-18: WER vs C/N0 for GPS L2C mode NAV with FEC signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise and 1 jerk of dynamic stress

The more interesting WER values that can be observed from Figure 5-18 are a WER equal to 10^{-3} and a WER equal to 10^{-4}. This figure shows for a WER equal to 10^{-3} a gain of 1.45 dB over the entire message when the method uses 100 candidates, and it shows a gain of 1.7 dB over the entire message when the method uses 4000 candidates. Moreover, this figure shows for a WER equal to 10^{-4} a gain of 1.6 dB over the entire message when the method use 100 candidates, and it shows a gain of 1.85 dB over the entire message when the method uses 4000 candidates.

The next figure presents the EER as a function of the signal C/N_0.

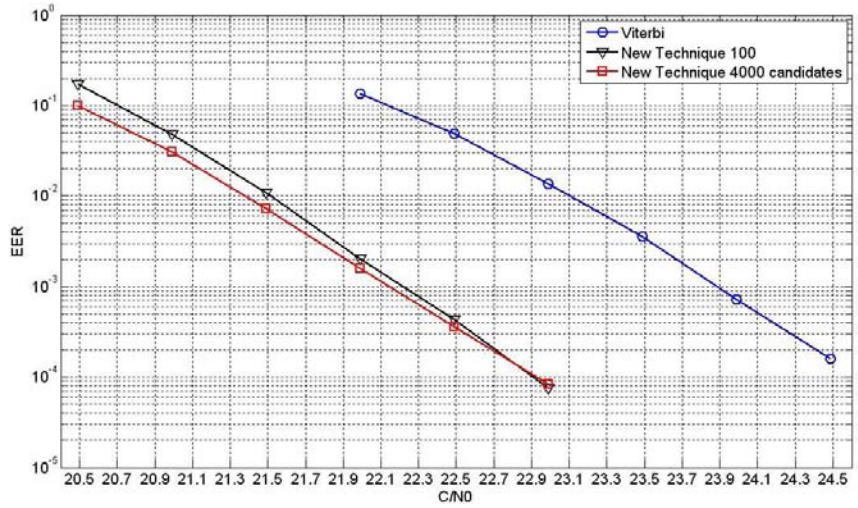

Figure 5-19: EER vs C/N0 for GPS L2C mode NAV with FEC signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise and 1 jerk of dynamic stress

The more interesting EER values that can be observed from Figure 5-19 are an EER equal to 10^{-2} and an EER equal to 10^{-3}. This figure shows for an EER equal to 10^{-2} a gain of 1.6 dB when the method uses 100 candidates, and it shows a gain of 1.75 dB when the method uses 4000 candidates. Moreover, this figure shows for an EER equal to 10^{-3} a gain of 1.65 dB when the method use 100 candidates, and it shows a gain of 1.75 dB when the method uses 4000 candidates.

One general conclusion that can be extracted from comparing these last three figures with figures from Figure 5-7 to Figure 5-9 is the following. The proposed method for GPS L2C mode NAV data with FEC signal provides slight better performance than the Viterbi algorithm with a priori bit probabilities when the carrier phase tracking process is affected by thermal noise and a dynamic stress error of 1 jerk than when the carrier phase tracking process is only affected by thermal noise.

The last figure presents the percentage of wrong words and wrong ephemeris detected by the proposed method as a function of the signal C/N_0. Note that the C/N_0 values are equal to the C/N_0 values used for the previous three graphics in order to allow a rapid association among all the graphics.

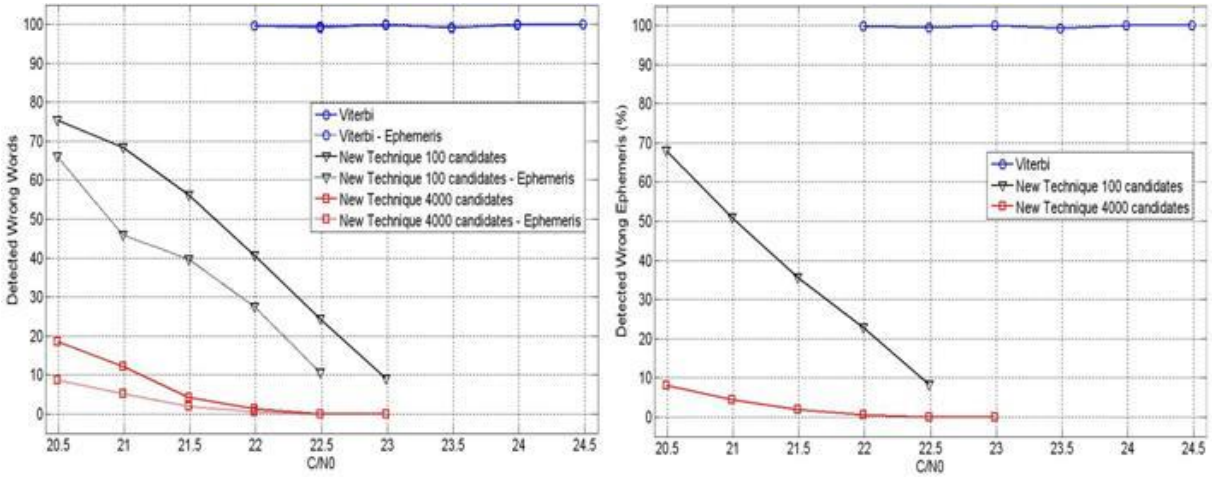

Figure 5-20: Percentage of detected wrong words and ephemeris for carrier phase tracking process noised by thermal noise and 1 jerk of dynamic stress error

The same conclusions of the case where a carrier phase tracking process affected only by thermal noise can be extracted from Figure 5-20 (see Figure 5-10).

Moreover, about the same percentage of detected wrong words or wrong ephemeris is obtained for this case as for the case where the carrier phase tracking process is only corrupted by thermal noise.

Finally, note that all the C/N_0 values used on the previous three figures are almost always lower than the PLL lock threshold calculated in section 3.1.3.3 Table 3-3. Therefore, the method based on the inner and outer channel codes combination could only be used for the higher C/N_0 values of the GPS L2C mode NAV with FEC signal. This means that whenever the PLL is locked, a receiver implementing the proposed technique will provide excellent demodulation performance.

5.2.3.3.3.2. GPS L2C mode CNAV data signal

In this subsection, the results of the GPS L2C mode CNAV data signal when the method based on the inner and outer channel codes combination is used and when the carrier phase tracking process is disturbed by the presence of thermal noise and a dynamic stress error of 1 jerk are presented.

The following results are always presented for a proposed method employing 100 and 4000 candidates. Moreover, these results are compared with the demodulation performance of the traditional Viterbi algorithm when the carrier phase tracking process is disturbed by the presence of thermal noise.

The BER as a function of the signal C/N_0 is presented below.

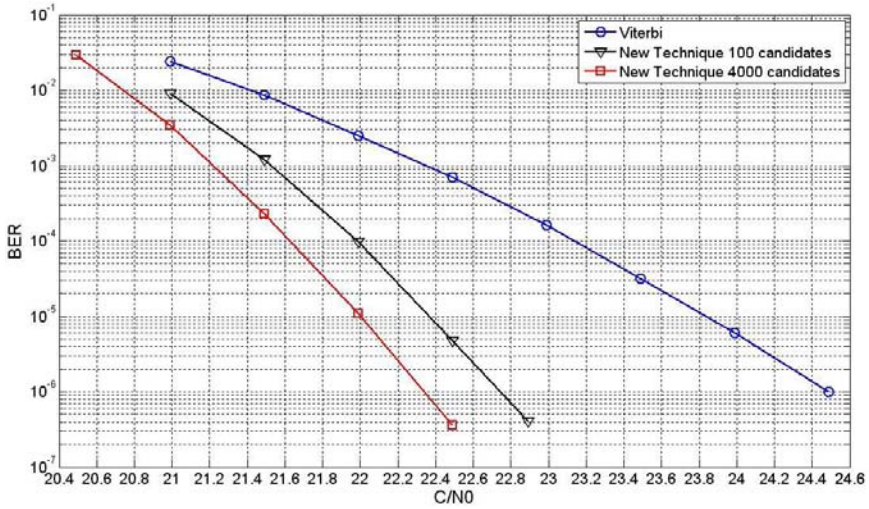

Figure 5-21: BER vs C/N0 for GPS L2C mode CNAV data signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise and 1 jerk of dynamic stress

In Figure 5-21, it can be seen that for a BER equal to 10^{-5}, the proposed method obtains a gain of 1.45 dB when 100 candidates are used and a gain of 1.8 dB when 4000 candidates are used. Moreover, for a BER equal to 10^{-6} the proposed method obtains a gain of 1.7 dB when 100 candidates are used and a gain of 2.1 dB when 4000 candidates are used.

The following figure presents the WER as a function of the signal C/N_0, and for GPS L2C mode CNAV data signal, the words refers to the any subframes inside the stream of bits.

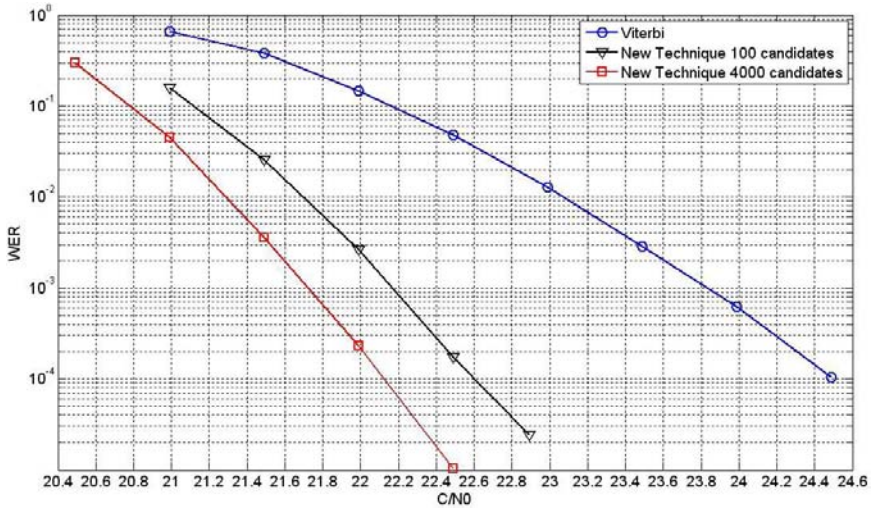

Figure 5-22: WER vs C/N0 for GPS L2C mode CNAV data signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise and 1 jerk of dynamic stress

The more interesting WER values that can be observed from Figure 5-22 are a WER equal to 10^{-3} and a WER equal to 10^{-4}. This figure shows for a WER equal to 10^{-3} a gain of 1.65 dB when the method uses 100 candidates, and it shows a gain of 2.1 dB when the method uses 4000 candidates. Moreover, this figure shows for a WER equal to 10^{-4} a gain of 1.9 dB when the method use 100 candidates, and it shows a gain of 2.4 dB when the method uses 4000 candidates.

The last figure presents the EER as a function of the signal C/N_0. Nevertheless, for GPS L2C mode CNAV data signal, the defined frame has not a rigid structure [ARINC, 2004]; therefore, the calculation made in this research work assumes that among 5 transmitted subframes, 2 of them carry the ephemeris. In other words, the same calculation used for GPS L2C mode NAV data with FEC signal has been implemented.

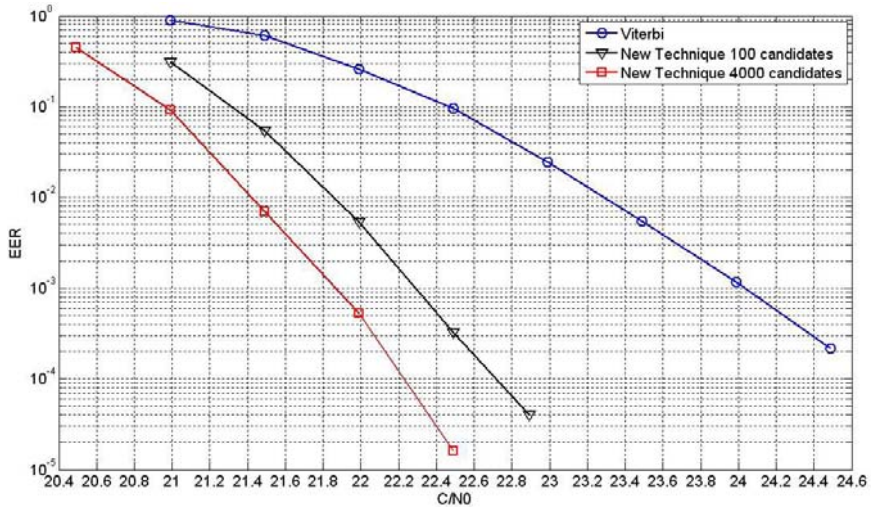

Figure 5-23: EER vs C/N0 for GPS L2C mode CNAV data signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise and 1 jerk of dynamic stress

The more interesting EER values that can be observed from Figure 5-23 are an EER equal to 10^{-2} and an EER equal to 10^{-3}. This figure shows for an EER equal to 10^{-2} a gain of 1.45 dB when the method uses 100 candidates, and it shows a gain of 1.9 dB when the method uses 4000 candidates. Moreover, this figure shows for an EER equal to 10^{-3} a gain of 1.7 dB when

the method use 100 candidates, and it shows a gain of 2.15 dB when the method uses 4000 candidates.

The same observation of the GPS L2C mode NAV data with FEC signal can be made for the current signal when comparing its demodulation performance between the case where the carrier phase tracking process is corrupted by thermal noise and a dynamic stress error of 1 jerk and the case where carrier phase tracking process is only corrupted by thermal noise: the proposed method obtains a large gain in comparison with the traditional Viterbi decoding when more sources of errors affects the carrier phase tracking process.

Finally, note that all the C/N_0 values used on the previous three figures are almost always lower than the PLL lock threshold calculated in section 3.1.3.3 Table 3-3. Therefore, the method based on the inner and outer channel codes combination could only be used for the higher C/N_0 values of the GPS L2C mode CNAV data signal. This means that whenever the PLL is locked, a receiver implementing the proposed technique will provide excellent demodulation performance.

5.2.3.3.3.3. GPS L5 signal

In this subsection, the results of the GPS L5 data signal when the method based on the inner and outer channel codes combination is used and when the carrier phase tracking process is disturbed by the presence of thermal noise and a dynamic stress error of 1 jerk are presented.

The following results are always presented for a proposed method employing 100 and 4000 candidates. Moreover, these results are compared with the demodulation performance of the traditional Viterbi algorithm when the carrier phase tracking process is disturbed by the presence of thermal noise.

The BER as a function of the signal C/N_0 is presented below.

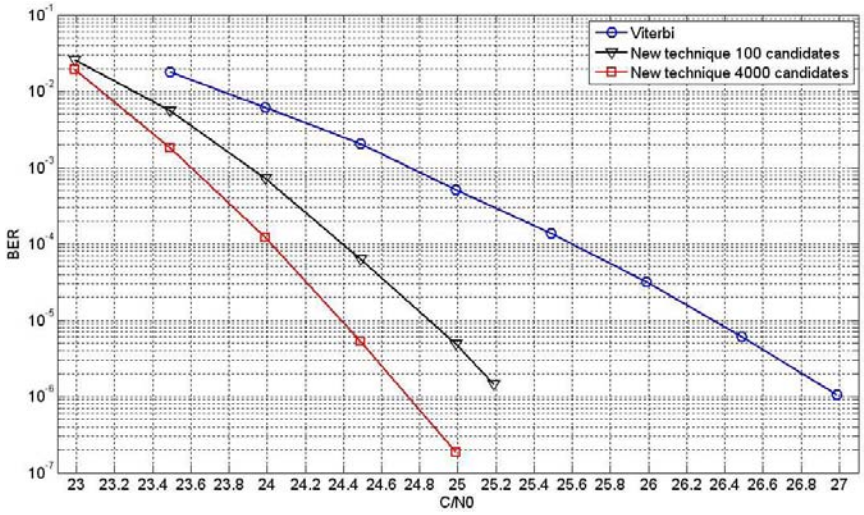

Figure 5-24: BER vs C/N0 for GPS L5 signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise and 1 jerk of dynamic stress

In Figure 5-24, it can be seen that for a BER equal to 10^{-5}, the proposed method obtains a gain of 1.5 dB when 100 candidates are used and a gain of 1.95 dB when 4000 candidates are used. Moreover, for a BER equal to 10^{-6} the proposed method obtains a gain of 1.75 dB when 100 candidates are used and a gain of 2.25 dB when 4000 candidates are used.

The following figure presents the WER as a function of the signal C/N_0, and for GPS L5 signal, the words refers to the any subframes inside the stream of bits.

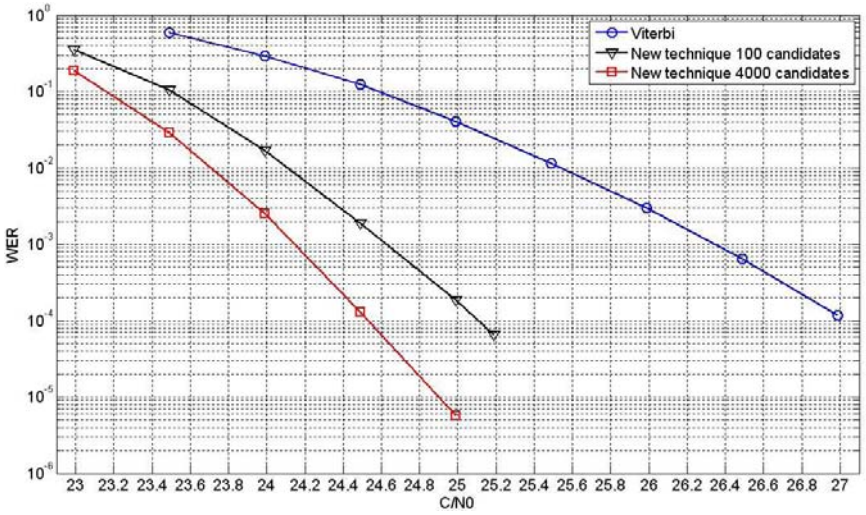

Figure 5-25: WER vs C/N0 for GPS L5 signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise and 1 jerk of dynamic stress

The more interesting WER values that can be observed from Figure 5-25 are a WER equal to 10^{-3} and a WER equal to 10^{-4}. This figure shows for a WER equal to 10^{-3} a gain of 1.75 dB when the method uses 100 candidates, and it shows a gain of 2.2 dB when the method uses 4000 candidates. Moreover, this figure shows for a WER equal to 10^{-4} a gain of 1.9 dB when the method use 100 candidates, and it shows a gain of 2.45 dB when the method uses 4000 candidates.

The last figure presents the EER as a function of the signal C/N_0. And the EER calculation for the GPS L5 signal has been done as the calculation of GPS L2C mode CNAV data signal since they have the same message structure.

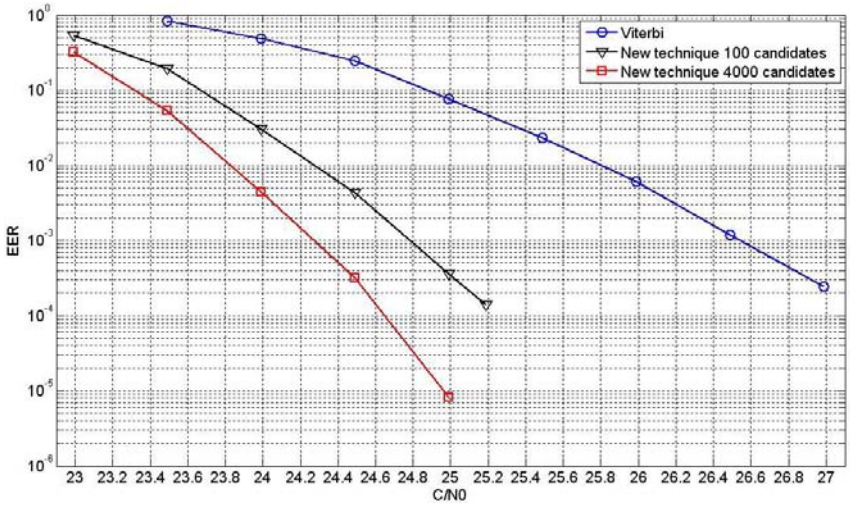

Figure 5-26: EER vs C/N0 for GPS L5 signal when transmitted through an AWGN channel with a carrier phase tracking process disturbed by thermal noise and 1 jerk of dynamic stress

The more interesting EER values that can be observed from Figure 5-26 are an EER equal to 10^{-2} and an EER equal to 10^{-3}. This figure shows for an EER equal to 10^{-2} a gain of 1.55 dB when the method uses 100 candidates, and it shows a gain of 2 dB when the method uses 4000 candidates. Moreover, this figure shows for an EER equal to 10^{-3} a gain of 1.75 dB

when the method use 100 candidates, and it shows a gain of 2.3 dB when the method uses 4000 candidates.

One observation that can be made from these GPS L5 signal figures when compared to the last GPS L2C mode CNAV figures is that, as well as in the case where the carrier phase tracking process is only affected by thermal noise, the gain between the proposed method and the traditional Viterbi decoding is a bit larger for the GPS L5 signal. Therefore, it can be concluded that the higher signal symbol transmission rate of the GPS L5 influences positively the gain of performance between the decoding methods. The reason is given next.

The gain of decoding performance between the method based on the inner and outer channel codes combination and the traditional Viterbi algorithm when these decoding methods are applied on the GPS L5 signal can be discussed from the gain of the GPS L2C mode CNAV data signal. More specifically, we analyze the GPS L5 signal case from the GPS L2C mode CNAV data signal case by interpreting the GPS L5 signal case as an extension of the GPS L2C mode CNAV signal case where the signal symbol transmission rate is increased. This analysis is presented below.

The available E_b/N_0 for a given value of C/N_0 depends on the symbol transmission rate; the higher the symbol rate is, the lower the E_b/N_0 value is. Therefore, when a signal symbol transmission rate is increased, the E_b/N_0 is decreased resulting into a rise of BER, WER and EER. However, the carrier phase tracking performance remains constant since the C/N_0 has not changed. Therefore, when the symbol transmission rate is increased for the GPS L5 signal with respect to the GPS L2C mode CNAV signal, the signal C/N_0 has to be also increased so that the original E_b/N_0 value remains constant and the levels of BER, WER or EER are conserved. Note that the initial C/N_0 of the GPS L2C mode CNAV signal is different for the proposed method than for the traditional Viterbi algorithm. More specifically, this necessary increase of E_b/N_0 is completely determined by the new higher symbol transmission rate when the carrier phase tracking process is ideal; however if the process is not ideal, it also depends on the carrier phase tracking performance. In fact, when the carrier phase tracking process is not ideal, each time that the C/N_0 is increased, the tracking performance is also improved. This means that in addition to increase the E_b/N_0 directly from the increase of the C/N_0 value, the E_b/N_0 is also increased through the improvement of the carrier phase tracking process. Therefore, since the curve representing the carrier phase tracking performance is steeper for lower C/N_0 values and the initial C/N_0 values required for the proposed method are lower than the traditional Viterbi algorithm values, the increase of E_b/N_0 is larger for the proposed method. This means that for the same increase of C/N_0, the proposed method obtains a larger increase of E_b/N_0 than the traditional Viterbi algorithm. That results into a larger gain between methods which means that the gain between the proposed method and the traditional Viterbi algorithm should be larger for the GPS L5 signal than for the GPS L2C mode CNAV signal.

Moreover, it can also be observed that the gain between the method based on the inner and outer channel codes combination and the traditional Viterbi algorithm is larger when the carrier phase tracking process is corrupted by thermal noise and a dynamic stress error of 1 jerk than when it is only noised by thermal noise. This phenomenon can be explained by the same justification used on the previous paragraph: the increase of lower C/N_0 values farther increases the signal E_b/N_0 than the increase of higher C/N_0 values since the carrier phase tracking performance is farther improved for lower C/N_0 values. Nevertheless, there is a significant difference with the previous situation which has to be explained. In this case, the signal transmission rate is not modified since we are inspecting the demodulation performance of the same signal but with different carrier phase tracking process assumptions. In fact, in this case, the demodulation performance is modified because the carrier phase

tracking performance is worsened, whereas in the previous situation the carrier phase tracking performance did not change. This means that whereas in the previous case the two curves representing the demodulation performance of the two decoding methods were modified equally, in this case they are changed differently. The reason is given next. A constant dynamic stress error adds a constant carrier phase offset over the estimated carrier phase; and this offset has a larger negative effect over the worse estimated carrier phases. Therefore, since the C/N_0 values of the Viterbi algorithm are larger than the C/N_0 values of the proposed method, the initial estimated carrier phase error is worse for the proposed technique. This means, that the dynamic stress error affects more negatively the initial proposed method performance. Nevertheless, from the results obtained, it appears that in this case this effect is less important than the better improvement of low C/N_0 values, and thus the difference of gain between the proposed method and the traditional Viterbi is increased. However, a more detailed analysis should be done whenever a dynamic stress error appears in order to quantify the different effects and in order to determine the outcome.

Finally, note that all the C/N_0 values used on the previous three figures are always higher than the PLL lock threshold calculated in section 3.1.3.3 Table 3-3.

5.2.3.3.4. Summarizing tables

In this section, two tables summarizing the more significant results obtained by the proposed technique over the different signals are presented. The tables show the C/N_0 values required to obtain the desired BER, WER or EER values and the gain of the proposed method with respect to the traditional Viterbi algorithm. Remember that we used the Viterbi algorithm with a priori bit probabilities instead of the traditional Viterbi algorithm for the GPS L2C mode NAV with FEC data signal.

The first table shows the values of the method based on the inner and outer channel codes combination when 100 candidates are used.

	Signal	C/N_0 (dB-Hz)						Gain difference (dB)					
		BER		WER		EER		BER		WER		EER	
		10^{-5}	10^{-6}	10^{-3}	10^{-4}	10^{-2}	10^{-3}	10^{-5}	10^{-6}	10^{-3}	10^{-4}	10^{-2}	10^{-3}
Th. Noise	L2C NAV	23.7	24.2	23	23.5	22.85	23.5	1.2	1.3	1.2	1.35	1.35	1.4
	L2C CNAV	20.65	21	20.45	20.85	20.2	20.6	1.35	1.6	1.5	1.7	1.3	1.55
	L5	23.2	23.6	23	23.4	22.65	23.1	1.4	1.7	1.55	1.85	1.4	1.65
Th. N. + Dyn.	L2C NAV	25.4	25.9	24.55	25.2	24.5	25.25	1.35	1.5	1.45	1.6	1.6	1.65
	L2C CNAV	22.35	22.75	22.15	22.6	21.85	22.3	1.45	1.7	1.65	1.9	1.45	1.7
	L5	24.85	25.25	24.65	25.1	24.25	24.8	1.5	1.75	1.75	1.9	1.55	1.75

Table 5-1: Demodulation performance summary of the inner and outer channel codes combination method using 100 candidates

The second table shows the values of the method based on the inner and outer channel codes combination when 4000 candidates are used.

	Signal	Signal C/N_0 (dB-Hz)						Gain difference (dB)					
		BER		WER		EER		BER		WER		EER	
		10^{-5}	10^{-6}	10^{-3}	10^{-4}	10^{-2}	10^{-3}	10^{-5}	10^{-6}	10^{-3}	10^{-4}	10^{-2}	10^{-3}
Th. Noise	L2C NAV	23.55	24.1	22.75	23.35	22.75	24.5	1.35	1.4	1.55	1.5	1.45	1.5
	L2C CNAV	20.35	20.65	20.1	20.45	19.85	20.2	1.65	1.95	1.85	2.1	1.65	1.95
	L5	22.85	23.15	22.6	22.95	22.35	22.7	1.75	2.15	1.95	2.3	1.7	2.05
Th. N. + Dyn.	L2C NAV	25.3	26	24.3	24.95	24.35	25.15	1.45	1.45	1.7	1.85	1.75	1.75
	L2C CNAV	22	22.35	21.7	22.1	21.4	21.85	1.8	2.1	2.1	2.4	1.9	2.15
	L5	24.4	24.75	24.15	24.55	23.8	24.25	1.95	2.25	2.2	2.45	2	2.3

Table 5-2: Demodulation performance summary of the inner and outer channel codes combination method using 4000 candidates

Remember that the calculation of the BER of the proposed decoding method, combination of the inner and outer channel code, and the calculation of the BER of the traditional decoding method, Viterbi algorithm plus verification of the outer channel code, takes into account the erasures as commented more in detail in section 5.2.3.3.

5.3. Conclusions

In this chapter, two alternative decoding methods to the traditional Viterbi decoding algorithm have been presented in order to improve the demodulation performance of the GPS L2C mode NAV with FEC data signal, the GPS L2C mode CNAV data signal and the GPS L5 signal. These methods are a Viterbi algorithm which has been modified in order to use the a priori probabilities of the received bits and a new algorithm which combines the inner and outer channel codes of the received signal.

The Viterbi algorithm with a priori bit probabilities has shown to provide a slight improvement, about 0.2 dB, with respect to the traditional Viterbi algorithm. Moreover, this gain decreases with higher C/N_0 values until the two algorithms obtain the same performance. Therefore this technique is considered as a complement of the current traditional Viterbi algorithm rather than a real substitute. Moreover, the Viterbi algorithm with a priori probabilities can also be used as modification of the second proposed decoding method.

The method based on the inner and outer channel codes combination has shown to provide an improvement larger than 1.2 dB with respect to the traditional Viterbi algorithm for any signal at any relevant BER, WER and EER values. More specifically, the larger gains are obtained for the GPS L5 signal, for the WER figure of merit and for the case where the carrier phase tracking process is corrupted by thermal noise and by a dynamic stress error of 1 jerk. Table 5-1 and Table 5-2 summarize all the results obtained by the method based on the inner and outer channel codes combination.

Moreover, the larger gains of the method based on the inner and outer channel codes combination with respect to the traditional Viterbi algorithm are obtained when this method uses the larger number of candidates. Therefore, the first impression is that the method based on the inner and outer channel codes combination should always be employed with the

maximum number of candidates. Nevertheless, there are some gain exceptions and some other considerations that have to be taken into account.

For GPS L2C mode NAV data with FEC signal, the gain obtained by the method based on the inner and outer channel codes combination when 100 are used converge with the gain when 4000 are used for high C/N_0 values. Even the gain appears to be better when 100 candidates are used. The reason of this convergence is the limitation of the outer channel code as explained during the presentation of the results: the probability of detection of the extended Hamming code is not high enough. Moreover, this limitation also causes that the method based on the inner and outer channel codes combination when 4000 candidates are used is able to detect much less wrong words than this method when 100 candidates are used. Therefore, since the gain is quite similar for both method options the suitable choice for the GPS L2C mode NAV data with FEC signal is the proposed method when 100 candidates are used. The reasons are the following. First, the percentage of detected wrong words is higher for the method when 100 candidates are used. Second, the execution speed of the method and the necessary receiver resources such as the power and the memory are lower when the method uses 100 candidates.

Additionally, it would be interesting to inspect the decoding performance of the method based on the inner and outer channel codes combination when 50 candidates are used since the required resources will decrease, the percentage of detected wrong words will increase and the gain should be only slightly lower than the gain when 100 candidates are used as shown in Figure 5-6.

Moreover, since even the method based on the inner and outer channel codes combination when 50 candidates are used allows the acceptation of wrong words, this method cannot be the suitable option to be implemented on applications which cannot accept any wrong words for GPS L2C mode NAV data with FEC signal.

For GPS L2C mode CNAV data signal and GPS L5 data signal, the gain is always larger for the method based on the inner and outer channel codes combination when 4000 candidates are used. This gain grows with the increase of the C/N_0 with respect to the traditional Viterbi algorithm and with respect to the gain of this same method but when 100 candidates are used. Moreover, due to the high probability of detection of the outer channel code of these signals, none wrong word has been accepted as correct in the simulations implementing this code for any number of candidates. Therefore, the only advantage of the method based on the inner and outer channel codes combination when 100 candidates are used in comparison with this method when 4000 candidates are used is the algorithm execution speed and the required resources of the receiver. This means that the choice of the number of candidates for these signals will depend on the application/user priority or limitations: the gain or the receiver resources.

Finally, when the carrier phase tracking process is corrupted by thermal noise and a dynamic stress error of 1 jerk, the method based on the inner and outer channel codes combination can provide interesting BER, WER and EER values for C/N_0 values which are lower than the PLL lock threshold for GPS L2C mode NAV with FEC data signal and the GPS L2C mode CNAV data signal. Nevertheless, this does not mean that this method cannot be applied for these signals. In fact, it means that whenever the C/N_0 is about the PLL locking threshold, the BER, WER and EER which can be obtained by the application of the proposed method will be very satisfying. The reason is that the proposed method evaluated at the lowest C/N_0 value at which the PLL should be locked provides a BER about 10^{-5} for both signals.

Chapter 6. Demodulation performance of Galileo E1 OS and GPS L1C navigation messages in different types of environments

GPS L1C and GALILEO E1 OS are two of the current navigation open service signals being developed nowadays. GPS L1C is the US open signal and is being conceived in order to provide enhanced capabilities to civilian users compared to previous GPS signals and in order to achieve interoperability with GALILEO Open Service signal [ARINC, 2006]. GALILEO is the European system and is being developed in order to obtain independence from any other navigation system, while searching to reach maximum interoperability with other GNSS systems, particularly with GPS. GALILEO, being a navigation system of a new generation, also searches to provide a very high positioning performance by means of its open service signal E1 OS [ESA, 2008].

One of the aspects improved by both systems is the navigation data message demodulation process. Indeed, the current GPS L1 C/A and GPS L2C signal do not provide satisfying demodulation performance in low C/N_0 environments such as indoor and urban environments. These signals were not originally conceived to work in mobile environments and thus one of the objectives of the GPS L1C signal and GALILEO E1 OS signal is to provide (the GPS L1C signal and GALILEO E1 OS signal have to be adapted to provide) positioning service in these new kinds of environments.

In fact, the necessity of adapting both signals to mobile and low C/N_0 environments is caused by the growth of popularity of the positioning services. This growth has changed the environments of application of the most demanded positioning services, from the traditional open environments to the mobile and low C/N_0 environments. More specifically, the current majority of applications are vehicular or pedestrian applications in cities and even indoor buildings. And all these new scenarios have several characteristics in common and are referred and modeled as mobile and/or low C/N_0 environments.

However, in spite of searching mostly the same objectives, each signal has been conceived in a different form and with a different signal structure. The PRN codes, the PRN codes period, the signal modulation, etc, are different for each signal. Moreover, some of these design differences such as the relative power distribution between the data and pilot channel, the symbol transmission rate, channel code, interleaver, etc, affect directly the signal demodulation performance. Therefore, since these signals will be the most popular due to their free accessibility and a priori good performance, it is interesting to study the demodulation performance of each signal and to compare them in order to detect the pros and cons of each design.

One big design difference which directly affects the demodulation performance of each signal is the larger range of services offered by GALILEO E1 signal. More specifically, whereas GPS L1C signal only provides free positioning service in its current public version document [ARINC, 2006], GALILEO signals in E1 frequency band have been designed to offer SoL (Safety of Life) service, integrity service and PRS service in addition to the free positioning service. Therefore, due to this high level design choice, GALILEO E1 navigation data message has to carry additional information in order to support these two extra services. This means that GALILEO E1 signal will carry (has to be adapted to) a larger stream of

information than GPS L1C; and normally, a higher information rate results into a lower demodulation performance.

In this chapter, we first present the main difference of the signal design between GALILEO E1 signal and GPS L1C signal affecting the demodulation performance. Second, each individual effect of these differences is detailed. Third, a theoretical study of the demodulation performance which can be expected from each signal is carried out. Fourth, the signal demodulation performance for an AWGN channel and for a mobile channel is calculated through simulations. Finally, the conclusions are presented.

6.1. Main differences in GPS L1C and GALILEO E1 OS signals

In this section, the main differences of design between the GPS L1C and GALILEO E1 OS signals which condition the demodulation performance are presented.

The main differences are the signal channel relative power distribution, the symbol transmission rate, the structure of the data message, the implemented interleaver and the data message content. Nevertheless, all the signal characteristics have already been explained in 0, during the definition of each signal. Therefore, in this section only a brief remainder is presented in order to quickly compare the signals.

6.1.1. Signal Channel Relative Power Distribution

Each signal, GALILEO E1 OS or GPS L1C, have a different distribution of the total amount of power at the receiver antenna output between the two transmitted channels, the data channel and the pilot channel.

In fact, since the pilot PRN code and the data PRN code are uncorrelated, the total amount of power is simply the addition between the pilot channel power and the data channel power.

$$Total\ Signal\ Power = Data\ Channel\ Power + Pilot\ Signal\ Power \tag{6-1}$$

The signal channel relative power distribution of the two signals can be summarized in the table shown below.

Signal	*Data Channel*	*Pilot Channel*
GPS L1C	25%	75%
GALILEO E1 OS	50%	50%

Table 6-1: Signal Channel Relative Power Distribution between the data and pilot channels of the GPS L1C and GALILEO E1 OS signals

6.1.2. Symbol transmission rate

The symbol transmission rate is the number of physical symbols which are transmitted by a signal each second. The notation used for the symbol transmission rate is R_D and it is 2.5 times higher for GALILEO E1 OS signal than for GPS L1C signal. Table 6-2 summarizes the values.

Signal	GPS L1C	GALILEO E1 OS
Symbol Transmission Rate (symb/sec)	100	250

Table 6-2: Symbol Transmission Rate for GPS L1C and GALILEO E1 OS

6.1.3. Structure of the data message structure

The structure of the data message defines the frame, the subframes, the words, etc, of a signal and specifies the channel code implemented for each word or subframe of the signal.

The structure of the data message of GPS L1C signal is completely different from the structure of the data message of GALILEO E1 OS signal. In fact, both the size and the frequency of the information units -word, subframe, frame- and the implemented channel code are different for each navigation signal. However, remember that the demodulation performance here is the ephemeris data demodulation performance; therefore the data message structure which conditions the ephemeris demodulation performance and thus which is analyzed in this chapter is the navigation message parts carrying the ephemeris data sets. These parts are not described in this chapter since a detailed description of each navigation signal data message structure is given in 0.

Nevertheless, it is important to note a great difference of design between the two signals which conditions all the following data message structure aspects. GALILEO E1 OS signal has a part of a page of 250 bits of size as its smallest information unit whereas the GPS L1C units of information or subframes have a variable size. Therefore, whereas GALILEO E1 OS signal requires 8 parts of a page (4 pages) to broadcast the satellite ephemeris data, GPS L1C only requires 1 subframe.

6.1.4. Interleaver

The interleaver is the component used to break the temporal correlation between consecutive bits of a transmitted message.

For both signals, the same type of interleaver is implemented but the interleaver parameters are different. The type of interleaver defines how the message is reordered before it is transmitted, and, in this case, the implemented type of interleaver for both signals consists in storing the message into a matrix by writing the information in rows/columns and sending the information to the channel by reading the matrix in columns/rows. The inverse action is made in reception to recover the natural bit message order. Therefore, in this case, the matrix rows and columns are the interleaver parameters. These parameters for both signals are defined below.

On one hand, the GPS L1C interleaver is applied over the message subframes 2 and 3 at the same time and the matrix dimensions are 38 rows and 46 columns. A more detailed description can be found in annex D.4.1. On the other hand, the GALILEO E1 OS interleaver is applied over only one part of a page and the matrix dimensions are 30 columns and 8 rows. A more detailed description can be found in annex D.5.1.2.

6.1.5. Data message Content

As said in subsection 6.1.3, the demodulation performance analyzed in this section is the ephemeris data demodulation performance of each signal. Therefore, the exact differences between the GPS L1C signal and GALILEO E1 OS signal ephemeris content are presented in this subsection.

For GPS L1C signal, the ephemeris and clock data correction are transmitted inside the message subframe 2. Moreover, the same ephemeris set and clock correction data set are

transmitted for at least 15 minutes, or, in other words, the ephemeris information is constant for at least 15 minutes. Therefore, since the subframe 2 does not contain any other type of information, the exact same subframe 2 is received by the user for at least 15 minutes.

For GALILEO E1 OS signal, the ephemeris and data clock correction are transmitted inside the pages from 1 to 4. However, in this case, in addition to the ephemerides, there is more information transmitted inside these pages. This additional information carries the SoL service and the integrity service, and since there is not a specific description about this information at the current public document [ESA, 2008], this information is considered variable. Therefore, only the ephemeris part of the pages from 1 to 4 is considered constant whereas the other part is considered variable.

Moreover, the current public document [ESA, 2008] does not guarantee any period of time inside which the transmitted ephemeris data set is constant. Besides, the alarm type page does not have a defined place inside the subframe layout; therefore, as it names indicates, we consider that this page can appear at any time.

6.2. Influence of the main signal differences over the demodulation performance

In this section, the influences of all the signal design differences over the demodulation performance are presented. Nevertheless, before describing the influence of each individual difference, guidelines about the analysis of a signal demodulation performance are presented.

The decoding performance of a channel code is calculated by obtaining the channel code BER as a function of the useful message $E_b/N0$, where the E_b is the energy per information bit. This calculation provides a curve which only contains the improvement inherent to the analyzed channel code, since no other factor takes part into the calculation. Therefore, this means that the demodulation performance of any signal implementing a channel code, included GNSS signals, can be divided into two factors. And thus, all the influences can be divided into two groups where the first group gathers the influences taking part into factor 1 and the second group gathers the influences taking part into factor 2. The two factors are detailed next.

The first factor consists in finding from the available C/N_0 at the receiver antenna output the useful E_c/N_0 which will be used to decode the information message, where the E_c is the energy per coded bit. The second factor consists in determining the BER values from the E_c/N_0 value provided by factor 1. This means that first the E_c/N_0 has to be transformed to E_b/N_0, and second the BER has to be obtained by evaluating the channel code decoding performance curve at the E_b/N_0 value. The E_c/N_0 to E_b/N_0 transformation relates the energy per coded bit with the energy per information bit; therefore its mathematical expression is simply:

$$\frac{E_b}{N_0} \cdot r = \frac{E_c}{N_0} \tag{6-2}$$

Where:

- r: channel code rate

Finally, in this section it is assumed an ideal code delay tracking process. Therefore, the power losses due to an inaccurate code delay estimation are not taken into account.

6.2.1. Influence of the signal channel relative power distribution

The signal channel relative power distribution belongs to the group of factor 1 since it determines the total amount of power carried by the signal which is directly used to transport the data message.

In fact, through the signal channel relative power distribution, the designers can establish the relationship between the signal C/N_0 at the receiver antenna output and the data channel C/N_0. Therefore, since the navigation signals normally carry the data message in only one channel, the C/N_0 available to demodulate the data and the C/N_0 available to track the signal have to be determined previously.

$$\begin{cases} \dfrac{C}{N_{0\ data}} = \dfrac{1}{Distribution} \cdot \dfrac{C}{N_{0\ total}} \\ \dfrac{C}{N_{0\ pilot}} = \left(1 - \dfrac{1}{Distribution}\right) \cdot \dfrac{C}{N_{0\ total}} \end{cases} \tag{6-3}$$

Where:

- *Distribution:* Inverse of the percentage of power associated to the data channel
- $(C/N_0)_{total}$: signal C/N_0 at the RF/IF output block
- $(C/N_0)_{data}$: data channel C/N_0 at the RF/IF output block
- $(C/N_0)_{pilot}$: pilot channel C/N_0 at the RF/IF output block

Once the C/N_0 available for the demodulation process and the tracking process have been determined, the transformation from the signal C/N_0 to the message E_c/N_0 is made as usual but taking into account the special structure of the navigation signal.

For a telecommunication signal, the available C/N_0 at the receiver antenna output cannot be directly transformed to the message E_c/N_0 because part of the signal power can be lost due to an inaccurate tracking process. Therefore, the C/N_0 at the receiver antenna output has to be transformed into the useful C/N_0 value employed to demodulate the message before obtaining the E_c/N_0. This transformation depends on the tracking performance and whereas in the telecommunications signal case, the tracking performance depends again on the C/N_0 at the receiver antenna output, in the navigation signal case, the tracking performance depends on the pilot C/N_0. And this means that for navigation signals, the useful C/N_0 employed to demodulate the message depends on the signal channel relative power distribution from two different ways. First, the distribution determines in a direct way the C/N_0 destined to demodulate the message before taking into account tracking considerations. Second, the distribution determines in an indirect way, the useful C/N_0 employed to decode the message from the data C/N_0 since the tracking performance is determined by the pilot C/N_0; and the pilot C/N_0 value is defined from the signal C/N_0 at the receiver antenna output by the signal channel relative power distribution.

The mathematical expression of the useful data $(C/N_0)_{use\_data}$ is shown below.

$$\frac{C}{N_{0\ use\_data}} = \frac{1}{Distribution} \cdot \frac{C}{N_{0\ total}} \cdot f\left(\tau, \hat{\tau}, \theta, \hat{\theta}\right) \tag{6-4}$$

$$\hat{\tau} = g\left(\frac{C}{N_{0\ pilot}}\right) \quad \hat{\theta} = h\left(\frac{C}{N_{0\ pilot}}\right) \tag{6-5}$$

Where:

- τ: Incoming signal code delay
- θ: Incoming signal carrier phase
- $\hat{\tau}$: Code delay estimation
- $\hat{\theta}$: Estimated carrier phase
- $f\left(\hat{\tau},\hat{\theta}\right)$: Function modeling the carrier phase and code delay tracking performance
- g($C/N_{0\ pilot}$): Function providing the code delay estimation from the C/N_0 available to the pilot channel.
- h($C/N_{0\ pilot}$): Function providing the carrier phase estimation from the C/N_0 available to the pilot channel.

Finally, note that, although providing more relative power to the data channel seems the right option to increase the demodulation performance, providing a significant power to the pilot channel is also required so that not too much useful power is lost due to a poor code delay or carrier phase estimation.

6.2.2. Influence of the symbol transmission rate

The symbol transmission rate belongs to the first factor group since it relates the C/N_0 of the useful signal component employed to demodulate the message to the useful E_c/N_0 employed to decode the message.

In fact, in order to express the figure of merit in energy (E_s) instead of in power (C), a conversion using a factor of time is applied. In this case, this time factor is the symbol duration (T_D) which is the inverse of the symbol transmission rate. Therefore, if the useful signal power is known, the symbol duration can be used to determine how much energy is converged by each symbol. The mathematical expression relying C/N_0 and E_s/N_0 is giving below. This relation was presented and commented in 0.

$$\frac{C}{N_{0\ use\_data}} \cdot T_D = \frac{E_s}{N_0} \tag{6-6}$$

Where:

- $C/N_{0\ use\_data}$: useful C/N_0 of the data channel used to demodulate the signal
- T_D: Symbol Duration
- E_s/N_0: Energy per symbol to power density noise ratio

The influence of the symbol transmission rate can be thus seen directly in equation (6-6). The lower the transmission rate is, the higher the symbol duration is and thus the more energy a bit contains. And the more energy a bit contains, the less the bit detection is affected by the noise.

Another way of analyzing the influence of the symbol transmission rate is to analyze the nature of the sampled value, d_m, which is used to determine the symbol value. This sampled

value is obtained by accumulating the input signal for a symbol period. Therefore, the longer the symbol period is, the better average of the noise is obtained, which for a centered white noise is 0.

Moreover, note that for a telecommunication signal or for a navigation signal not having a pilot channel, the PLL and DLL integration time is limited to the symbol duration. This means that the symbol transmission rate also plays a role on the C/N_0 of the useful signal component since it conditions the tracking performance. However, the C/N_0 used for the demodulation of GPS L1C signal, GALILEO E1 OS signal and any other navigation signal implementing a pilot channel is not restricted by the symbol duration rate.

Finally, the E_c/N_0 is obtained from the E_s/N_0 by dividing E_s/N_0 by the number of bits (k) represented by one symbol:

$$\frac{E_c}{N_0} = \frac{E_s}{N_0} \cdot \frac{1}{k} \tag{6-7}$$

Where:

- E_c/N_0: Energy per coded bit to power density noise ratio
- k: Number of coded bits mapped by a symbol

6.2.3. Influence of the structure of the data message

The data message structure belongs to the group of the second factor since it defines the relationship between the E_c/N_0 value and the obtained BER and EER levels. Therefore, the structure of the data message influence is divided into two parts. The first part is the influence on the BER and the second part is the influence on the EER.

The structure of the data message determines the message BER first by determining the size of the information words carrying the ephemeris data and clock correction data; and second by defining the channel code implemented over the previous determined information words. More specifically, the number of bits which can be corrected by a channel code depends on the type of channel code and on its lengths. And the increase of the size of an information word leads to the implementation of a more powerful channel code over the word as it was demonstrated by Shannon [PROAKISd, 2001]. Additionally, note that the relationship between E_c and E_b depends on the channel code rate, which is defined by the channel code.

The structure of the data message also determines the message EER through the number of code words carrying the ephemeris information. The number of code words does not affect the BER and WER calculation. In fact, the calculation of the EER is made from the WER value, and that means that the structure of the data message determines the relationship between the WER and the EER. The reason is that if all the code words carrying the different ephemeris data parts have not been well decoded, the ephemeris information cannot be recovered. And this means that whereas only 1 well decoded word is required in the case where 1 word carries all the ephemeris information, Y well decoded words are required in the case where Y words carry the ephemeris information. Therefore, assuming that all the words have the same probability of error, the probability of correctly decoding the ephemeris information decreases when the number of words carrying the ephemeris increases. Moreover, note that normally, the word probability of error when only one word carries the ephemeris data will be lower with respect to the probabilities of error of the words carrying a part of the

total ephemeris data information since the word size will be bigger and thus the implemented channel code will be more powerful.

Finally, although the statements of the two previous paragraphs are completely true for an AWGN channel where the errors are independent along the time, these statements have to be modified for mobile channels. A mobile channel introduces a variation of the received signal C/N_0 along the time, whereas for an AWGN channel the C/N_0 is considered constant. This means that it is natural for a mobile channel to have large series of consecutives bits with such low C/N_0 that the channel code is not able to correct them. Or, even the bits cannot be demodulated because the PLL is not locked. Therefore, in these cases, it is possible that a large packet always contains too many errors to be corrected and thus, even with several consecutive receptions of the same word, its correct reception is almost impossible. But, on the contrary, if many small packets are sent instead of a large packet, it is possible that after several consecutive receptions of the packets, the receiver is able to recover the whole ephemeris set by correctly receiving some packets from each repetition until completing the entire information. Therefore, it can be seen that the advantage of having long packets is not so obvious for mobile channels.

6.2.4. Influence of the interleaver

The interleaver belongs to the group of the second factor since it defines the relationship between the E_b/N_0 and the obtained BER level.

At the beginning of section 6.2, we said that the decoding performance of a channel code is modeled through a curve which relates the useful message E_b/N_0 with the BER obtained by the code. Nevertheless, this curve is usually generated assuming a transmission through an AWGN channel, where the noise added to one symbol is completely independent from the noise added to any other symbol of the signal. Therefore, it is not obvious whether this previous calculated curve can be applied when the transmission is made through a mobile channel. The answer is no and the reason is given next.

The number of errors which can be corrected by a channel code depends on two factors. The first factor is the instantaneous amplitude of the noise with respect to the symbol amplitude; because it is not the same to transmit a BPSK symbol with value equal to 1 and to receive a noised value equal to -0.1 than to receive a noised value equal to -2. The second factor is the pattern of errors found in the word because, for convolutional codes and the LDPC codes, it is much easier to correct n bit errors when they are uniformly distributed along a word than when they are concentrated in only one part of the word. Therefore, this means that the channel code decoding performance curve has been calculated for its best scenario. And, since the scenario of a mobile channel is far from the ideal scenario because large series of consecutives errors can be found on the message after its transmission through a mobile channel, a new calculation of the channel code BER as a function of the useful message E_b/N_0 has to be made.

However, instead of calculating a new relationship between the channel code BER and the useful message E_b/N_0, it would more interesting to break the time correlation between the errors introduced by a channel. In doing so, we could use the optimal relationship between the channel code BER and the useful message E_b/N_0. Nevertheless, note that for mobile channels, in addition of introducing consecutives symbol errors, the instantaneous E_b/N_0 is not the same for all the bits. Therefore, in these channels, we can really use a close relationship to the optimal one between the channel code BER and the useful message E_b/N_0 if the temporal

correlation of the errors is broken. And the correlation break of the errors is accomplished by introducing an interleaver.

The objective of an interleaver is to break the temporal relationship between consecutive bits in order to distribute uniformly the errors along the word and to distribute uniformly the low and high C/N_0 bits along the word as well. Therefore, it can be concluded that the increase of the capacity of an interleaver leads to the break of the longer series of time. Note that this statement assumes that it is made an optimal use of the interleaver capacity.

6.2.5. Influence of the data message content

The data message content belongs to the group of the first factor since it changes the useful C/N_0 value employed to demodulate the message or the E_c/N_0 value.

A particularity of the GNSS navigation messages is that they broadcast several times the same ephemeris data set; therefore it would be a great advantage for the receiver to be able to accumulate coherently different words carrying the same information in order to increase the useful C/N_0 value employed to demodulate the message. More specifically, if some information is repeated on the n^{th} frame and on the $(n+1)^{th}$ frame, part of the coded word containing this information can be equal in both frames. More specifically, the equal part of the coded words depends on the implemented channel code. Therefore, the receiver can simply add coherently the i^{th} sampled value of the n^{th} frame to the i^{th} sampled value of the frame $(n+1)^{th}$ when both symbols are equal at the emission. In this case, the resulting coherently added signal has an increased C/N_0 value because the useful signal is coherently added whereas the noise is not, since the noise values of the coherently added samples are independent. Therefore, if after the coherent addition the new sample is divided by the number of added components, the useful signal power remains constant whereas the noise is averaged and transformed into a new noise with a lower power spectral density level. Nevertheless, note that in order to accomplish this C/N_0 increase, the receiver must know in advance which symbols are equal. This means that the navigation signal has to carry the same navigation message for a given period of time and has to determine the time periods slots where the same navigation message is repeated.

Moreover, this coherent addition also provides a gain of diversity if the transmission channel is modeled as a Rayleigh channel. This gain of diversity also exists for Rice or Loo channels but in a lower scale [PROAKISg, 2001]. Note that this gain means that the signal needs a lower value of accumulated C/N_0 than the value of C/N_0 required when only one word is used to demodulate the message in order to obtain the same value of BER.

Finally, two remarks can be made concerning the influence of the channel code on the repetition of information between consecutive words. First, if all the information is repeated from one word to the next one, the two coded words are always identical regardless of the implemented channel code (except maybe for the first bits if a convolutional channel code without tail bits is implemented). Second, if the information is not exactly repeated from one word to the next one, the coded words are never identical. However, depending on the implemented channel code, if there is a part of the information which is repeated from one word to the next one, the two coded words will not be identical but they can have identical parts. The mathematical model presenting the exact increase of C/N_0 or E_c/N_0 due to the coherent accumulation of words is given in section 6.3.1.2.

6.3. Preliminary theoretical study

In this section, a preliminary theoretical study of the relevant factors conditioning the demodulation performance is made and the results are particularized for each navigation signal. This study is achieved by following two steps, one step for each factor defined in subsection 6.2. Therefore two individual subsections are presented below in order to analyze each step.

It is assumed along this section that the code delay tracking process is ideal and that the level of total power at the receiver RF/IF block output is the same for GALILEO E1 OS signal as for GPS L1C signal.

6.3.1. Relation between signal C/N_0 and message E_c/N_0

The elements belonging to the group of the first factor defined in section 6.2 are the signal channel relative power distribution, the symbol transmission rate and the data message content. These elements are used to model the useful message E_c/N_0 value employed to decode the message as a function of the signal C/N_0 at the receiver RF/IF block output and they can be divided into two groups depending on how they influence this relationship.

The signal channel relative power distribution and the symbol transmission rate are the elements which determine the conversion between the signal C/N_0 at the receiver RF/IF block output and the message E_c/N_0. The data message content can allow the increase of the useful $(C/N_0)_{data}$ value at the receiver correlator output or the E_c/N_0 value by means of coherently accumulating several words. Therefore, the analysis conducted in this section is divided into two parts. The first part studies the conversion between the C/N_0 at the RF/IF block output and the E_c/N_0 value, and the second part analyzes the gain obtained by the coherent accumulation of words.

6.3.1.1. Theoretical conversion between signal C/N_0 and message E_c/N_0

In this subsection, the mathematical model relating the signal C/N_0 at the receiver RF/IF block output to the message E_c/N_0 is presented. Moreover, note that this mathematical model can be used to calculate an initial demodulation performance difference in dB between GPS L1C and GALILEO E1 OS signals when only one word is used to demodulate the ephemeris data.

The mathematical expression relating C/N_0 to E_c/N_0 is shown below. This expression has been calculated by using equations (6-3), (6-4) and (6-6).

$$\frac{E_c}{N_0}_{dB} = \frac{C}{N_0}\bigg|_{totaldB} - f_{tracking}\left(\frac{C}{N_0}\bigg|_{totaldB}, D, v\right)\bigg|_{dB} - 10\log\left(R_D \cdot D\right) \tag{6-8}$$

$$f_{tracking}\left(\frac{C}{N_0}\bigg|_{totaldB}, D, v\right)\bigg|_{dB} = 10 \cdot \log 10\left[f_{tracking}\left(\frac{C}{N_0}\bigg|_{totaldB}, D, v\right)\right] \tag{6-9}$$

Where:

- E_c/N_0 : Energy per coded bit to power density noise at the correlator output
- $C/N_{0\,total}$: Total signal C/N_0 at the receiver RF/IF block output

- R_D: Symbol transmission rate
- D: Inverse of the percentage of the total power provided to the data channel
- v: Receiver speed
- $f_{tracking}(X)$: Attenuation factor of the signal component used for demodulation due to tracking.

Some remarks can be made about equation (6-8). First, this equation concerns the E_c/N_0 of symbols representing only one coded bit. Second, the function calculating the attenuation coefficient due to tracking of the signal is a simplification since it only takes as inputs the total signal C/N_0 at the receiver RF/IF block output, the inverse of the percentage of the total power provided to the data channel and the receiver speed. In fact, the function providing the attenuation coefficient should also include the type of oscillator of the PLL, the characteristics of the Allan deviation noise and the oscillator vibrations, etc, in order to fully characterize the attenuation coefficient. Nevertheless, the used inputs of the function are enough to present a first expression of the attenuation coefficient which shows the dependence of this coefficient with the different characteristics of each signal.

Starting from this initial expression relating the C/N_0 to the E_c/N_0, an average value for the attenuation coefficient due to tracking of the signal propagated through an AWGN channel can be found in order to further simplify the study. More specifically, the attenuation coefficient due to the tracking of the signal is computed for a tracking process conducted by a PLL but only taking into account the thermal noise as source of errors. The reason is that the other sources of errors are not conditioned by the C/N_0 at the receiver antenna output, which means that they are not influenced by the signal characteristics. Therefore, there is no interest in considering them in this preliminary study. Additionally, note that since the modeled channel is an AWGN channel, the influence of the receiver speed on the attenuation coefficient can be removed even if the user is in motion (assuming a constant velocity). Finally, after all the simplifications, the average coefficient factor is calculated in a similar way as done in section H.3.1.2.1.

$$f_{tracking}\left(\left.\frac{C}{N_0}\right|_{total}\right) = \int_{-\infty}^{+\infty} \cos^2(\varepsilon_\theta) \cdot p(\varepsilon_\theta) \cdot d\varepsilon_\theta \tag{6-10}$$

$$p(\varepsilon_\theta) = \sqrt{\frac{(1-1/D)\cdot(C/N_0)_{total}}{2\pi \cdot B_L}} \cdot \exp - \left(\frac{(1-1/D)\cdot(C/N_0)_{total}}{2 \cdot B_L} \cdot \varepsilon_\theta \right) \tag{6-11}$$

Where:

- B_L: PLL filter bandwidth
- ε_θ: Error between the incoming signal carrier phase and the local carrier phase generated by the PLL. ε_θ follows a Gaussian distribution with a null average and a variance defined by the type of discriminator of the PLL, in this case a Q discriminator. The different expressions of the variance are described in section 3.1.3.2.
- $p(\varepsilon_\theta)$: probability of ε_θ.

Finally, this simplified expression is evaluated for the characteristics of each navigation signal in order to compare the E_c/N_0 values of each signal. The results are presented below.

For GPS L1C, the numerical values are T_D = 1/100 seconds, D equal to 4 and B_L equal to 10 Hz.

$$\frac{E_c}{N_{0\ dB}} = \frac{C}{N_{0\ dB}} - 10 \cdot \log_{10}\left(\sqrt{\frac{3}{80\pi} \cdot \frac{C}{N_0}} \cdot \int_{-\infty}^{+\infty} \cos^2(\varepsilon_\theta) \cdot \exp\left(-\frac{3}{80} \frac{C}{N_0} \cdot \varepsilon_\theta \right) \cdot d\varepsilon_\theta \right) - 26 \qquad (6\text{-}12)$$

For GALILEO E1 OS, the numerical values are T_D equal to 250 bps, D equal to 2 and B_L equal to 10 Hz.

$$\frac{E_c}{N_{0\ dB}} = \frac{C}{N_{0\ dB}} - 10 \cdot \log_{10}\left(\sqrt{\frac{1}{40\pi} \cdot \frac{C}{N_0}} \cdot \int_{-\infty}^{+\infty} \cos^2(\varepsilon_\theta) \cdot \exp\left(-\frac{1}{40} \frac{C}{N_0} \cdot \varepsilon_\theta \right) \cdot d\varepsilon_\theta \right) - 27 \qquad (6\text{-}13)$$

From these numerical results, some preliminary conclusions can be made. First, GPS L1C signal obtains a larger E_c/N_0 value than GALILEO E1 OS signal for the same C/N_0 value at the receiver antenna output. This means that if both signals had the same channel code, the users receiving GALILEO E1 OS signal would need a higher C/N_0 at the receiver RF/IF block output in order to obtain the same navigation message demodulation performance.

Second, the extra power required by GALILEO E1 OS signal in order to obtain the same demodulation performance as GPS L1C signal if both signals had implemented the same channel code would always be at least 1dB bigger. The reason is that in equations (6-12) and (6-13) the attenuation coefficient due to tracking of the signal grows more quickly for GALILEO E1 OS than for GPS L1C as a function of the total signal C/N_0 at the receiver RF/IF block output. And this means that whereas for a high C/N_0 the tracking performance term can be neglected, for a low C/N_0 the tracking performance term becomes significant. Therefore, since the constant term of equations (6-12) and (6-13) which does not depend on the total C/N_0 is 1dB smaller for GPS L1C signal than GALILEO E1 OS signal, GALILEO E1 OS signal will always need at least 1 extra dB to obtain the same demodulation performance as GPS L1C signal.

Third, this model assumes an AWGN channel where the C/N_0 is considered constant or following a very slow variation. Therefore, in this case, the carrier phase tracking process is easily achieved when the C/N_0 is high enough. However, the carrier phase tracking process becomes more complicated to achieve when the transmission is made through a mobile channel. And this means that in this case, the higher distribution of power assigned to the pilot channel by the GPS L1C signal in relation to the GALILEO E1 OS signal should farther increase the difference between signals by means of equations (6-12) and (6-13) tracking performance term.

6.3.1.2. Coherent accumulation of words

The differences on the required C/N_0 for an identical E_c/N_0 presented at the end of the previous subsection were obtained assuming that only one word is used to demodulate the navigation message. However, due to the data message content, the signals can allow the coherent accumulation of several words containing the same information in order to improve the message E_c/N_0. In fact, the message coherent accumulation process will likely be a common strategy of any future GNSS receiver since it improves significantly the signal demodulation performance. Note that this demodulation performance improvement can be very important in mobile channels.

Therefore, in order to analyze the influence of data message content of each signal onto the message E_c/N_0, the gain obtained by each signal due to the coherent accumulation of words is presented in this section. Nevertheless, before analyzing the signals gain, the ideal case where the words are identical is studied in order to compare the improvement of the signals E_c/N_0 with the ideal case gain.

In this section, the gain diversity introduced by the coherent accumulation of words is not presented. The reason is that this gain is significant for Rayleigh channels but not so much for Loo channels; the type of channel implemented in the simulations of this research work.

6.3.1.2.1. Coherent accumulation of identical words

The gain provided by the coherent accumulation of identical message words is calculated by first modeling the coherent accumulation of two words and second by generalizing the obtained expression to any number of words. The reception of two consecutives words and the message E_c/N_0 can be modeled as shown below. These signals are modeled at the receiver correlator output.

$$r_l[k] = A \cdot d[k] + n[k] \tag{6-14}$$

$$\begin{cases} S_1[k] = r_l[k] = A \cdot d[k] + n[k] \\ S_2[k] = r_l[k + m \cdot L] = A \cdot d[k + m \cdot L] + n[k + m \cdot L] \end{cases} \quad 0 \le k < L \tag{6-15}$$

Where:

- $r_l[k]$: Received signal at the receiver correlator output
- $S_1[k]$: Sample k of the 1st received word at the receiver correlator output
- $S_2[k]$: Sample k of the 2nd received word at the receiver correlator output
- A: Useful signal amplitude at the receiver correlator output
- d[k]: Coded bit k
- n[k]: Gaussian noise, $N(0, \sigma^2 = N_0 \cdot R_D)$
- L: Number of symbols composing of a word
- m: Number of words between the identical first and second received words

Knowing that the words are identical and thus that they carry the same information symbols, we can model equation (6-15) as:

$$\begin{cases} S_1[k] = A \cdot d[k] + n_1[k] \\ S_2[k] = A \cdot d[k] + n_2[k] \end{cases} \tag{6-16}$$

$$\begin{cases} n_1[k] = n[k] \\ n_2[k] = n[k + m \cdot L] \end{cases} \tag{6-17}$$

Where:

- $n_1[k]$, $n_2[k]$: Gaussian noises, $N(0, \sigma^2 = N_0 \cdot R_D)$. $n_1[k]$ and $n_2[k]$ are independent since they represent different temporal samples of the same narrow-band Gaussian noise.

The coherent accumulation of words S1 and S2 is equal to:

$$S^{2w}[k] = S_1[k] + S_2[k] = 2A \cdot d[k] + n_1[k] + n_2[k] \tag{6-18}$$

The E_c/N_0 of the reception of only one word, either $S_1[k]$ or $S_2[k]$, is equal to:

$$\frac{E_c^{1\ word}}{N_0} = \frac{P_{S1}}{P_N} = \frac{P_{S2}}{P_N} = \frac{A^2}{N_0 \cdot R_D} = \frac{A^2 \cdot T_D}{N_0} \tag{6-19}$$

And the E_c/N_0 when the receiver coherently accumulates the two words is equal to:

$$\frac{E_c^{2\ words}}{N_0} = \frac{P_{S_1+S_2}}{P_N} = \frac{(A+A)^2}{2 \cdot N_0 \cdot R_D} = \frac{2^2 \cdot A^2 \cdot T_D}{2 \cdot N_0} = \frac{2 \cdot E_c^{1\ word}}{N_0} \tag{6-20}$$

Therefore, from equation (6-20), it can be seen that the coherent accumulation of two identical words implies an increase of 3dB of the E_c/N_0 of the final signal used for decoding. And this gain is improved with the coherent accumulation of more words. More specifically, following the same process as in equation (6-18), the general expression of the gain of E_c/N_0 for any number of coherently accumulated words is:

$$Gain\ E_c / N_0 = 10 \cdot \log_{10}(number\ of\ accumulated\ subframes) \tag{6-21}$$

6.3.1.2.2. Coherent accumulation of GPS L1C subframes carrying the same ephemeris data set

For GPS L1C signal, as said in section 6.1.5, the same ephemeris and clock correction data are transmitted for at least 15 minutes. Therefore, since the subframe carrying the ephemeris and clock correction data does not contain any other information, it can be assured that all the subframes inside the 15 minutes period are identical.

Moreover, the time slots of these 15 minutes periods are well defined; which means that the receiver knows when the change of the ephemeris data happens and thus the receiver is able to avoid the coherent accumulation of two different words. Therefore, it can be concluded that the ideal case gain can be obtained with GPS L1C.

6.3.1.2.3. Coherent accumulation of GALILEO E1 OS pages carrying the same ephemeris data set

For GALILEO E1 OS signal, as said in section 6.1.5, the 4 pages (8 parts of a page) carrying the ephemeris and the clock correction data also contain the information required to provide the SoL and integrity services. Therefore, since each page has implemented a channel code, it cannot be guaranteed that pages carrying the same ephemeris and clock correction data are identical. However, knowing the two channel codes implemented over each page (over one part of a page and over the other one), the identical parts of the pages carrying the same ephemeris data set can be identified.

Each page is formed by two parts which are encoded individually by a convolutional code. The first part of a page only contains a part of the ephemeris data set and a part of the clock correction data. Therefore, the first parts of any page carrying the same ephemeris data set are identical.

The second part of a page, in addition to containing the remaining ephemeris and clock correction data information also carries the SoL information, the integrity information and the

CRC bits resulting from the CRC-24Q channel code application over the entire page bits. More specifically, the second part of a page first contains the ephemeris information, then the SoL and integrity bits, and finally the CRC bits. This means that the entire second part of a page is not identical for any page carrying the same ephemeris data set since the SoL and integrity information value are unknown. Nevertheless, due to the distribution of the bits into the second part of a page and due to the memory of a convolutional code –the value of an encoded bit depends on the value of the previous information bits–, the bits of some parts of the second part of a page are identical to the bits of the same parts of another second part of page when they carry the same ephemeris data set. More specifically, the part containing the ephemeris data is always the same but it cannot be guaranteed that the remaining parts values are.

Nevertheless, the previous analysis is only valid when it can be guaranteed that two or more pages carrying the same ephemeris set are coherently accumulated, which is not true for GALILEO E1 OS signal. More specifically, GALILEO E1 OS signal does not define any period of time inside which the transmitted ephemeris set is constant. Therefore, this signal does not define time slots where the broadcasted ephemeris set is the same. And this means that there is no guarantee to identify a priori that two frames are carrying the same ephemeris set. Moreover, it remains to consider the alarm page, which has not yet been completely defined in the current public documents version [ESA, 2008], but which indicates a random value inside a frame.

To summarize, when two pages carrying the same ephemeris data set are coherently accumulated, GALILEO E1 OS signal obtains the ideal case E_c/N_0 gain over more than 50% of the bits of the pages carrying the ephemeris data set and does not obtain any gain at all for the remaining bits of the page. Nevertheless, the coherent accumulation of words carrying the same ephemeris data set cannot be guaranteed for this signal due to the alarm page and because no time slots of constant ephemeris data set have been defined.

Finally, note that one interpretation of the E_c/N_0 gain obtained by only a part of the bits belonging to an ephemeris page is the following. Before accumulating the words, all the page bits are equally weak against the noise; however, once the accumulation has been done, the page constant parts are more robust to the noise and the errors can be more easily corrected in these parts.

6.3.1.2.4. Conclusions of the comparison between GALILEO E1 OS signal and GPS L1C signal

The main conclusion extracted from the previous subsections is that due to the data message content of each signal, the coherent accumulation of words containing the same ephemeris set provides a larger E_c/N_0 gain for GPS L1C than for GALILEO E1 OS signal.

Moreover, in addition to provide a larger E_c/N_0 gain, the coherent word accumulation process is achieved faster for GPS L1C than for GALILEO E1 OS since the frame duration is smaller for GPS L1C. Indeed, for GALILEO E1 OS the frame lasts 30 seconds, while for GPS L1C each 18 seconds a new frame is broadcasted.

6.3.2. Relation between message E_c/N_0 and the demodulation performance

The elements belonging to the group of the first factor defined in section 6.2 are the data message structure and the interleaver. These elements determine the demodulation

performance, BER, WER and EER, obtained by each signal as a function of the message E_c/N_0.

This final part of the preliminary theoretical study begins by assuming that the transmission channel is AWGN. This means that the interleaver has no influence on the final demodulation performance. Afterwards, some comments about the interleaver and its influence over the signal transmission through mobile channels are made.

6.3.2.1. Additive White Gaussian Noise channel

Both signals have a channel code with a channel code rate equal to ½. Therefore, the relationship between the E_b/N_0 available for each signal and the total signal C/N_0 at the receiver antenna output can be extracted from equations (6-12) and (6-13). Equation (6-22) presents the expression for GPS L1C signal and Equation (6-23) presents the expression for GALILEO E1 OS signal:

$$\frac{E_c}{N_{0\ dB}} = \frac{C}{N_{0\ dB}} - 10\cdot\log_{10}\left(\sqrt{\frac{3}{80\pi}\cdot\frac{C}{N_0}}\cdot\int_{-\infty}^{+\infty}\cos^2(\varepsilon_\theta)\cdot\exp\left(-\frac{3}{80}\frac{C}{N_0}\cdot\varepsilon_\theta\right)\cdot d\varepsilon_\theta\right) - 23 \qquad (6\text{-}22)$$

$$\frac{E_c}{N_{0\ dB}} = \frac{C}{N_{0\ dB}} - 10\cdot\log_{10}\left(\sqrt{\frac{1}{40\pi}\cdot\frac{C}{N_0}}\cdot\int_{-\infty}^{+\infty}\cos^2(\varepsilon_\theta)\cdot\exp\left(-\frac{1}{40}\frac{C}{N_0}\cdot\varepsilon_\theta\right)\cdot d\varepsilon_\theta\right) - 24 \qquad (6\text{-}23)$$

Note that the value of the C/N_0 is the same for the 3 times that this term appears in equations (6-22) and (6-23) since in all 3 cases it represents the total C/N_0 at the receiver antenna output.

From both equations, it can be seen that the difference of available E_b/N_0 is the same as the difference of E_c/N_0 since the channel code rate is the same for both signals. Therefore, it only remains to analyze the decoding curve of each signal channel code in order to obtain the expected demodulation performance of each signal. However, note that since the curves of the channel codes are not linear, the previous expressions provide an average E_b/N_0 which can be used to calculate an approximation of the BER obtained by the signal. However, this expression is no longer an approximation but becomes the exact expression when the tracking performance term is assumed ideal, i.e. when the middle term is 0.

The curves of the channel code of each signal were already calculated in [DAFESH et al., 2007] for GPS L1C and in [PROAKISe, 2001] for GALILEO E1 OS. Taking them as references, the difference of BER performance between the signals can be determined. In order to obtain a BER equal to 10^{-3}, GPS L1C channel code needs 1.5dB less in terms of E_b/N_0 than GALILEO E1 OS channel code, and in order to obtain a BER of 10^{-5}, the difference is equal to 2.2dB.

Finally, GALILEO E1 OS signal also needs a bigger E_b/N_0 value than GPS L1C in order to obtain the same WER since this figure of merit depends on the BER value. And due to the division of the ephemeris data set into 4 words in GALILEO E1 OS signal, the difference of E_b/N_0 required by both signals in order to obtain the same EER value is larger than the difference of E_b/N_0 found for the WER value. This last statement is justified in section 6.2.3.

6.3.2.2. Mobile channel

The relationship between the message E_b/N_0 and the signal C/N_0 at the receiver antenna output for the GPS L1C signal and GALIEO E1 OS signal when they are transmitted through a mobile channel can be modeled from equations (6-22) and (6-23). In fact, as said at the end of section 6.3.1.1, the term modeling the tracking performance losses should be modified and this should result into an increase of the difference between the signals. Therefore, for a mobile channel, expressions (6-22) and (6-23) can be taken as E_b/N_0 upper bounds, and the difference between them can be considered as a lower bound of the difference between signals.

Now, once the E_b/N_0 has been determined, it only remains to relate this value with the signal channel code decoding performance curve. As explained in section 6.2.4, the curves of the channel codes are obtained for an AWGN channel when the bits errors are independent, but, for a mobile channel, the bits errors appear in bursts. Therefore, in order to break these bursts, an interleaver is introduced.

The interleaver applied for GPS L1C signal has a bigger capacity than the interleaver applied for GALILEO E1 OS signal. Moreover, the symbol transmission rate is higher for GALILEO E1 OS signal which means that GPS L1C signal can support much larger zones of low E_b/N_0. The next table illustrates the maximum duration of a low E_b/N_0 zone supported by each signal before allowing two consecutive coded bits be erroneous. This table is easily calculated by multiplying the duration of a symbol by the number of symbols introduced by the interleaver between two consecutives symbols.

Signal	GPS L1C	GALILEO E1
Maximal low E_b/N_0 zone duration	380 ms	120 ms

Table 6-3: Maximal duration of a low E_b/N_0 zone before having two consecutive error bits

From Table 6-3, it can be seen that GPS L1C signal accepts a burst of errors about 3 times longer than the burst of errors allowed by GALILEO E1 OS signal before presenting two consecutive bit errors. Therefore, only taking into account this fact, it can be concluded that GPS L1C signal interleaver is much more powerful than GALILEO E1 OS signal interleaver, and that the former signal is much more adapted for mobile channels. However, there are two additional factors to take into account which further improves the GPS L1C interleaver in comparison with the GALILEO E1 OS interleaver. First, GPS L1C signal interleaver separates farther apart two consecutive channel bit errors than the GALILEO E1 OS interleaver does. More specifically, when a burst of errors is received, GPS L1C interleaver inserts 45 bits between two consecutive error bits, whereas GALILEO E1 OS only inserts 7 bits. Second, GPS L1C interleaver is applied over two different subframes whereas GALILEO E1 OS interleaver is applied over the same page. Therefore, GPS L1C interleaver can distribute the errors between the subframes, and this can help the final decoding. The reason is that the errors introduced by the channel are uniformly distributed in the two subframes. This implies that the variance of the numbers of errors of each subframe is decreased and that the average of the number of errors of each subframe remains the same. Therefore, more subframes can be decoded since the possibility of having a subframe with a large number of errors is reduced as the variance decreases.

Finally, in addition to its more powerful interleaver, GPS L1C signal is also better adapted to transmissions through mobile channels than GALILEO E1 OS signal. This is due to the better

performance of its implemented channel block code LDPC in the presence of burst of errors than the GALILEO E1 OS convolutional code.

6.3.3. Conclusions of the comparison between GALILEO E1 OS and GPS L1C signal

The first conclusion extracted from the theoretical comparison is that if both signals had implemented the same channel code with the same interleaver, GALILEO E1 OS signal would always need at least 1 extra dB at the receiver antenna output in order to obtain the same performances as GPS L1C signal. This is due to the different relative power channel distribution and the different symbol transmission rate of each signal.

However, both signals have implemented different channel codes. The code used by GPS L1C is much more powerful than the code used by GALILEO E1 OS. Therefore, at least 3.2 extra dB are required for the latter signal than the former signal in order to obtain a BER equal to 10^{-5}. This means that GALILEO E1 OS signal also needs a larger C/N_0 at the receiver antenna output than GPS L1C in order to obtain the same WER, and this difference is even bigger for the EER figure of merit.

Finally, the differences between BER, WER and EER of both signals are increased when the signals are transmitted through a mobile channel since the interleaver is better for GPS L1C signal, its channel code is more robust to bursts of errors, and also because this signal provides more power than GALILEO E1 OS signal to the pilot channel. Remember that the tracking process is more difficult for a mobile channel and that this process is made using the pilot channel.

It can be concluded from the theoretical study that GPS L1C signal outperforms GALILEO E1 OS from the demodulation point of view.

6.4. Simulation results

This section presents the demodulation performance results obtained for GALILEO E1 OS signal and GPS L1C signal in different simulated environments. More specifically, these environments are an AWGN channel and a mobile channel. These results are the BER, WER and EER obtained by each signal as a function of either the message E_b/N_0 or the signal C/N_0 at the receiver antenna output.

These results were calculated by simulations which characterize the GPS L1C signal or GALILEO E1 OS signal transmission through the considered channels. The chosen simulator structure and the channel characteristics have been defined in sections 3.1.5 and 3.2.5, which means that the simulator does not generate samples at the real signal sampling frequency but at a much lower rate. Moreover, the simulator even directly generates the receiver correlator output signal samples for the AWGN channel case.

Nevertheless, before presenting the results of both methods, the assumptions and conditions under which the simulations were conducted are described. Finally, a summarizing table is presented at the end of the section.

6.4.1. Assumptions of the analysis

The assumptions taken into account for this analysis are the same assumptions as for 0: an ideal code delay tracking process and a seamless synchro-frame process However, two

possible assumptions about the carrier phase tracking process are considered in this chapter. The first assumption is applied for a signal transmitted through an AWGN channel and it consists in assuming an ideal carrier phase tracking process. The second assumption is applied for a signal transmitted through a mobile channel and it consists in assuming a carrier phase tracking process only corrupted by thermal noise. The dynamic stress error, which is defined in section 3.1.3.2 and A.3.3.2, was not considered. The reason is that when the user travels by car at 30 km/h, the GPS receiver is assumed to be attached at the vehicle. And this means that no dynamics higher in order than the acceleration are found in the received signal. Nevertheless, the dynamic stress error could be added for a pedestrian travelling at 5 km/h since the GPS receiver can be assumed to be a handheld device. Moreover, it has also been assumed a seamless code delay tracking process.

The general mathematical model of the signal at the correlator output can be divided into two more models, the mathematical model of the data channel at the correlator output and the mathematical model of the pilot channel at the correlator output. The mathematical model of each channel is based on equations (3-50) and (3-51) but we ignored the channels PRN code since we are modeling the signal at the correlator output. Moreover, the model is further simplified by assuming that the incoming signal phase is introduced by the channel. The mathematical expression is shown below.

$$r_{pilot}[k] = \cdot\left(A \cdot c_1[k]\right)/\left(1 - 1/D\right) + n_{pilot}[k] \tag{6-24}$$

$$r_{data}[k] = \left(A \cdot c_1[k] \cdot d_m\right)/D + n_{data}[k] \tag{6-25}$$

Where:

- $r_{pilot}[k]$: Pilot channel at the receiver correlator output
- $r_{data}[k]$: Data channel at the receiver correlator output
- A: signal amplitude
- d_m: navigation data
- D: relative signal channel power distribution
- $c_1[k]$: Complex coefficient defining the signal transmission through a channel.
    - AWGN channel: The complex coefficient is equal to 1
    - Mobile channel: Loo random variable described in section 3.2.2.3.4
- $n_{data}[k]$, $n_{pilot}[k]$: Gaussian noise with $N(0, \sigma^2)$

Note that this model is justified in subsections 3.1.5 and 3.2.5.

Finally, the last assumption taken into account for the following analysis is that the receiver needs 1 second to reacquire the signal from a hot start each time that the PLL loses its lock.

6.4.2. Demodulation performance for an AWGN channel transmission

In this section, the ephemeris demodulation performance of the GPS L1C signal and GALILEO E1 OS signal when they are transmitted through an AWGN channel is presented. Nevertheless, instead of directly showing the signals BER, WER and EER values as a function of the signal C/N_0 at the receiver antenna output, the individual decoding performance of each channel code implemented by each signal is first presented.

In fact, we chose to show first the decoding performance of the signal channel code in order to better analyze the difference of performance of the GALILEO E1 OS and GPS L1C signals. Indeed, the individual analysis of the channel code decoding performance allows a better identification of the factors which provide a better demodulation performance for one signal with respect to the other signal.

Moreover, in addition to inspecting the decoding performance of the channel codes, the improvement of performance obtained through the accumulation of words is also presented.

The following figure illustrates the results of the BER comparison.

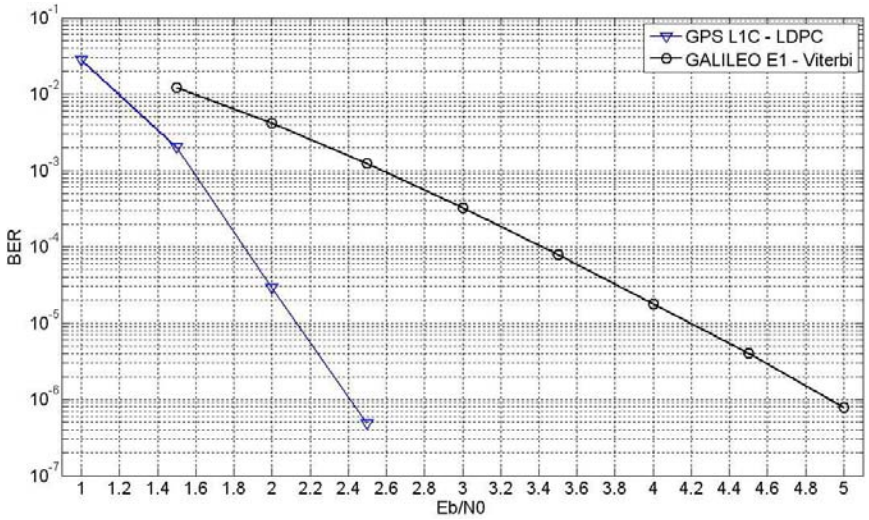

Figure 6-1: BER comparison between GPS L1C and GALILEO E1 OS signal channel codes

In Figure 6-1, it can be observed that the difference of decoding performance between the GPS L1C channel code and the GALILEO E1 channel code is about 1dB favorable to GPS L1C for a BER equal to 10^{-3}, about 2.1dB for BER equal to 10^{-5} and about 2.5dB for BER equal to 10^{-6}.

Additionally, Figure 6-1 does not show the improvement due to the accumulation of words since for GALILEO E1 OS all the ephemeris data bits cannot be accumulated.

The next figure illustrates the results of the WER comparison.

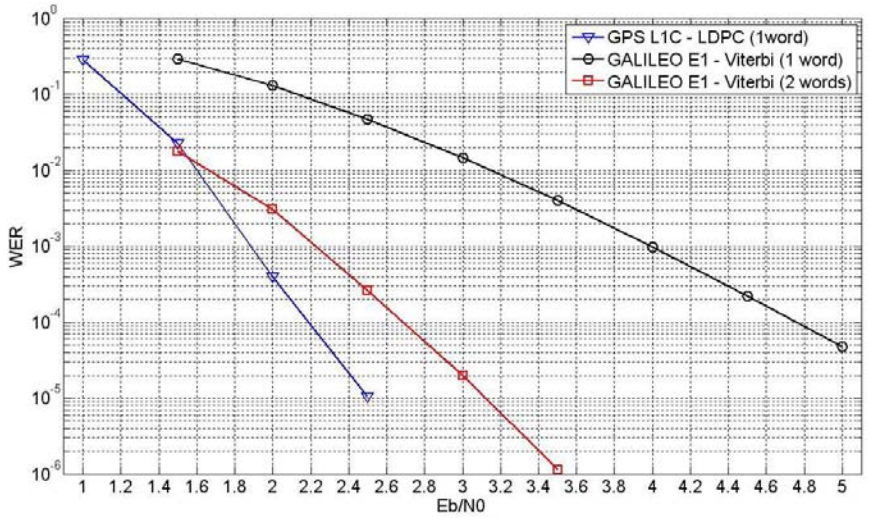

Figure 6-2: WER comparison between GPS L1C and GALILEO E1 OS signal channel codes

From Figure 6-2, it can be observed that when only one word of each navigation message has been received, GALILEO E1 navigation message needs 2.1 extra dB in comparison with GPS L1C navigation message to obtain the same WER equal to 10^{-3} and 2.5 extra dB to obtain the same WER equal to 10^{-4}.

Moreover, Figure 6-2 shows that GPS L1C navigation message obtains a better demodulation performance when only 1 word has been received than GALILEO E1 OS navigation message does when this latter signal accumulates 2 words. Besides, the case where the receiver accumulates 2 words before decoding the GPS L1C navigation message ephemeris has been tested but it is not presented because even for the E_b/N_0 value equal to 1dB, no wrong word has been found among the 400000 tested words. Therefore, it can be concluded that the accumulation strategy works better for GPS L1C signal than for GALILEO E1 signal as it was expected.

The last figure of the section illustrates the EER comparison.

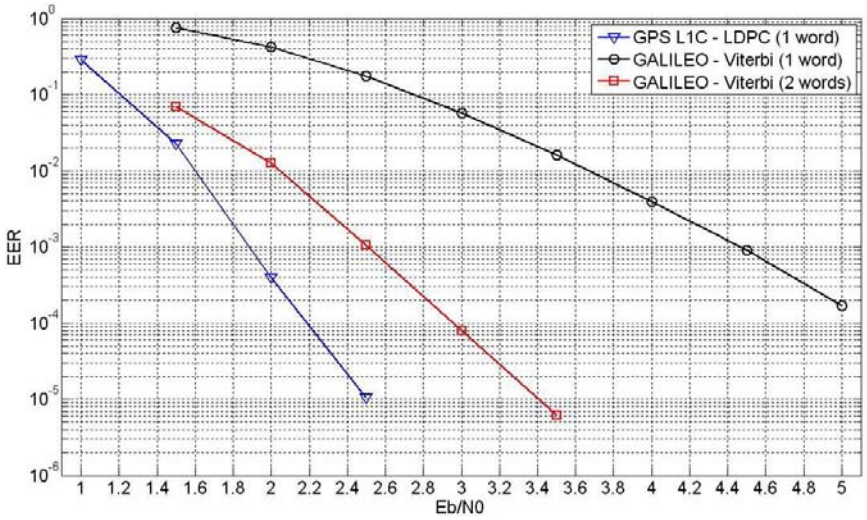

Figure 6-3: EER comparison between GPS L1C and GALILEO E1 OS signal channel codes

From Figure 6-3, it can be observed that when only one word of each navigation message has been received, GALILEO E1 navigation message needs 2.1 extra dB in comparison with GPS L1C navigation message to obtain the same EER equal to 10^{-2} and 2.6 extra dB to obtain the same EER equal to 10^{-3}. Moreover, the same remarks about the word accumulation strategy made for the WER figure are valid for the EER figure.

Besides, from Figure 6-2 and Figure 6-3 it can be observed that the strategy of dividing the ephemeris data set into 4 words provides a worse EER value than the strategy of transmitting the ephemeris data set into only 1 word. This statement is easily verified by comparing the difference in E_b/N_0 between GALILEO E1 signal and GPS L1C for a WER equal to the EER. The E_b/N_0 required by GPS L1C signal is the same for WER = EER = 10^{-3} and it is equal to 1.9dB. However, for GALILEO E1 signal, this value is lager for the EER; 2.2dB are needed for the WER and 2.5dB are needed for the EER. Therefore, since GALILEO E1 signal needs 0.3 extra dB to obtain an EER equal to 10^{-3} than this signal needs to obtain a WER equal to 10^{-3}, it can be concluded that the division strategy fails. And the reason is that if GALILEO E1 signal had implemented over only one word a channel code which provides a WER equal to 10^{-3}, this signal would not need 0.3 extra dB to obtain an EER equal to 10^{-3}.

Finally, only an offset has to be added to the E_b/N_0 values of each navigation signal in order to transform their required E_b/N_0 values into their total C/N_0 values at the receiver antenna

output when the signals are transmitted through an AWGN channel and the carrier phase tracking is assumed ideal. This value is simply the constant term calculated in equations (6-22) and (6-23), which is 23dB for GPS L1C and 24dB for GALILEO E1 OS, since the carrier phase tracking is assumed ideal. Therefore, all the differences calculated in this section have to be increased by 1dB in order to compare the GALILEO E1 OS and GPS L1C signals demodulation performance when they are transmitted through an AWGN channel and when the carrier phase tracking is assumed ideal.

Additionally, these differences should be smaller when the carrier phase tracking process is not ideal and the transmission is still made through an AWGN channel. The reason is that although GPS L1C provides more percentage of power to the pilot channel than GALILEO E1 signal, it is the latter signal which provides more total power to the pilot channel for the inspected values of BER, WER and EER. Indeed, GPS L1C signal provides 1.75 extra dB to the pilot channel than GALILEO E1 OS signal does, but the signal C/N_0 at the receiver antenna output for the latter signal when obtaining the desired levels of BER, WER or EER is more than 1.75dB larger than the C/N_0 required for the former signal. Therefore, without entering into the details, since GPS L1C loses more useful E_b than GALILEO E1 OS signal (due to carrier phase tracking degradation), the former signal requires a larger addition of power than the latter signal.

6.4.3. Demodulation performance for a mobile channel transmission

In this section, the ephemeris demodulation performance obtained for GALILEO E1 and GPS L1C signal when they are transmitted through a mobile channel are presented. Three different cases are analyzed in this section. The first two cases are analyzed using the same simplified channel mathematical model based on the complete model presented in section 3.2.2.3, where the first case represents a receiver travelling at 30 km/h and the second case represents a receiver travelling at 5 km/h. The third case uses the complete mathematical model presented in section 3.2.2.3 and represents a mobile travelling at 30 km/h. Nevertheless, a more detailed description of the three cases is given below.

The first two cases inspect the demodulation performance of both signals when the mobile channel mathematical model described in section 3.2.2.3 is simplified to only one of the three defined states. This state is called LOS conditions and it is the state which allows the receiver to better acquire, track and demodulate the signals since its attenuation parameters are the smallest of the three states. Nevertheless, the signal amplitude modeled in this state continues to follow a Loo distribution. Therefore, the first two analyzed cases calculate the best ephemeris demodulation performance which can be obtained in the simulated scenario. The simulated scenario is described in subsection 6.4.3.1 and presents a general scenario of a mobile channel.

Moreover, for the first two cases 3 different demodulation performance for each signal are inspected. These 3 different performances correspond to the 3 different techniques for tracking the signal carrier phase presented in section 3.2.3: the traditional PLL, the ideal pilot channel estimation and the channel estimation method presented in section 3.2.3.2.2. Therefore, in addition to comparing and analyzing the different demodulation performances of each signal, a first preliminary study is done using the optimal technique that can be used to track the signal carrier phase in mobile channels. Additionally, a phase lock detector has been implemented in order to know when the receiver is no longer able to track the signal carrier phase. The complete information on the PLL and both channel estimations techniques can be found in section 3.2.3.

The third case analyzes the ephemeris demodulation performance obtained by the GPS L1C signal when the simulation takes into account the three states of the mathematical model defined in section 3.2.2.3. In this case, the receiver travels at 30km/h and the only carrier phase tracking method analyzed is the PLL.

The coherent integration time of the PLL for all the analyzed cases and scenarios is 20ms and the channel estimation time of the technique described in 3.2.3.2.2 is also equal to 20ms.

Finally, note that the 5 km/h and 30 km/h speeds have been chosen in order to simulate a representative situation of a pedestrian and of a car searching position service into a urban environment. Moreover, note that the selected mathematical model can only be applied when the signal is slowly fading at the correlator output. And for the two chosen speeds, the signal fulfills this requirement as shown in Table 3-4.

6.4.3.1. Mobile channel scenario

The scenario chosen to obtain representative demodulation performance figures such as BER, WER and EER of the GPS L1C and GALILEO E1 OS signals in a mobile environment is presented in this section.

The main characteristics are the satellite elevation angle, the first-order Markov chain probabilities, the Loo random variable parameters, the states frame length and the average correlation length of the slow variations. All these parameters were extracted from the papers [PEREZ-FONTAN et al, 1998] [PEREZ-FONTAN et al, 2001], where the Loo random variable parameters are the values provided by the DLR in an urban environment for the L-Band [PEREZ-FONTAN et al, 2001]. The antenna for which the values of the Loo random variable parameters are given is a hand-held or car roof antenna [PEREZ-FONTAN et al, 2001].

The previous values chosen for this scenario are summarized in Table 6-4 and Table 6-5.

Elevation	State 1: Line-of-Sight			State 2: Intermediate Sh.			State 3: Deep shadow		
	α (dB)	ψ (dB)	MP (dB)	α (dB)	ψ (dB)	MP (dB)	α (dB)	ψ (dB)	MP (dB)
30°	0.45	1.9	-16.9	-11.7	4.8	-21.3	-23.8	9.9	-22.5
	State Frame Length			State Frame Length			State Frame Length		
	8.9 m			8.7 m			4.5 m		
				Average Correlation Length					
				2.42 m					

Table 6-4: Simulation scenario parameters

[P] Elevation 30°	0.8199	0.1273	0.0528	[W] Elevation 30°	0.5051
	0.2551	0.4280	0.3169		0.1126
	0.1629	0.0003	0.8368		0.3823

Table 6-5: First-order Markov chain probabilities

Finally, the satellite elevation chosen for this scenario is low and a GNSS receiver should normally expect to acquire and to track satellites with a higher angle. Nevertheless, in this

research work, it is preferred to present an unoptimistic value of the performance than giving an average or optimistic value.

6.4.3.2. Demodulation performance for a receiver speed of 30km/h at the LOS conditions state

In this section, the GALILEO E1 OS and GPS L1C signals demodulation performance is calculated for the case where the signals are received into a urban environment, when the user travels at a speed of 30 km/h and the carrier phase tracking process is noised by the presence of thermal noise.

Moreover, in the following graphics, in addition to showing the demodulation performance obtained for a mobile channel, the navigation signals demodulation performance obtained for an AWGN channel assuming an ideal carrier phase tracking process is also plotted. Therefore, it is possible to analyze and to compare the demodulation performance degradation suffered by each individual navigation signal.

More specifically, the curves presented in Figure 6-1, Figure 6-2 and Figure 6-3 are used along this section. Nevertheless, these curves were adapted since they were presented as a function of the message E_b/N_0. Therefore, 23 dB have been added to the GPS L1C E_b/N_0 values and 24 dB have been added to the GALILEO E1 E_b/N_0 values. Nevertheless, note that since the chosen scenario presents a positive offset of the LOS signal power of 0.45 dB (average of the log-normal variable, α), 0.45 dB is subtracted from the original 23 dB and 24 dB offsets of the AWGN channel curves in order to better compare the degradation introduced by the mobile channel. This means that the final added offset is 22.55 dB for GPS L1C and 23.55 dB for GALILEO E1.

The next figure illustrates the BER obtained by each signal.

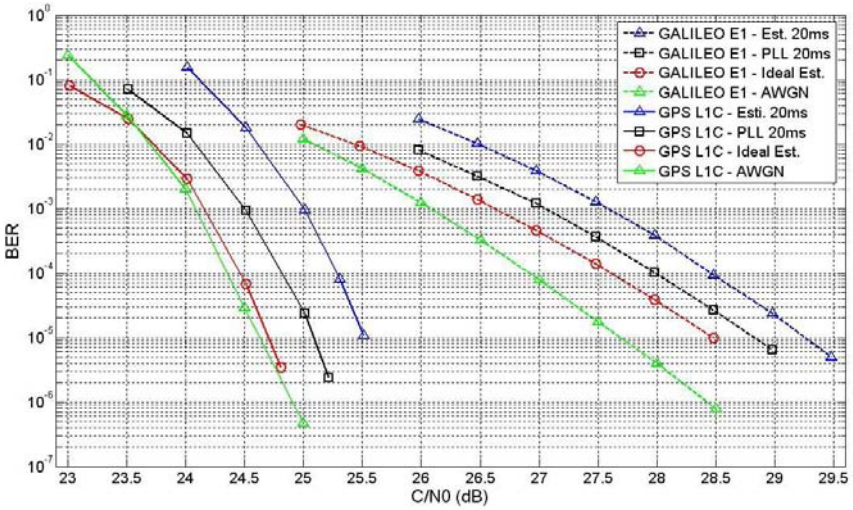

Figure 6-4: BER comparison between GPS L1C and GALILEO E1 OS signals for a mobile channel transmission with a receiver travelling at 30 km/h and the conditions specified in Table 6-4 and Table 6-5

The first thing that can be observed from Figure 6-4 is that the difference of performance between the PLL and the channel estimation method presented in section 3.2.3.2.2 is rather small. For GPS L1C signal the difference is about 0.5 dB for a BER equal to 10^{-3}, and about 0.4 dB for a BER equal to 10^{-5}. Approximately the same difference can be found for GALILEO E1 signal.

Moreover, it can be seen that the ideal pilot channel estimation outperforms the results of the PLL for both signals since the ideal pilot channel estimation requires 0.4 dB less to obtain a BER equal to 10^{-3} or to 10^{-5} than the PLL. This means that the receiver should have the capacity to obtain the desired level of BER with a lower C/N_0 value if a better pilot estimation channel is implemented. In fact, remember that the applied channel estimation method is probably the simplest existing technique and thus is not optimal. Therefore, the red and black curves could be considered as lower and upper channel estimation techniques bounds. However, the difference between the pilot ideal estimation and the PLL demodulation performance is also rather small. This means that in order to use a channel estimation method instead of a PLL, the channel estimation method should be able to demodulate the signal in situations where the PLL cannot be locked since the demodulation gain which can be obtained in the simulated situation is not very significant.

The second observation that can be made from Figure 6-4 is that GPS L1C outperforms GALILEO E1 in terms of BER into a mobile channel scenario when the receiver travels at 30km/h. More specifically, GALILEO E1 needs 2.5 extra dB for a BER equal to 10^{-3} and 3.7 dB for 10^{-5} than GPS L1C signal does when both signals are received using a PLL. Moreover, it can be observed that this difference grows as the C/N_0 increases since the slope of the GPS L1C demodulation performance curve is steeper than the GALILEO E1 curve. Note that the slope of the curves is determined by the channel code implemented in each signal.

Moreover, it can be observed that the difference between GALILEO E1 signal and GPS L1C signal when they are transmitted through this simulated mobile channel is larger than the difference when they are transmitted through an AWGN channel. In fact, it can be seen that the difference between the AWGN channel curve and the mobile channel curve for GALILEO E1 is larger than the difference between the same curves for GPS L1C. For example, GALILEO E1 signal transmitted through a mobile channel requires 1.1 extra dB to obtain a BER equal to 10^{-5} than GALILEO E1 signal transmitted through an AWGN channel does. However, GPS L1C signal only requires 0.6 extra dB. Therefore, it can be concluded that GPS L1C is better adapted to a mobile channel transmission than GALILEO E1. The reasons are given next.

Three factors can be identified as the main different elements affecting the change of demodulation performance between the AWGN channel transmission simulations and the mobile channel transmission simulations. These factors are the carrier phase tracking process noised by thermal noise, the signal phase variations, and the signal power variations. Each factor is individually commented next.

The inclusion of thermal noise during the carrier phase tracking process for the mobile channel transmission simulations, instead of assuming an ideal carrier phase estimation as it is done for the AWGN channel transmission simulations, should farther affect the GPS L1C signal than the GALILEO E1 signal. The reason is that although GPS L1C signal provides a higher percentage of signal power to the pilot channel than GALILEO E1 signal, for the inspected values of C/N_0 at which both signals obtain the same BER values into the AWGN channel case, the pilot C/N_0 value is larger for GALILEO E1. Therefore, since both signals use the same PLL and since GPS L1C signal should need to farther increase the signal C/N_0 at the receiver antenna output in order to obtain the previous useful E_b/N_0 of the AWGN channel case than GALILEO E1 signal should, the difference should be reduced.

The signal phase variation introduced by the mobile channel transmission simulations should also decrease the difference of demodulation performance between the navigation signals. The same reason given in the previous paragraph can be used to justify this statement. GALILEO E1 signal provides a larger pilot channel C/N_0 value than GPS L1C signal does

when both signals obtain the same BER for the AWGN channel case. Therefore, since both signals use the same PLL, the tracking performance is farther worsened for the GPS L1C signal than for GALILEO E1 signal. And this means that GPS L1C needs a bigger increase of C/N_0 at the receiver antenna output than GALILEO E1 signal does in order to obtain the previous useful E_b/N_0 of the AWGN channel case.

Therefore, since the two previous factors should reduce the demodulation performance difference but, in reality, this difference is increased, it can be concluded that GPS L1C is much more adapted to the signal power variations than GALILEO E1 signal. The influence of the signal power variation onto the demodulation performance is given next.

As described in section 3.2.2.3, there are three types of signal power variations depending on their variation rate. Nevertheless, in this study, only the S1 state has been simulated which means that only the slow variations and the multipath variations are taken into account. The slow variations have an average correlation length equal to 2.42m which means that if the receiver travels at a speed of 30 km/h, this distance is covered in about 290ms. Therefore, for 290ms periods of time the signal power can be quite low which implies that the symbols received during this period will be very noisy and demodulated with error. Nevertheless, each signal has an interleaver in order to avoid this kind of burst of errors. The interleaver is different for each navigation signal and comparing the 290ms with the burst length supported by each interleaver (see Table 6-3), it can be seen that whereas GPS L1C signal can easily support bursts of this length, the GALILEO E1 interleaver is largely overwhelmed. This means that whereas GPS L1C can use its channel code decoding performance calculated by a transmission through an AWGN channel, GALILEO E1 cannot. The individual analysis for each signal is presented below.

On one hand, GPS L1C signal can exploit the channel code AWGN decoding performance since its interleaver can support the maximum burst of errors length. Therefore, the demodulation performance presented in Figure 6-4, Figure 6-5 and Figure 6-6 should be similar to the demodulation performance obtained for an AWGN channel transmission when the carrier phase tracking process is noised by thermal noise. Nevertheless, note that in addition to the thermal noise, the carrier phase tracking process for the analyzed mobile channel also has to track the signal phase variation introduced by the channel. Therefore, Figure 6-4, Figure 6-5 and Figure 6-6 performance should be worse or slightly worse than the AWGN performance case when the carrier phase tracking process is noised by thermal noise.

On the other hand, GALILEO E1 signal cannot break the 290ms burst of errors and thus, the channel code decoding performance calculated for the AWGN is no longer applicable. In fact, since the burst of errors cannot be totally broken, the channel code receives consecutive wrong bits which are very harmful for a convolutional code. Therefore, since the channel code is not able to correct consecutive wrong bits, the channel code loses part of its correction capacity. Moreover, in this situation, the only remaining form to avoid the presence of consecutive wrong bits is to increase the signal C/N_0 at the receiver antenna output. Nevertheless, this C/N_0 increase also decreases the total average number of errors present into a word. Therefore, in this situation, the channel code decoding performance curve begins to resemble the demodulation performance of a BPSK modulated signal without channel code. In fact, the increase of the number of consecutive bits errors provided to the channel convolutional code leads to the decrease of its correction capacity. Therefore, the curve of the decoding performance of a convolutional code resembles more the curve of the demodulation performance of an uncoded BPSK modulated message. This statement can be verified from Figure 6-4 where the GALILEO E1 signal curves are not as steep as the GALILEO E1 AWGN channel curve.

Finally, the multipath variations have not been taken into account in this analysis since a decrease of C/N_0 due to the fading lasts an average of 2.3 symbols for GPS L1C signal and an average of 5.7 symbols for GALILEO E1 OS signal. The interleaver of each signal can break the average burst of error associated to each signal.

The next figure illustrates the WER obtained by each navigation signal.

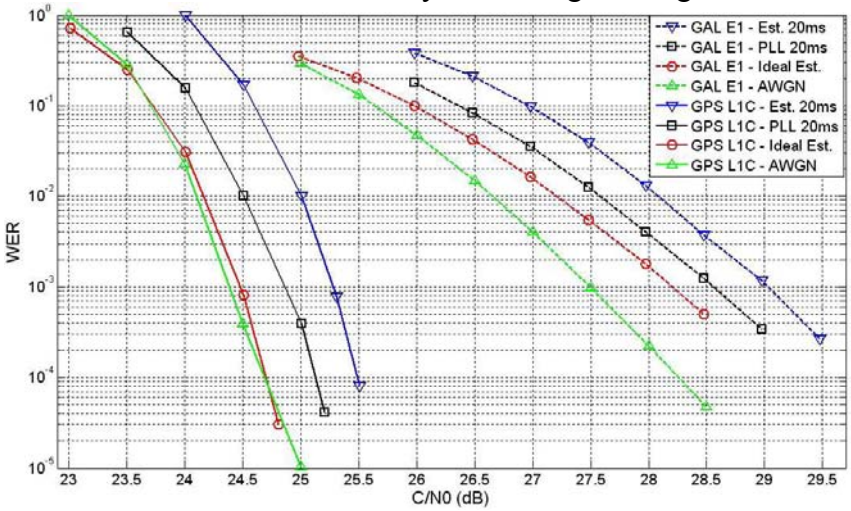

Figure 6-5: WER comparison between GPS L1C and GALILEO E1 OS signals for a mobile channel transmission with a receiver travelling at 30 km/h and the conditions specified in Table 6-4 and Table 6-5

From Figure 6-5, the same conclusions as for the BER can be extracted. First, the PLL demodulation performance for both signals continues to be better than the demodulation performance obtained for the proposed channel estimation method but this demodulation performance is worse than the ideal pilot channel estimation demodulation performance.

Second, GPS L1C signal outperforms GALILEO E1 signal in terms of WER as a function of the C/N_0 at the receiver antenna output for this modeled mobile channel. For example, when the signals phase tracking process is achieved with a PLL, there is a difference of 3.1 dB for a WER of 10^{-2} and a difference of 3.8 dB for 10^{-3}. The same degradation effect between the AWGN channel curves and the mobile channel curves is also observed and the reason is the same as before.

The next figure illustrates the EER obtained by each navigation signal.

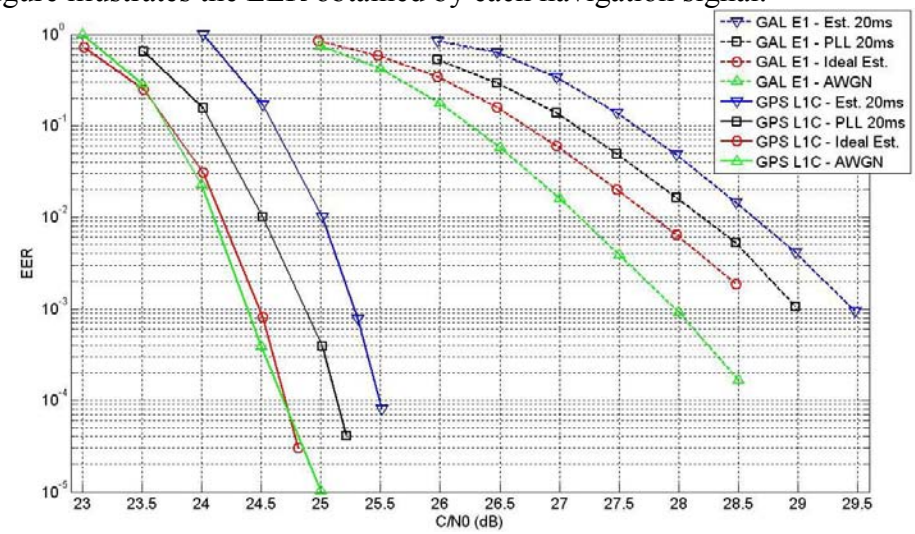

Figure 6-6: EER comparison between GPS L1C and GALILEO E1 OS signals for a mobile channel transmission with a receiver travelling at 30 km/h and the conditions specified in Table 6-4 and Table 6-5

In Figure 6-6, it can be observed that the gain difference between the GPS L1C signal and GALILEO E1 signal is equal to 3.7 dB for an EER equal to 10^{-2} and 4.2 dB for an EER equal to 10^{-3} when both signals conduct the carrier phase tracking process with a PLL. Moreover, all the previous observations are confirmed.

Additionally, it can be seen that the strategy of dividing the ephemeris set into 4 words also obtains worse demodulation performance than the strategy of transmitting the entire ephemeris data set in only 1 word. The reason is that the difference between GALILEO E1 signal and GPS L1C signal is larger for the EER figure of merit than for the WER figure of merit.

Finally, during all the simulations conducted in order to obtain the presented performance, both signals have obtained a PLL lock percentage larger than 99.9%. This means that when the signals are transmitted through a mobile channel and they are at the line-of-sight conditions state, the carrier phase tracking process can be done with the PLL and there is no real need to use a channel estimation technique.

6.4.3.3. Demodulation performance for a receiver speed of 5km/h at the LOS conditions state

In this section, the GALILEO E1 and GPS L1C signals demodulation performance is calculated for the case where the signals are received into a urban environment, when the user travels at a speed of 5 km/h and the carrier phase tracking process is noised by the presence of thermal noise.

Moreover, in the following graphics, in addition to showing the demodulation performance obtained for a mobile channel, the navigation signals demodulation performance obtained for an AWGN channel assuming ideal carrier phase tracking process is also plotted. Therefore, it is possible to analyze and to compare the demodulation performance degradation suffered by each individual navigation signal. The AWGN channel curves are adapted as explained in previous section 6.4.3.2.

The next figure illustrates the BER obtained by each signal.

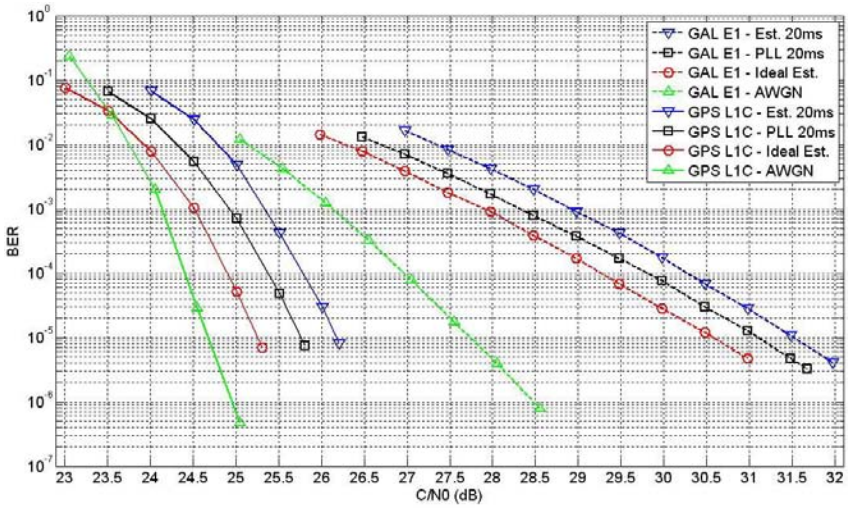

Figure 6-7: BER comparison between GPS L1C and GALILEO E1 OS signals for a mobile channel transmission with a receiver travelling at 5 km/h and the conditions specified in Table 6-4 and Table 6-5

The first thing that can be observed from Figure 6-7 is that, as well as in the 30km/h case, the difference of performance between the PLL and the proposed channel estimation method is small but favorable to the PLL method. In fact, the differences are almost the same as before with maybe a slight difference of 0.1 dB. Moreover, in this case, the ideal pilot channel estimation also outperforms the PLL.

The second observation that can be made from Figure 6-7 is that GPS L1C also outperforms GALILEO E1 OS in terms of BER into a mobile channel scenario when the receiver travels at 5 km/h. More specifically, GALILEO E1 needs 3.4 extra dB for a BER equal to 10^{-3} and 5.4 dB for 10^{-5} than GPS L1C signal does when both signals are received using a PLL. Moreover, it can be observed that this difference grows as the C/N_0 increases since the slope of the GPS L1C demodulation performance curve is steeper than the GALILEO E1 curve. In fact, it can be observed that this difference is significantly larger than in the case where the receivers travel at 30 km/h. The reason is the same as explained for the 30 km/h case.

In this case, when the receiver travels at a speed of 5 km/h, the slow variations have an average duration of 1.74 seconds. This means that none of the navigation signals interleavers can completely break a possible burst of errors caused by a low signal power period. Nevertheless, GPS L1C signal is much less affected than GALILEO E1 signal due to two main factors.

The first factor is the better GPS L1C interleaver which allows less consecutive bit errors than GALILEO E1 interleaver. On one hand, the GPS L1C interleaver inserts 45 bits between two received consecutive bits, which implies that at most 4 or 5 consecutives wrong bits can be found after the desinterleaving process, and that about 41 a priori correct bits are found between two consecutives bursts of errors. On the other hand, the GALILEO E1 interleaver only inserts 7 bits between two consecutive received bits, which means that for a 1.74 seconds burst of errors the interleaver have not effect at all. In fact, the burst of error length is longer than a GALILEO E1 signal word; therefore it is really quite impossible for any channel code to correct the word. Nevertheless, even for bursts of errors having a length of about 0.5 seconds, the received word would still have 4 or 5 correct bits followed by 4 or 5 wrong bits all along its length.

The second factor is the channel code implemented for each navigation signal. GPS L1C has an LDPC channel code which in addition to obtaining better decoding performance for an AWGN channel transmission than the convolutional code implemented for GALILEO E1, is also more robust to bursts of errors than any convolutional code.

To summarize, it can be seen from Figure 6-7 that GPS L1C signal begins to be affected by the slow speed of the receiver but stills obtains a satisfactory BER. However, it can also be seen that the GALILEO E1 OS signal channel code loses a large part of its correction capacity due to the implemented interleaver, due to the short duration of a GALILEO E1 OS word and due to its channel code.

The next figure illustrates the WER obtained by each signal.

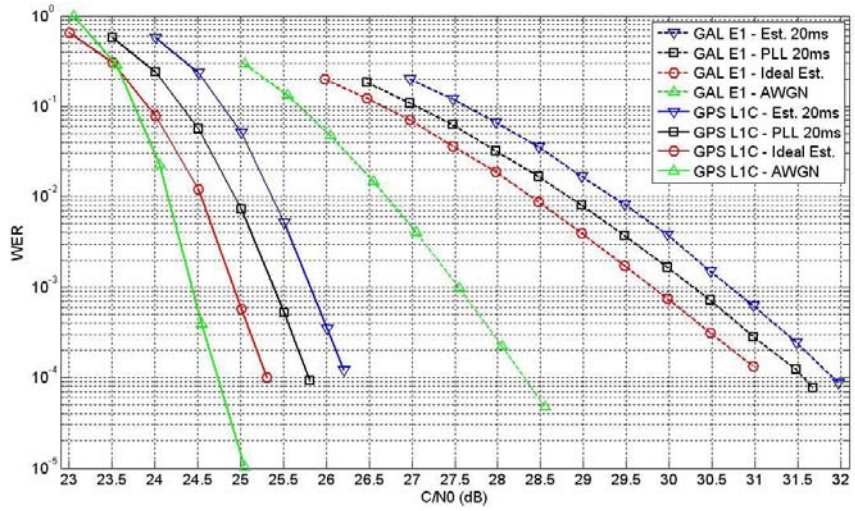

Figure 6-8: WER comparison between GPS L1C and GALILEO E1 OS signals for a mobile channel transmission with a receiver travelling at 5 km/h and the conditions specified in Table 6-4 and Table 6-5

From Figure 6-8, the same conclusions as for the BER can be drawn. First, the PLL demodulation performance for both signals is still better than the demodulation performance obtained for the proposed channel estimation method but worse than the demodulation performance of the ideal pilot channel estimation.

Second, GPS L1C outperforms GALILEO E1 OS in terms of WER as a function of the C/N_0 at the receiver antenna output for this mobile channel. For a WER of 10^{-3} there is a difference of 4.8 dB and for 10^{-4} a difference of 5.8 dB when the signals phase tracking process is achieved with a PLL. The same degradation effect between the AWGN channel curves and the mobile channel curves is also observed and the reason is the same as before.

The next figure illustrates the EER obtained by each navigation signal.

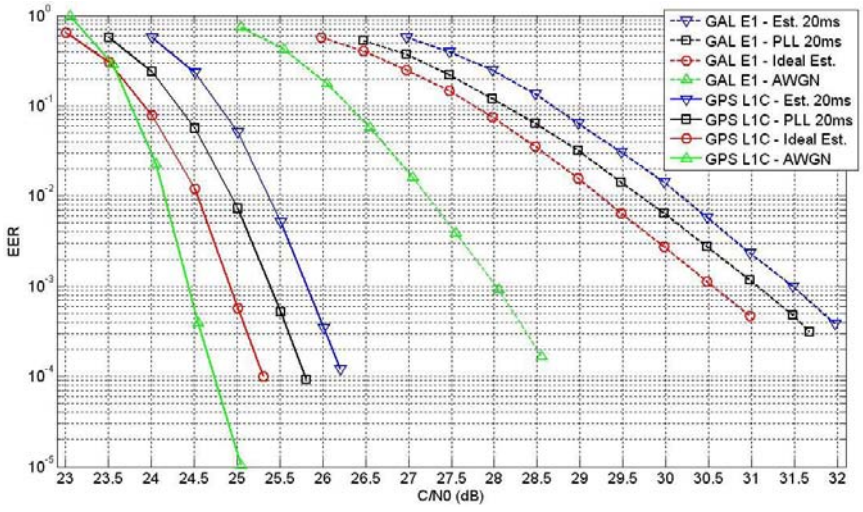

Figure 6-9: EER comparison between GPS L1C and GALILEO E1 OS signals for a mobile channel transmission with a receiver travelling at 5 km/h and the conditions specified in Table 6-4 and Table 6-5

In Figure 6-9, it can be observed that the difference between the GPS L1C signal and GALILEO E1 signal is equal to 4.8 dB for an EER equal to 10^{-2} and 5.7 dB for an EER equal

to 10^{-3} when both signals conduct the carrier phase tracking with a PLL. Moreover, all the previous observations are confirmed.

Additionally, as well as it was confirmed for the 30km/h case, it can be seen that the strategy of dividing the ephemeris data set into 4 words also obtains worse demodulation performance than the strategy of transmitting the entire ephemeris data set in only 1 word. Indeed, the difference between GALILEO E1 OS signal and GPS L1C signal is larger for the EER figure of merit than for the WER figure of merit.

Finally, during all the simulations conducted in order to obtain the presented performance, both signals have obtained a PLL lock percentage larger than 99.9%. This means that when the signals are transmitted through a mobile channel and they are at the line-of-sight conditions state, the carrier phase tracking process can be done with the PLL and there is no real need to use a channel estimation technique (due to the small difference performance).

6.4.3.4. GPS L1C demodulation performance for a receiver speed of 30km/h for the 3 states model

In this section, the GPS L1C signal demodulation performance is calculated for the case where the signal is received into a urban environment, when the user travels at a speed of 30 km/h and the carrier phase tracking process is made by a PLL. The only source of error considered in this analysis which corrupts the carrier phase tracking of the incoming signal is the thermal noise.

In this section, only the BER and the WER are shown since GPS L1C signal broadcasts its ephemeris data sets inside 1 only word. Moreover, three different performance types are shown depending on the bits and words used to calculate these figures of merit.

In the first case, all the received bits are used even if the receiver PLL is not locked. This means that this performance shows the BER and WER which can be obtained for a continuous broadcasting of information which is not repeated in time. This could be the case of future services.

In the second case, only the bits and words received during the line-of-sight state and the intermediate shadows state are used even if the receiver PLL is not locked. In fact, when the receiver is in the deep shadows state the simulation directly determines that the PLL cannot be locked.

In the third case, the same bits and words as the bits and words of the second case are used but only when the PLL is locked and the transient period has passed.

6.4.3.4.1. 3 states model demodulation performance – 1st Case

In this section, the demodulation performance of the GPS L1C ephemeris data is presented. The calculation of this performance takes into account all the received bits and words even when the received PLL is not locked or the PLL is in the transient zone.

The performance figures presented in the figure below is the BER and WER when only one word is used to demodulate the ephemeris data set, and the BER and the WER when two consecutives words are used to demodulate the ephemeris data set.

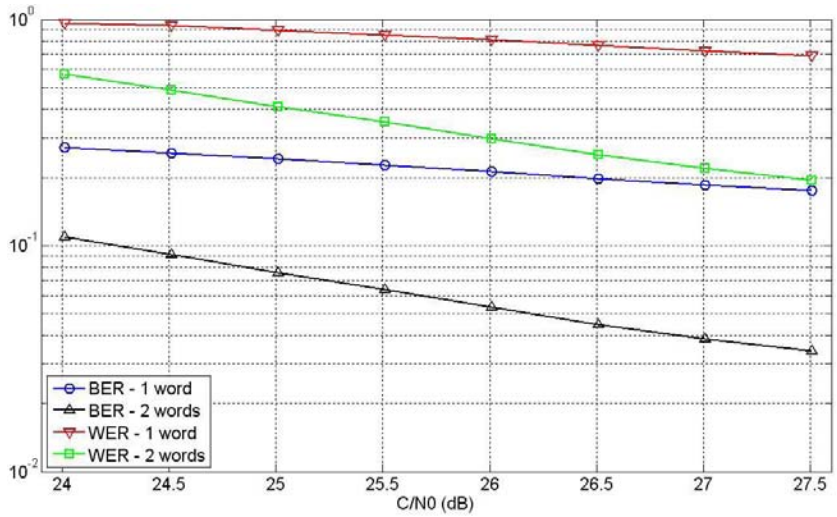

Figure 6-10: GPS L1C 1st situation demodulation performance for a mobile channel transmission modelled with a 3 state mathematical model with a receiver travelling at 30 km/h

From Figure 6-10, it can be observed that the BER and the WER obtained when either 1 or 2 words are used to demodulate the signal is not satisfactory. Moreover, it can be seen that the curves have a very mild slope, which means that even for higher C/N_0 values, no satisfactory BER and WER values can be obtained.

The main reason for this behavior is the fact that the deep shadows state (S3) does not allow the correct reception of the transmitted bits. More specifically, when the PLL is not locked, the value of the received bits is assumed equiprobable. Therefore, these bits are considered as erasures from the point of view of the channel code. To sum up, only the words containing a high percentage of bits received when the PLL is locked can be corrected.

Finally, in order to analyze how the S3 state affects the demodulation performance, the PLL lock percentage of time is presented in the figure below. Moreover, in addition to plotting the total percentage of time during which the PLL is locked, the PLL lock percentage of time taking into account only the time when the receiver dwells into the S1 and S2 states is also displayed. In other words, two PLL locks percentages of time are presented. The first one is the PLL lock percentage when the receiver travels the three states, S1, S2 and S3, defined by the mathematical model presented in section 3.2.2.3.5. The second one is the PLL lock percentage when the receiver is travelling the S1 and S2 states.

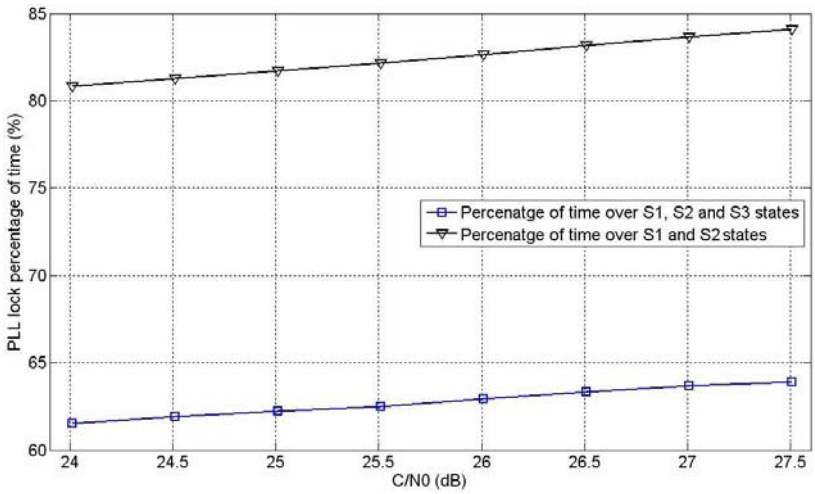

Figure 6-11: GPS L1C PLL lock percentage of time when the signal is transmitted through a mobile channel and the receiver travels at 30 km/h

From Figure 6-11, it can be seen that the total percentage of time that the PLL is locked is lower than 65%. It can also be seen that more than half of the time during which the PLL is not locked is when the receiver dwells in the S3 state. In fact, comparing the blue and black curves, it can be seen that they have about the same slope but with a different positive offset. This offset should be equal to the percentage of time that the receiver dwells in the S3 state in relation to the total time of the simulation.

Therefore, from Figure 6-10 and Figure 6-11, it can be concluded that due to the S3 high attenuation levels, the total BER and WER values obtained during the transmission of the GPS L1C signal are not satisfactory. Nevertheless, this fact does not mean that the positioning service cannot be provided due to this unsatisfactory performance. In fact, the BER and WER values calculated in this section represent the demodulation performance of a continuous stream of information whereas the navigation signal only requires that the receiver is able to demodulate the ephemeris data set once every 2 hours. Nevertheless, it can be seen from Figure 6-11 that the positioning service is at risk since 35% of the time the receiver cannot track the signal.

6.4.3.4.2. 3 states model demodulation performance – 2nd Case

In this section, the demodulation performance of the GPS L1C ephemeris data is presented. The calculation of this performance takes into account all the bits and words received when the user dwells in the S1 and S2 states. Moreover, note that this case is quite equivalent to the case where the demodulation performance is calculated for words containing at least some bits received when the PLL is locked.

The performances values presented in the figure below are the BER and WER when only one word is used to demodulate the ephemeris data set, and the BER and the WER when two consecutives words are used to demodulate the ephemeris data set.

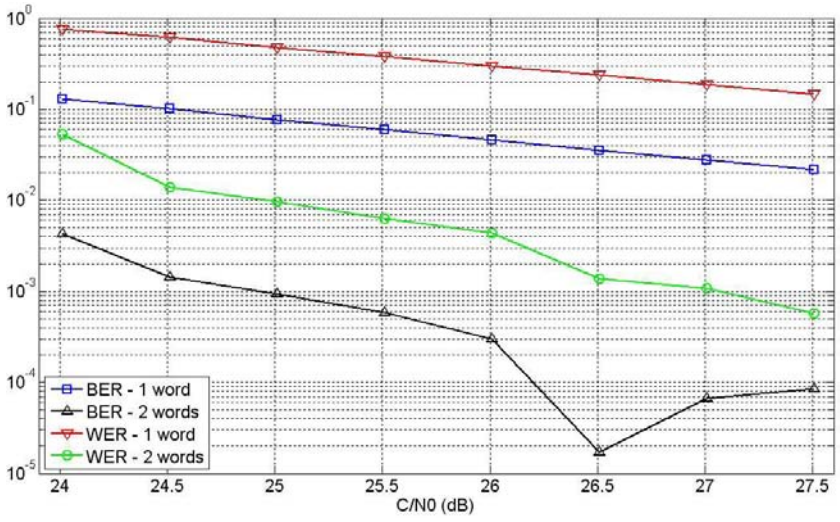

Figure 6-12: GPS L1C 2nd situation demodulation performance for a mobile channel transmission modelled with a 3 states mathematical model with a receiver travelling at 30 km/h

The first observation that can be made from Figure 6-12 is that the BER and WER obtained when only 1 word is used to demodulate the navigation message are not satisfactory. In fact, although these values are better than the values obtained for the first situation, they are still far from the desired 10^{-5} for the BER and the desired 10^{-2} and 10^{-3} for the WER.

This unsatisfactory performance can also be explained by the percentage of time that the PLL is not locked when the receiver dwells either in the S1 state or in the S2 state. From Figure 6-11, it can be seen that 15% of the time the bits cannot be received because the PLL is not locked. Nevertheless, this time seems a little larger taking into account that when the receiver was in the S1 state, the PLL was locked more than 99.9% of the time, see section 6.4.3.2. Therefore, it can be concluded that the S2 state is responsible for almost all the time where the PLL is not locked. However, it has to be noted that this simulation has not taken into account the transitions between states and has imposed that the signal amplitude between states has to be continuous. Therefore, the calculation of the PLL lock percentage uses signal amplitude samples having low amplitudes because they correspond to the transition between the S3 state and either the S1 or S2 state. And these samples are considered to belong to the latter states although their amplitude is too low to be representative of the S1 or S2 state.

The second observation that can be made from Figure 6-12 is that the accumulation of two consecutives words improves the demodulation performance. Nevertheless, this improvement is not as good as the improvement found for the AWGN channel case and the cause is still the bits received during the PLL loss of lock. However, the strategy of coherent word accumulation can be improved with respect to the simple strategy which performance is presented in Figure 6-12 by coherently accumulating all the bits which should be equal, even if they are separated farther apart than 2 consecutive words.

More specifically, the implemented strategy of coherent word accumulation of Figure 6-12 consists in coherently accumulating two consecutive words which carry the same ephemeris data. This means that the simulation does not search to accumulate all the words which carry the same ephemeris data, i.e. the words transmitted inside a period of 15 minutes. Therefore, in order to improve the strategy of the coherent word accumulation of Figure 6-12, the option where all words transmitted inside a 15 minutes period are accumulated has to be inspected. In fact, this strategy could be further improved by coherently accumulating all the bits which should be equal to other bits since they represent the same ephemeris data.

Finally, the number of two consecutive words which are received either at S1 state or at S2 state even when the receiver PLL is not locked is very small. In fact, the relationship between the number of pair of words having the previous characteristics and the total number of accumulated words groups is about 7%, which means that the demodulation performance is calculated without too many points and thus the obtained values are subject to possible biases. This phenomenon can be observed in the last 4 points of the WER and BER curves.

6.4.3.4.3. 3 states model demodulation performance – 3rd Case

In this section, the GPS L1C ephemeris data demodulation performance is presented. The calculation of this performance only takes into account the bits and words received when the PLL is locked.

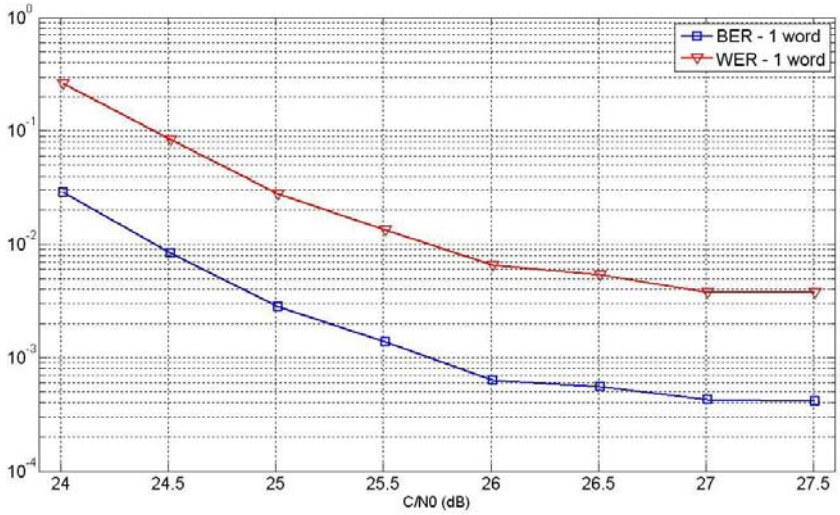

Figure 6-13: GPS L1C 3rd situation demodulation performance for a mobile channel transmission modelled with a 3 states mathematical model with a receiver travelling at 30 km/h

From Figure 6-13, it can be observed that although the demodulation performance obtained when only one word is used to demodulate the signal is better than the previous performance calculated for the first two situations, this demodulation performance is still not satisfactory. In fact, it can be seen that the curves lose their descending slope as the C/N_0 increases at the antenna output, until it seems that the demodulation performance reaches a lower bound.

This result is a priori surprising since it was expected to obtain the same demodulation performance as in Figure 6-4 and Figure 6-5. Nevertheless, in this case, the PLL can achieve a lock when the receiver dwells in the S2 state. This means that whereas in Figure 6-4 and Figure 6-5 case, the C/N_0 value before taking into account the signal Loo distribution amplitude modification is very similar to the C/N_0 after taking into account the signal Loo distribution amplitude modification, in this case, the latter value can be quite low in relation to the former value. Therefore, the demodulation performance presented in Figure 6-13 is the result of worsening Figure 6-4 and Figure 6-5 demodulation performance with the S2 demodulation performance influence. Moreover, the influence of the transition between the S3 state and the S1 or S2 states also decreases the final demodulation performance. However, this effect should not be very important since we are only taking into account the bits of the words received when the PLL is locked.

Moreover, it can be observed from Figure 6-13 that the curves for the 2 consecutive words accumulation strategy case are not displayed. In fact, no such curve is plotted because no error has been found for any of the inspected values. Therefore, it can be concluded that the gain obtained from the accumulation strategy is the same as for the AWGN channel case. Nevertheless, this simulation only accumulates two consecutives words, which results into a very limited use of this strategy. More specifically, the percentage of the groups of accumulated words which are received either at S1 state or at S2 state when the PLL is locked with respect to the total number of words transmitted during all the simulation is about 0.1%. Therefore, much longer simulations should be made in order to obtain more reliable results.

To summarize, the demodulation performance calculated by taking into account only the bits and words received when the PLL is locked is still not satisfactory. Besides, the demodulation performance reaches a lower bound at $4 \cdot 10^{-4}$ for the BER and at $4 \cdot 10^{-3}$ for the WER due to the bad performance obtained when the receiver dwells into the intermediate shadows state. Moreover, this performance cannot be used the majority of the time since only 4% of the total

number of simulated words are completely received either at S1 or S2 state when the PLL is locked.

Finally, it can be concluded that in order to obtain a satisfactory demodulation performance, the receiver must use the word accumulation strategy enlarged to any number of words or even bits transmitted inside the same period of 15 minutes. Remember that inside this period the words are equal.

6.4.4. Table summarizing the results

In this section, a table summarizing the results of sections 6.4.2, 6.4.3.2 and 6.4.3.3 is presented. Moreover, only the results obtained when employing a PLL to track the signal carrier phase are shown in the table below.

Channel		AWGN			Mobile 30 km/h			Mobile 5 km/h		
Figure of Merit		C/N_0 (dB-Hz)		G.D. (dB)	C/N_0 (dB-Hz)		G.D. (dB)	C/N_0 (dB-Hz)		G.D. (dB)
		GPS	GAL		GPS	GAL		GPS	GAL	
BER	10^{-3}	24.6	26.6	2	24.5	27	2.5	24.9	28.3	3.4
	10^{-5}	25.1	28.2	3.1	25.1	28.8	3.7	25.7	31.1	5.4
	10^{-6}	25.4	28.9	3.5	--	--	--	--	--	--
WER	10^{-2}	23.6	27.1	2.5	24.5	27.6	3.1	24.9	28.8	3.9
	10^{-3}	24.9	28	3.1	24.8	28.6	3.8	25.4	30.2	4.8
	10^{-4}	25.2	28.7	3.7	25.2	--	--	25.8	31.6	5.8
EER	10^{-2}	23.6	24.7	3.1	24.5	28.2	3.7	24.9	29.7	4.8
	10^{-3}	23.9	28.5	3.6	24.8	29	4.2	25.4	31.1	5.7

Where:

- G.D. = C/N_0 difference between the GALILEO E1 OS and GPS L1C signal

Remember that the mobile channel results have a power offset of 0.45 dB with respect to the AWGN channel since the average of the log-normal variable is equal to 0.45 dB for the chosen scenario. See section 6.4.3.1.

6.5. Conclusions

In this chapter, the design of the GPS L1C signal and the GALILEO E1 OS signal has been tested and compared in order to analyze which design provides a better ephemeris data demodulation performance. The analyses were conducted in theoretical and experimental manners, and several conclusions were extracted, all of them leading to the same overall conclusion: GPS L1C signal is better suited to ephemeris data demodulation performance than the GALILEO E1 signal. The reason is that GPS L1C requires a smaller signal C/N_0 value at the receiver antenna output than GALILEO E1 signal does in order to obtain the same BER, WER or EER value in either open or urban environments. All the conclusions used to obtain this general conclusion and some other conclusions also extracted in this chapter are presented next.

From the theoretical point of view, the design characteristics which condition the ephemeris data demodulation performance are the signal channel relative power distribution, the symbol transmission rate, the data message structure, the interleaver and the data message content. And, in this chapter, for this performance it was shown that all of them are more optimized for GPS L1C signal than for GALILEO E1 signal. The reason is that GALILEO E1 OS signal also provides additional services than GPS L1C signal and thus these key characteristics are conditioned by this choice of design.

More specifically, the signal channel relative power distribution and the symbol transmission rate are responsible for providing at least 1 extra dB in any kind of environment to the useful GPS L1C message E_b/N_0 compared to the GALILEO E1 useful message E_b/N_0 signal when both signals have the same C/N_0 value at the receiver antenna output. Moreover, this difference can be enlarged because GPS L1C signal provides more percentage of power to the pilot channel and thus the carrier phase tracking performance is better for GPS L1C signal. And this means that since the tracking performance also affects the useful message E_b/N_0 value, the difference of 1 dB is enlarged when both signals have the same C/N_0 value at the receiver antenna output.

Nevertheless, the 1 dB difference is mainly caused by the different symbol transmission rates selected for each signal. The symbol transmission rate is 2.5 times faster for GALILEO E1 OS and thus, a simple reduction of this rate could balance the performance. However, one of the most important high level choices made during the conception of GALILEO E1 signal consists in providing SoL and integrity services in addition to the typical open positioning service. This means that in order to transmit all the additional services, GALILEO E1 OS signal needs a higher information transmission rate than GPS L1C does. Therefore, this high symbol transmission rate is selected knowing perfectly well the consequences over the demodulation performance.

The other three remaining factors which condition the demodulation performance increase farther the 1dB difference between the signals. First, the data message content allows a perfect word accumulation for GPS L1C whereas the accumulation only affects a little more than 55% of the ephemeris data bits for GALILEO E1 OS. Besides, the correct word accumulation process cannot be guaranteed for GALILEO E1 OS signal. Second, the GPS L1C data message structure provides a more powerful and more robust channel code to the burst of errors than the GALILEO E1 OS data message structure does. Moreover, the division of the GALIEO E1 ephemeris data into 4 words instead of broadcasting the ephemeris data set in only 1 word is not a priori a good choice since it makes the complete reception of the ephemeris data set harder to achieve. Third and last, the GPS L1C interleaver is able to break bursts of errors having a length 3 times longer than the bursts of errors length that the GALILEO E1 interleaver can break.

All the previous presented conclusions summarize the results obtained from the theoretical studies. Nevertheless, all these theoretical results had to be verified and quantified through simulations. These simulations represent open environments by means of a simulated AWGN channel and urban environments by means of a mobile channel. Therefore, the previous theoretical conclusions are completed with the experimental conclusions presented below.

The demodulation performance obtained for a signal transmission through an AWGN channel when assuming an ideal carrier phase tracking process confirms that GPS L1C signal outperforms GALILEO E1 OS signal when they are received into an open environment. More specifically, GALILEO E1 OS signal needs 3.2 extra dB than GPS L1C signal does to obtain the same BER of 10^{-5}, 3.1 dB for a WER of 10^{-3} and 3.6 dB for an EER of 10^{-3}. And these differences grow as the C/N_0 increases. Note that this difference is mainly due to the 1 extra

dB obtained by GPS L1C and due to the more powerful channel code implemented for this signal.

Moreover, from the AWGN channel simulations it is confirmed that the accumulation strategy works much better for GPS L1C signal. Indeed, the GALILEO E1 OS signal obtains worse WER and EER figures of merit when 2 words are accumulated to demodulate the message, than GPS L1C signal when only 1 word is used.

The results obtained for a signal transmission through a mobile channel when assuming a carrier phase tracking process noised by thermal noise also confirms the better ephemeris data demodulation performance obtained by GPS L1C signal. And, besides, these results present larger demodulation performance differences than the differences obtained for the AWGN channel case due to the better interleaver implemented for GPS L1C signal.

However, these simulations conducted to compare the demodulation performance of both signals really analyze the best demodulation performance which can be obtained at a punctual moment for each navigation signal in the simulated scenario. In fact, for these simulations, the receiver is assumed to always dwell in the line-of-sight state, the state where the attenuation parameters are the smallest ones. Nevertheless, the satellite elevation angle chosen for the simulated scenario is quite low, 30^{o}, when, in reality, the receivers should expect to acquire, track and demodulate the signal with satellites having higher elevation angles. Therefore, it can be stated that the results calculated for these simulations present the best demodulation performance which can be obtained by a signal in one of the worse but realistic scenarios. And this means that the conclusions extracted from these results are valid under these conditions.

The conclusions from these last simulations were customized for each different mobile velocity case, since the bursts of errors and the signal power variations depend on the receiver motion. Therefore, two speeds were analyzed in this chapter, a car travelling at 30 km/h and a pedestrian walking at 5 km/h. Nevertheless, in both cases the demodulation performance differences between the GPS L1C signal and the GALILEO E1 OS signal are increased. The reason in both cases is the better interleaver implemented for GPS L1C signal. A detailed conclusion is presented below for each velocity case.

For a receiver moving at 30 km/h, the GPS L1C interleaver is able to break the great majority of bursts of errors introduced by the mobile channel. Therefore, the demodulation performance obtained by this signal is very similar to the AWGN case performance. More specifically, the only difference is an offset between the demodulation performance curves since the curves have the same slope. This offset is caused by the losses of E_b/N_0 introduced by the tracking performance. However, the GALILEO E1 OS interleaver for a receiver moving at 30 km/h is only able to break very few bursts of errors introduced by the mobile channel. Therefore, since the convolutional channel code implemented by this signal is very sensitive to the bursts of errors, the demodulation performance is degraded compared to the AWGN case. In fact, in this case, the difference with the AWGN case is an offset and a different slope of the demodulation performance, which is steeper for the AWGN case. Therefore, the differences between the signals obtained for the presented scenario when the receiver travels at a speed of 30 km/h are larger than the difference for the AWGN case. Some representative values are 3.7 dB for a BER of 10^{-5}, 3.8 dB for a WER of 10^{-3} and 4.2 for an EER of 10^{-3}. And, as well as in the AWGN case, these differences grow as the C/N_0 increases.

For a receiver moving at 5 km/h, neither the GPS L1C interleaver nor the GALILEO E1 OS interleaver can break the great majority of bursts of errors introduced by the mobile channel. Nevertheless, whereas GPS L1C interleaver is able to reduce the bursts of errors to a maximal

length of 4 or 5 bits, the GALILEO E1 OS interleaver cannot reduced at all the effect of the burst. Moreover, the short size of the GALILEO E1 OS signal words make that some burst of errors are longer than the word itself, and thus it is impossible to recover the word for any existing channel code since the interleaver is only applied over individual words. Therefore, the difference in demodulation performance between the signals is increased compared to the AWGN case and to the 30 km/h case. Moreover, this difference is farther increased by the fact that the LDPC code implemented for the GPS L1C signal is much more robust to the burst of errors than the convolutional code implemented for GALILEO E1 OS signal. Some representative values are 5.4 dB for a BER of 10^{-5}, 4.8 dB for a WER of 10^{-3} and 5.7 for an EER of 10^{-3}. And, as well as in the AWGN case, these differences grow as the C/N_0 increases.

Additionally, from the previous mobile channel results and the AWGN channel results, it is confirmed that the strategy of dividing the transmission of the ephemeris data set into 4 words does not work either into an AWGN channel or into a mobile channel. Indeed, the difference between the signals obtained when searching the same WER and EER values is increased.

In a previous paragraph, it was commented that the demodulation performance results presented before were the best punctual results that each signal could expect to obtain in the simulated scenario. Nevertheless, the average ephemeris data demodulation performance obtained by the GPS L1C signal was also analyzed in this chapter. The extracted conclusions are presented below.

First, when the GPS L1C signal is received with a C/N_0 value equal to or smaller than 27.5 dB-Hz, the PLL is locked only 65% of the time. This means that the GPS L1C signal cannot provide positioning service during all the journey of a user through a urban environment when the received satellite has an elevation angle of 30^o. Moreover, the demodulation performance obtained when using any received word at any moment of the communication is not satisfactory, which means that the user cannot expect to recover the ephemeris information when the PLL is not locked. Besides, this means that no constant stream of information can be received by the user.

Second, the demodulation performance obtained using words which only contain bits received when the PLL is locked is still not satisfactory. In fact, since the PLL can be locked in situations where the C/N_0 is lower than the optimal situation of C/N_0 presented in the previous paragraphs, the final demodulation performance is degraded. Therefore, in order to recover the ephemeris data set with the desired demodulation performance, the user must wait to receive the ephemeris data set at a punctual moment where the optimal conditions are present. Nevertheless, these moments are very scarce since only 4% of the total received words have been completely received with the PLL being locked. Therefore, the only practical solution to obtain the desired performance is to use the accumulation strategy. More specifically, the receiver must only accumulate identical words or identical bits of the words received when the PLL is locked.

Finally, from all the mobile channel simulations conducted when the signal only dwells in the LOS conditions state, it can be concluded that the PLL method obtains satisfactory performance to be used as the main technique to estimate the received signal carrier phase. Indeed, the implementation of a channel estimation method to estimate the signal carrier phase does not provide a significant gain compared to the PLL method. More specifically, the ideal pilot channel estimation only requires about 0.4 dB less than the PLL to obtain the same BER, WER or EER value. Therefore, only if a channel estimation technique is able to allow a coherent demodulation with a satisfying performance the majority of the time where the PLL cannot achieve a lock, the channel estimation method should be used instead of the PLL.

Chapter 7. Code Shift Keying implementation for GALILEO E1 signal

The European satellite navigation system, also known as GALILEO, had as one of its main objectives to provide free positioning service to civilian users. In order to attain this objective, two signals were developed, the GALILEO E1 signal and the GALILEO E5b signal. More specifically, it was the GALILEO E1 signal which targeted the civilian market whereas the GALILEO E5b signal was mainly thought to be used for civil aviation applications. Moreover, the GALILEO E5b signal was also conceived as a complement to the GALILEO E1 signal for receivers accepting dual frequency.

A first definition of the GALILEO E1 signal has already been made and its description can be found in the document [ESA, 2008]. Nevertheless, current studies are being conducted in order to analyse and to identify future opportunities of optimization of the future GNSS signals structures. In fact, some organizations such as the ESA (European Space Agency), CNES (the National French Space Agency) or THALES ALENIA SPACE are conducting studies in that sense.

Some of these studies target the PRN codes and other ones, such as the analysis made in this dissertation, inspect the current navigation message. More specifically, two main navigation message aspects are analysed in this research work. First, the improvement of the current demodulation performance of the GALILEO E1 signal is sought. Or in other words, this study searches future structures which decrease the current C/N_0 level necessary to correctly demodulate the GALILEO E1 navigation message and, specially, the ephemeris data. Second, the increase of the signal information transmission capacity is sought in order to allow future GNSS signals to provide extra services in addition to the existing ones: positioning, integrity and safety of life (SoL) for GALILEO E1 signal for example. The following paragraphs provide a first brief explanation of the solutions proposed in this research work.

On one hand, a future navigation message having an improved demodulation performance with respect to the current navigation message demodulation performance of GALILEO E1 OS was proposed by THALES ALENIA SPACE. This proposal consists of changing the current navigation message structure by a structure similar to the one implemented by GPS L1C signal. Since this new structure has not been developed by this research work, the increase of the demodulation performance with respect to the original GALILEO E1 signal is not commented in this dissertation. The description of the new proposed message can be found in annex D.6.

On the other hand, in order to improve the signal information transmission capacity or information rate, the implementation of a technique known as Code Shift Keying or CSK is analysed. The CSK consists of shifting the PRN code associate to the data channel, and interpreting the code shift as a transmitted group of bits. Therefore, for example, since for GALILEO E1 signal, a PRN code period spans 1 symbol length, the CSK implementation increases the information rate by a factor equal to the number of bits represented by a PRN code shift. And thus a possible future GNSS signal structure implementing the CSK technique will have a higher information transmission capacity than a current GNSS signal having the same symbol duration but not implementing the CSK technique.

To sum up, in this chapter, the Code Shift Keying modulation technique is described and its possible implementation on a future GALILEO E1 signal proposed by TAS-France is presented.

This chapter begins by presenting the definition of the Code Shift Keying technique, its mathematical model and the modulator and demodulator block schemes. This section continues by presenting the CSK theoretical BER expressions for a hard demodulation output. Afterwards, the mathematical expression of the likelihood ratio of a bit transmitted inside a CSK symbol is calculated, and the soft demodulation performance obtained from the application of the bits likelihood ratios is compared to previously presented hard demodulation performance.

The interest of using the sign or polarity of the CSK symbols in order to transmit an extra bit is also analyzed and compared to the demodulation performance obtained when this polarity is not used. Besides, an optimal mapping between the bits belonging to the same code word and the number of these bits transported inside the same CSK symbol is presented. The BER as a function of the signal C/N_0 at the receiver antenna output is also shown for the TAS proposal of a future GALILEO E1 signal when the CSK technique is applied on its subframe 4 and none channel code is implemented. And the same results are presented when the channel code of the subframe 2 of GPS L1C signal has been implemented on the CSK transmitted bits.

Finally, the advantages and the drawbacks of the CSK signaling technique on a GNSS signal are explained in this section. The drawbacks are due to the lack of synchronization between the data channel PRN code and the pilot channel PRN code. This implies that the data channel can no longer be used to acquire or to track the signal in the signal parts where the CSK is implemented. Therefore, the acquisition performance of the TAS proposal of a possible future GALILEO E1 signal implementing the CSK technique is analysed and compared to acquisition performance of the current GALILEO E1 signal when the CSK technique is not implemented. Moreover, the optimal mapping of the CSK symbols which reduce the acquisition false alarm art when the data channel is employed to acquire the signal is also presented. Some initial conclusions over the impact on the tracking performance are explained. Besides, the impact of the implementation of the CSK technique on a GNSS signal over the pseudo-range calculation is presented.

One last remark to make about the CSK signalling technique is that this technique is planned to be implemented in the LEX signal of the Japanese global positioning service, QZSS, as indicated in the public released interface specification [JAXA, 2010].

7.1. Code Shift Keying definition

The Code Shift keying is a direct-sequence spread-spectrum (DS-SS) signaling method which overcomes the spreading gain versus data rate limitations [WONG and LEUNG, 1997]. In other words, the CSK was developed in order to increase the transmission efficiency of spread spectrums systems [HAMMER and SCHAEFER, 1982].

More specifically, the CSK is a form of M-ary orthogonal signaling over a communication channel [ENDSLEY and DEAN, 1994] since M orthogonal signaling waveforms are used in order to transmit $k = \log_2(M)$ bits. The special characteristic of the CSK modulation with respect to the typical M-ary orthogonal signaling is that each waveform is obtained from a different circular phase shift of a single maximal length PRN code; where each circular phase shift is made by an integer number of chips [WONG and LEUNG, 1997]. In other words, the

CSK modulation generally adopts a maximal-length sequence (m-sequence) with its period equal to M as its fundamental sequence. And each set of input data bits is represented by a signaling code, where each signaling code is obtained by cyclically shifting the adopted fundamental sequence. Each cyclically shifted sequence is assumed to be a full period version of the fundamental sequence [TSAI, 2009].

Additionally, in its simplest form and from the mathematical point of view, the CSK is defined by a chosen base function f(t). Cyclically (circularly) shifted versions of f(t) are used to modulate a carrier and to represent an input set of bits. And this function f(t) has the property that its cyclic autocorrelation has a distinct peak and "low" side-lobes [DILLARD et al., 2003].

In reception, if chip synchronization is assumed, the receiver can choose two possible types of demodulation schemes. In fact, these two demodulations schemes are the same for the CSK signaling technique; however, the second scheme cannot be applied for the general M-ary orthogonal signaling (MOS) technique. The first option consists in correlating the received signal at the RF/IF block output with each of the orthogonal functions representing a symbol. The correlation with the highest amplitude value determines the transmitted symbol [DILLARD et al., 2003].

The second option simplifies the signal processing of the first option. This option consists in conducting the correlation in the frequency domain. First, the Fourier transform of the received signal is computed. Second, the inverse Fourier transform of the product of the received signal Fourier transform and the complex conjugate of the base function Fourier transform is calculated. Third and last, the point with the largest amplitude of the inverse Fourier transform is identified. This point determines the transmitted circular code shift of the fundamental code by measuring the distance from the point to the origin [DILLARD et al., 2003]. In both options, once the transmitted shift code version has been identified, the receiver provides to higher layers the bits assigned to this shift.

Finally, in this chapter, the CSK method inspired from the M-ary orthogonal signaling is referred as orthogonal CSK. The orthogonal CSK is thus the CSK method where only one polarity –or sign– of the fundamental code is used to map the bits. Moreover, in this chapter, the CSK method inspired from the M-ary bi-orthogonal signaling is referred as bi-orthogonal CSK. The orthogonal CSK is thus the CSK method where both polarities –positive and negative sign- of the fundamental code are used to map the bits.

7.2. CSK mathematical characterization

In this section, the CSK mathematical characterization used along this chapter is first presented. Second, the mathematical modulation and demodulation schemes are shown.

Each single CSK symbol modulates k bits using the CSK technique. The number of circularly shifted versions of the fundamental code is equal to M, where $M = 2^k$. The CSK fundamental code is called $c_d(t)$ and has a period length equal to T which spans over L chips. L is not necessarily equal to M and the chip interval is equal to T_c. From this fundamental code $c_d(t)$, the modulator generates the M circularly shifted versions of the code which are called $c_d(t)$ to $c_{d-M+1}(t)$. Each symbol is represented by different k bits, and the set of symbols generates a M-ary alphabet. Moreover, each circularly shifted version of the fundamental code is also called waveform. A mathematical expression of a generic circularly shifted version of the code is shown below:

$$c_{d-x}(t) = c_d\left(\left[t - m_x \cdot T_c\right]_{\mathrm{mod}(L \cdot Tc)}\right) \quad x = 0..M-1 \tag{7-1}$$

$$c_{d-x}[k] = c_d\left[\left[k - m_x\right|_{\mathrm{mod}\,L}\right] \tag{7-2}$$

Where:

- m_x: Integer number representing the code shift of the x^{th} waveform.

Once the waveform has been selected, it is modulated at a carrier frequency ω_0, and after its transmission through an AWGN channel the received signal at the receiver antenna output can be modeled as:

$$v(t) = s(t) + n(t) \tag{7-3}$$

$$s(t) = A \cdot c_{d-x}(t) \cdot \cos(\omega_0 t) \tag{7-4}$$

Where

- v(t): Received signal at the receiver antenna output
- s(t): Transmitted signal at the transmitter antenna input
- n(t): Additive white Gaussian noise with power equal to σ^2.

7.3. CSK modulator and demodulator block schemes

In this section, the modulator and demodulator block schemes are presented. Moreover, the mathematical model of the demodulator outputs is given. Note that this mathematical model is used to decide which CSK symbol has been transmitted or to calculate the likelihood ratio of the bits transmitted inside a CSK symbol.

The modulator scheme of the transmitted signal can be modeled as shown below:

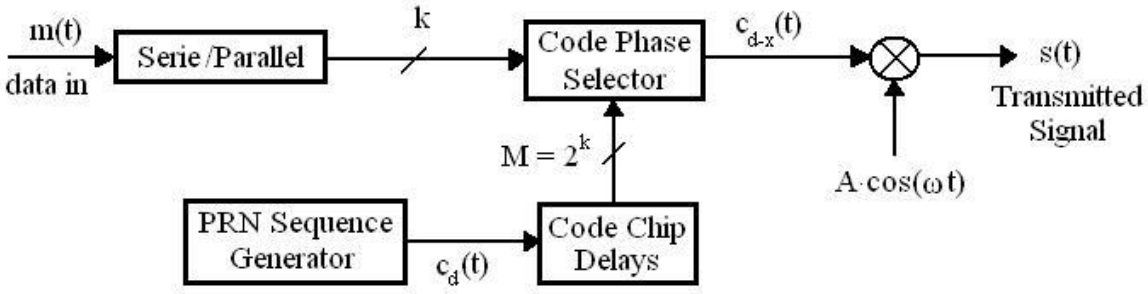

Figure 7-1: CSK modulator scheme

The received signal after the filtering of the RF/IF block can be modeled as:

$$w(t) = h_{RF/IF}(t) * v(t) \tag{7-5}$$

$$w(t) = s'(t) + n'(t) \tag{7-6}$$

Where:

- w(t): Received signal at the RF/IF block output
- $h_{RF/IF}(t)$: RF/IF block filter

- s'(t): Transmitted signal at the transmitter antenna input filtered by the RF/IF block
- n'(t): Additive narrow-band Gaussian noise with a power $P_n = B \cdot N_0$
- B: RF/IF filter bandwidth

In order to simplify the studies conducted in this dissertation, the filtering of the RF/IF block output is assumed to not affect the received useful signal. Therefore, the signal can be modeled as:

$$w(t) = s(t) + n'(t) \tag{7-7}$$

The demodulator scheme of the received signal at the RF/IF block output can have two different structures as has been commented in the previous section. Nevertheless, remember that these two structures are equivalent. In fact, the only difference is the complexity of the scheme, because the first scheme uses a bank of matched filters and the second one only Fourier transforms blocks. Therefore, the output provided by both demodulator schemes can be modeled with the same mathematical expression. The demodulator scheme of the Fourier transforms configuration is shown below:

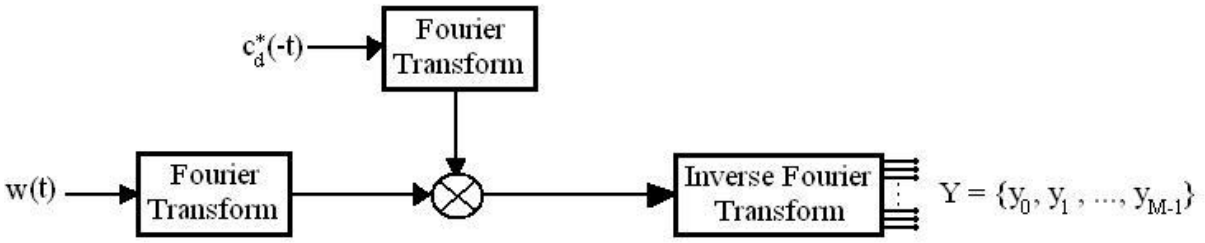

Figure 7-2: CSK demodulator scheme

The previous demodulator scheme works as explained next. The product between the conjugate fundamental code Fourier transform and the received signal Fourier transform is equivalent to conduct the correlation between the fundamental code and the received signal but in the frequency domain. And, after the application of the inverse Fourier transform, the temporal correlation between the fundamental code and the received signal is obtained. Since the received signal is a circularly shifted version of the fundamental code, the result is equal to the fundamental code autocorrelation. This autocorrelation is not centered at 0, but it has its main peak at the integer value marking the transmitted circularly shifted version of the fundamental code. This means that in order to find which circularly shifted version of the fundamental code has been transmitted, the demodulator has to inspect all the inverse Fourier transform outputs. Note that if the fundamental code has more chips, L, than symbols belonging to alphabet, M, the only outputs of the inverse Fourier transform which has to be inspected are the outputs which marks the correlation between the fundamental code and one of the circularly shifted versions of the fundamental code. The inverse Fourier Transform (IFT) outputs are the y_i components of the Y vector of Figure 7-3.

Normalizing the IFT outputs by the maximum amplitude value of the fundamental code autocorrelation, assuming that the autocorrelation function of the fundamental code is 0 for values farther apart than T_c from the autocorrelation center and knowing that the shifts are m times T_c, where m is an integer number, each y_i component can be modeled when the $c_{d\text{-}x}(t)$ symbol has been transmitted as:

$$y_i = \begin{cases} \pm 1 + n_i & x = i \\ n_i & x \neq i \end{cases} \qquad i = 0..M-1 \tag{7-8}$$

Where:

- n_i: Independent narrow-band Gaussian noises with power equal to $\sigma^2 = R_s/(2 \cdot C/N_0)$
- R_s: CSK symbol transmission rate

The noises n_i are independent because the transmitted symbols are orthogonal by definition, M-ary orthogonal signaling, and it is assumed that the beginning of the received symbol is known. Note that the correlation of two orthogonal symbols $R_{X1X2}(\tau)$ is 0 at $\tau=0$, and the Y vector components output are the result of the correlation evaluated at $\tau=0$ between the received symbol and the symbol specified by the component when the beginning of the received signal is known.

Finally, note that the receiver must also know the beginning of the transmitted sequence in order to determine which circularly shifted version of the fundamental code has been transmitted. If the beginning is not known, the receiver cannot determine which component of the Y vector represents which circularly shifted version of the code, and thus cannot recover the transmitted bits.

7.4. Code Shift Keying theoretical hard demodulation performance

The Code Shift Keying technique, as has been commented in section 7.1, is a type of M-ary orthogonal signaling (MOS). This fact has been verified by [DILLARD et al., 2003]. Therefore, the hard demodulation performance in terms of BER as a function of the E_b/N_0 has already been calculated and largely commented in the literature [PROAKISc, 2001]. Moreover, the hard demodulation performance of the bi-orthogonal CSK technique, which is equivalent to a type of M-ary bi-orthogonal signaling technique, has also been thoroughly analyzed.

The main results of both cases are summarized next.

7.4.1. M-ary orthogonal signaling demodulation performance

The demodulation process of a M-ary orthogonal signaling is a coherent demodulation process. This process consists in choosing among the M components of the Y vector the component which has the largest positive value if the polarity of the transmitted symbols has been chosen positive or in choosing the largest negative value if the polarity of the transmitted symbols has been chosen negative.

The BER as a function of the message E_s/N_0 is equal to [PROAKISc, 2001]:

$$P_M = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} \left[1 - \left(\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{y} e^{-x^2/2} dx \right)^{M-1} \right] \exp\left[-\frac{1}{2} \left(y - \sqrt{\frac{2E_S}{N_0}} \right)^2 \right] dy \tag{7-9}$$

$$BER = \frac{2^{k-1}}{2^k - 1} P_M \tag{7-10}$$

$$\frac{E_b}{N_0} = \frac{E_S}{N_0} - 10 \cdot \log_{10}(k) \tag{7-11}$$

Where:

- E_s/N_0: signal symbol energy per symbol to noise power density ratio
- P_M: Probability of orthogonal CSK symbol error

7.4.2. M-ary bi-orthogonal signaling demodulation performance

The demodulation process of a M-ary bi-orthogonal signaling is a non-coherent demodulation process. This process consists in following two steps. First, the receiver chooses among the M components of the Y vector the component which has the largest absolute value, in other words, the largest energy. The selected component determines the values of the bits which have selected the M/2-ary orthogonal symbol to transmit; these bits are, with respect to annex I.1.2 definition, the bits of the B^{k-1} set. Second, the receiver determines the value of the bit which has selected the polarity of the transmitted M-ary bi-orthogonal symbol. This bit is, with respect to section I.1.2 definition, the β bit.

The symbol error rate as a function of the E_s/N_0 is equal to [PROAKISc, 2001]:

$$P_M = 1 - \frac{1}{\sqrt{2\pi}} \int_{-\sqrt{2E_S/N_0}}^{+\infty} \left(\frac{1}{\sqrt{2\pi}} \int_{-\left(v+\sqrt{2E_S/N_0}\right)}^{v+\sqrt{2E_S/N_0}} e^{-x^2/2} dx \right)^{M/2-1} e^{-v^2/2} dv \tag{7-12}$$

Where:

- E_s/N_0: signal symbol energy per symbol to noise power density ratio
- P_M: Probability of bi-orthogonal CSK symbol

Finally, the BER can be found by applying the same transformation as in equation (7-10). Nevertheless, whereas in equation (7-10) the transformation is exact, in the M-ary bi-orthogonal signaling case, the transformation is an approximation. In fact, the transformation assumes that all the symbols errors are equiprobable. But, in reality, the symbol error probability of choosing the transmitted symbol with the inverse polarity is lower than the symbol error probability of choosing another M-ary orthogonal signal instead of the transmitted one.

$$BER \approx \frac{2^{k-1}}{2^k - 1} P_M \tag{7-13}$$

7.5. Code Shift Keying likelihood ratios mathematical expressions

In previous sections, the demodulation process of orthogonal and bi-orthogonal CSK signaling techniques have been described and their demodulation performance expressions have been given.

These processes and expressions are only valid when the demodulator searches to provide hard outputs -values of the bits- and the autocorrelation function between the received circularly shifted version of the code and the generated local fundamental code is assumed to be perfect. This means that the previous demodulation processes can only be used when their output is used as either demodulated information or as hard inputs for channel decoders.

However, the more powerful channel codes only accept as inputs the transmitted bits probabilities, or their likelihood ratios (LR). Therefore, in order to allow the implementation of such soft input channel codes over a CSK signal transmission, the CSK bits likelihood ratios have to be determined. The demodulation which uses likelihood ratios to determine the bits values is called soft demodulation.

This section is divided into three subsections. In the first subsection, the likelihood ratio (LR) expression for the M-orthogonal CSK signaling technique is presented. In the second subsection, the likelihood ratio (LR) expression of a bi-orthogonal M-ary CSK is calculated by modifying the expression calculated in the first subsection. In the third and last subsection, the soft demodulation is compared through simulations to the hard demodulation.

Finally, note that the likelihood ratio expressions are also valid for a M-ary orthogonal signaling and for a M-ary bi-orthogonal signaling.

7.5.1. M-ary orthogonal signaling likelihood ratio mathematical expression

The likelihood ratio (LR) of a transmitted bit is defined as shown below [PROAKISc, 2001]:

$$LR = \frac{P(Bit = 1|Y)}{P(Bit = 0|Y)} \tag{7-14}$$

Assuming that each set of k bits represent a M-ary orthogonal signal where $M=2^k$, the probability of the received bit b_m being equal to 0 or 1 when the Y vector is observed is defined as has been done in annex C.1:

$$P(b_m = 1 \ or \ 0|Y) = \frac{P(Y|b_m = 1 \ or \ 0) \cdot P(b_m = 1 \ or \ 0)}{P(Y)} \tag{7-15}$$

Therefore, since the bits are equiprobable, the likelihood ratio expression is equal to:

$$LR(b_m) = \frac{P(Y|b_m = 1)}{P(Y|b_m = 0)} \tag{7-16}$$

This means that in order to find the likelihood ratio, the probabilities specified in the numerator and in the denominator have to be calculated. Nevertheless, these probabilities cannot be directly obtained since each M-ary orthogonal signal carries a symbol mapping k bits. Therefore, the probabilities obtained from the observation of the Y vector determine the probability of a symbol rather than the probability of only one individual bit.

The observed probability expression for a given group of transmitted bits is shown below.

$$P_{observed} = P(Y|b_{k-1}, \ldots, b_1, b_0) = P(Y|B^k) \tag{7-17}$$

$$B^k = b_{k-1}, \ldots, b_1, b_0, \tag{7-18}$$

Where:

- B^k: Set of k bits which determine the transmitted symbol

From equation (7-17), the probability of receiving Y when b_m is equal to 0 or to 1 can be calculated as presented next:

$$P\left(Y\middle|b_m=1\right)=\sum_{x_0}\sum_{x_1}\cdots\sum_{x_{k-1}}P\left(Y\middle|b_{k-1}=x_{k-1},\ldots,b_m=1,\ldots,,b_1=x_1,b_0=x_0\right)\cdot \cdot P\left(b_{k-1}=x_{k-1},\ldots,b_m=1,\ldots,b_1=x_1,b_0=x_0\right) \quad (7\text{-}19)$$

Note that in equation (7-19), the only summation which is not applied is the summation having as sub-index x_m.

The summation having as sub-index x_p has to be interpreted as shown below.

$$\sum_{x_p}P(Y|b_p=x_p)\cdot P(b_p=x_p)=P(Y|b_p=1)\cdot P(b_p=1)+P(Y|b_p=0)\cdot P(b_p=0) \quad (7\text{-}20)$$

Therefore, in this case, the summation is:

$$\begin{aligned}&\sum_{x_p}P(Y|b_{k-1}=x_{k-1},\ldots,b_p=x_p,\ldots,b_0=x_0)\cdot P(b_{k-1}=x_{k-1},\ldots,b_p=x_p,\ldots,b_0=x_0)=\\&P(Y|b_{k-1}=x_{k-1},\ldots,b_p=1,\ldots,b_0=x_0)\cdot P(b_{k-1}=x_{k-1},\ldots,b_p=1,\ldots,b_0=x_0)+\\&+P(Y|b_{k-1}=x_{k-1},\ldots,b_p=0,\ldots,b_0=x_0)\cdot P(b_{k-1}=x_{k-1},\ldots,b_p=0,\ldots,b_0=x_0)\end{aligned} \quad (7\text{-}21)$$

And, since the bits are equiprobable, expression (7-19) is simplified to:

$$P\left(Y\middle|b_m=1\right)=\frac{1}{M}\cdot\sum_{x_0}\sum_{x_1}\cdots\sum_{x_{k-1}}P\left(Y\middle|b_0=x_0,\ldots,b_m=1,\ldots,b_{k-1}=x_{k-1}\right) \quad (7\text{-}22)$$

The individual terms of equation (7-22) can be determined since they are the probability of the Y vector observation when a determined group of bits, B^k, has been transmitted. Note that each individual term of equation (7-22) is the same term as the term in equation (7-17). Therefore, denoting each different group of bits as $B^k{}_n$, where n represents the symbol mapping this group of bits, equation (7-22) is equal to:

$$P\left(Y\middle|b_m=1\right)=\frac{1}{M}\cdot\sum_{n}P\left(Y\middle|B^k{}_{n,bm=1}\right) \quad (7\text{-}23)$$

Where:

- $B^k{}_{n,bm=1}$: Set of k bits mapped by symbol n. The bit b_m of symbol n is equal to 1.

The conditioned probability of the Y vector observation can be modeled as a product of the conditioned probabilities of the observation of each individual Y vector component since the Y vector components are independent among them. The reason is that the noise of each Y vector component is independent from the noise of any other component as has been explained in section 7.3. Therefore, the conditioned probability of the Y vector observation can be expressed as

$$P\left(Y\middle|B^k\right)=P\left(y_0\middle|B^k\right)\cdot P\left(y_1\middle|B^k\right)\cdot\ldots\cdot P\left(y_{M-1}\middle|B^k\right) \quad (7\text{-}24)$$

Each individual Y component is one of the demodulator outputs of Figure 7-3. Therefore, the conditioned probability of each individual component observation can be modeled as a Gaussian variable with variance equal to σ^2, and with a mean that depends on the transmitted groups of bits, $B^k{}_n$. More specifically, the mean of the Gaussian variable representing the y_i component observation is equal to 1, if the transmitted group of bits, $B^k{}_n$, is mapped by the M-ary orthogonal symbol i; in other words, if i is equal to n. Otherwise the mean is 0.

The probabilities of the observation of the Y vector components when the group of bits, $B^k{}_n$, is transmitted are:

$$P\left(y_i \middle| B^k{}_n\right)=\begin{cases}\dfrac{1}{\sqrt{2\pi\sigma}}\cdot\exp\left(-\dfrac{1}{2}\cdot\dfrac{y_i^2}{\sigma^2}\right) & i\neq n\\ \dfrac{1}{\sqrt{2\pi\sigma}}\cdot\exp\left(-\dfrac{1}{2}\cdot\dfrac{(y_i-1)^2}{\sigma^2}\right) & i=n\end{cases} \tag{7-25}$$

Therefore, expression (7-24) is equivalent to:

$$P\left(Y\middle|B^k{}_n\right)=\frac{1}{(2\pi\sigma)^{M/2}}\cdot\exp\left(-\frac{1}{2\sigma^2}\cdot\sum_{i=0}^{M-1}y_i^2\right)\exp\left(\frac{1}{2\sigma^2}\cdot(2\cdot y_n-1)\right) \tag{7-26}$$

Expression (7-26) can be expressed as:

$$P\left(Y\middle|B^k{}_n\right)=K(Y)\cdot\exp\left(\frac{y_n}{\sigma^2}\right) \tag{7-27}$$

$$K(Y)=\frac{1}{(2\pi\sigma)^{M/2}}\cdot\exp\left[-\frac{1}{2\sigma^2}\left(1+\sum_{i=0}^{M-1}y_i^2\right)\right] \tag{7-28}$$

Once the previous expression has been determined, equation (7-23) can be rewritten as:

$$P\left(Y\middle|b_m=x_m\right)=\frac{K(Y)}{M}\cdot\sum_{\substack{i\\ b_m=x_m}}\exp\left(\frac{y_i}{\sigma^2}\right) \tag{7-29}$$

Where the summation includes all the y_i components of the vector Y which represent a set of bits, B^k, having the bit b_m equal to x_m.

The final expression of the likelihood ratio of the bits mapped by a M-ary orthogonal symbol is found by using equation (7-16) and (7-29) and by taking into account that K(Y)/M has the same value for b_m=1 and b_m=0. The expression is equal to:

$$LR(b_m)=\sum_{\substack{i\\ b_m=1}}\exp\left(\frac{y_i}{\sigma^2}\right)\Bigg/\sum_{\substack{i\\ b_m=0}}\exp\left(\frac{y_i}{\sigma^2}\right) \tag{7-30}$$

Therefore, it can be seen that any bit likelihood ratio can be obtained by only changing the distribution of the Y vector components between the numerator and the denominator of equation (7-30). The general rule is the following one: in order to calculate the likelihood ratio of the bit b_m, the Y vector components which map b_m equal to 1 are used at the numerator, and the Y vector components which map b_m equal to 0 are used at the denominator.

For example, the likelihood ratio of b_0, when the mapping between the transmitted bits and the orthogonal signals is the mapping defined in subsection I.1.1, is:

$$LR(b_0)=\frac{\exp\left(\frac{y_1}{\sigma^2}\right)+\exp\left(\frac{y_3}{\sigma^2}\right)+\exp\left(\frac{y_5}{\sigma^2}\right)+\ldots+\exp\left(\frac{y_{M-1}}{\sigma^2}\right)}{\exp\left(\frac{y_0}{\sigma^2}\right)+\exp\left(\frac{y_2}{\sigma^2}\right)+\exp\left(\frac{y_4}{\sigma^2}\right)+\ldots+\exp\left(\frac{y_{M-2}}{\sigma^2}\right)} \tag{7-31}$$

Finally, an algorithm to speed the bits likelihood ratio calculation is presented:

1. Calculating and storing all the exponential values: $T_m = \exp(y_m/\sigma^2)$
2. Setting the value $M = 2^k$
3. For i = 0 to k-1
    a. Calculating the b_i likelihood ratio
        i. $P_1 = T_1 + T_3 + \ldots + T_{M-1}$
        ii. $P_0 = T_0 + T_2 + \ldots + T_{M-2}$
        iii. $LR(b_i) = P_1/P_0$
    b. Calculating the new T_m values
        i. For m = 0 to (M/2 – 1) → $T_m = T_{2m} + T_{2m+1}$
    c. Setting the new M value: M = M/2

This algorithm is illustrated and justified in the figure displayed below. In this example, 4 bits are modulated in a 16-ary orthogonal symbol.

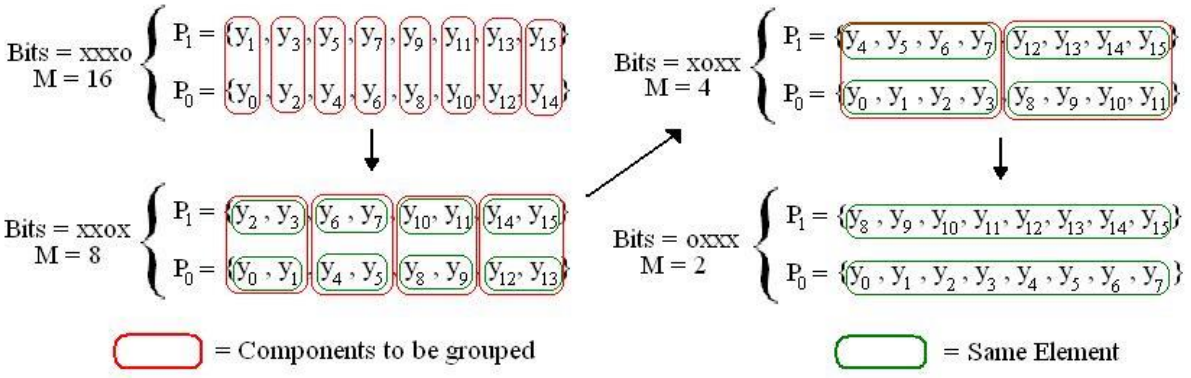

Figure 7-3: 16-ary orthogonal signalling example - Bits likelihood ratio calculation for 4 bits mapping an orthogonal signal

7.5.2. M-ary bi-orthogonal signaling likelihood ratio mathematical expression

The likelihood ratio mathematical expression of a bit transmitted inside a bi-orthogonal M-ary signal can be calculated from the same expressions (7-16) and (7-23) used to calculate the likelihood ratio of a bit transmitted inside an orthogonal M-ary signal. However, in this case, the likelihood ratio of the bit mapped with the symbol polarity –or sign– has to be calculated differently from the likelihood ratio of the bits mapped by the M/2-ary signal. Remember that a bi-orthogonal M-ary signaling technique is equivalent to an orthogonal M/2-ary signaling technique which uses both polarities –or signs– of the symbols to transmit the bits.

In this research work, two types of soft demodulation processes having each one its own likelihood ratio mathematical expression have been proposed. The first option consists in deciding which polarity the received symbol has, and in using the estimated polarity of the symbol in order to decide which the likelihood ratios of the bits transmitted inside the M/2-ary orthogonal symbol are. In this case, the likelihood ratio calculations of the bits mapped by the M/2-ary orthogonal symbol take into account the probability of the bit mapped by the symbol polarity. The second option consists in calculating the likelihood ratios of all the bits, the bit mapped by the symbol polarity and the bits mapped by the M/2-ary orthogonal symbol, at the

same time. In this case, the probability of the bit mapped by the symbol polarity is not taken into account in the calculation of the likelihood ratios of the bits mapped by the M/2-ary orthogonal symbol.

7.5.2.1. Likelihood ratio of the bit mapped by the symbol polarity

The two previous presented demodulations processes have in common the likelihood ratio calculation of the bit mapped by the signal polarity. This means that the following likelihood ratio expression is valid for both options.

First of all, remember that from section I.1.2, the last bit fed to the modulator determines the symbol polarity. The denotation of the group of input bits mapped by a bi-orthogonal CSK symbol is given below, where β denotes the bit mapped by the symbol polarity:

$$R^k = \{\beta, b_{k-2}, \ldots, b_1, b_0\} = \{\beta, B^{k-1}\} \tag{7-32}$$

Second, in this case, the observed conditional probability at the demodulator output is:

$$P_{observed} = P\left(Y \middle| \beta, b_{k-1}, \ldots, b_1, b_0\right) = P\left(Y \middle| R^k\right) \tag{7-33}$$

Third, the probabilities of the numerator and the denominator of the expression (7-16) can still be calculated using expression (7-23), but in this case the terms of the summation are the new observed conditional probabilities expressed in (7-33). Therefore, each individual term is slightly different than the term defined in the M-ary orthogonal signaling technique in this case. In fact, the difference is the mean of the Gaussian variable: the mean is positive if the symbol polarity is positive, β=0, and the mean is negative is the symbol polarity is negative, β=1. Therefore, denoting each different group of transmitted bits mapped by the M/2-ary orthogonal symbol as $B^{k-1}{}_n$ and denoting each different group of transmitted bits by the M-ary bi-orthogonal symbol as $R^k{}_n$, the conditioned probability of the observation of the Y vector component y_i can be modeled as:

$$P\left(y_i \middle| R^k{}_n\right) = \begin{cases} \dfrac{1}{\sqrt{2\pi\sigma}} \cdot \exp\left(-\dfrac{1}{2} \cdot \dfrac{y_i^2}{\sigma^2}\right) & i \neq n \\ \dfrac{1}{\sqrt{2\pi\sigma}} \cdot \exp\left(-\dfrac{1}{2} \cdot \dfrac{(y_i - 1)^2}{\sigma^2}\right) & i = n;\ \beta = 0 \\ \dfrac{1}{\sqrt{2\pi\sigma}} \cdot \exp\left(-\dfrac{1}{2} \cdot \dfrac{(y_i + 1)^2}{\sigma^2}\right) & i = n;\ \beta = 1 \end{cases} \tag{7-34}$$

Consequently, expression (7-34) is equal to:

$$P\left(Y \middle| R^k{}_n\right) = \frac{1}{(2\pi\sigma)^{M/4}} \cdot \exp\left(-\frac{1}{2\sigma^2} \cdot \sum_{i=0}^{M/2-1} y_i^2\right) \exp\left(\frac{1}{2\sigma^2} \cdot \left(2 \cdot \left(\frac{1}{2} - \beta\right) y_n - 1\right)\right) \tag{7-35}$$

Finally, since each component of the Y vector can represent the two polarities of its associated M/2-ary orthogonal symbol, the summation of the numerator and the denominator contain all the Y vector components. The only difference between the numerator and denominator summations is the β value. Therefore, the likelihood ratio expression of the bit mapped by the symbol polarity is found using (7-16), (7-23) and (7-35) and is equal to:

$$LR(\beta) = \sum_{i=0}^{M/2-1} \exp\left(-\frac{y_i}{\sigma^2}\right) \Big/ \sum_{i=0}^{M/2-1} \exp\left(\frac{y_i}{\sigma^2}\right) \tag{7-36}$$

7.5.2.2. Likelihood ratio of the bits mapped by the M/2-ary orthogonal symbol

The two different options presented at the beginning of section 7.5.2 which can be used to calculate the likelihood ratio of the bits mapped by the M/2-ary orthogonal symbol of a bi-orthogonal M-ary signal are presented in this subsection.

Both options calculate the likelihood ratio expression of a bit mapped by the M/2-ary orthogonal symbol by using the total probability formula:

$$LR(b_m) = \frac{P(Y|b_m = 1, \beta = 0) \cdot P(\beta = 0) + P(Y|b_m = 1, \beta = 1) \cdot P(\beta = 1)}{P(Y|b_m = 0, \beta = 0) \cdot P(\beta = 0) + P(Y|b_m = 0, \beta = 1) \cdot P(\beta = 1)} \tag{7-37}$$

Moreover, the reader must know that the same algorithm implemented to speed up the bits likelihood ratios for the M-ary orthogonal signaling case, can be also implemented for M-ary bi-orthogonal signaling case.

7.5.2.2.1. First Option

The first option uses the estimated probability of the bit mapped by the symbol polarity. Therefore equation (7-37) becomes:

$$LR(b_m) = \frac{P(Y|b_m = 1, \beta = 0) \cdot P(\beta = 0|Y) + P(Y|b_m = 1, \beta = 1) \cdot P(\beta = 1|Y)}{P(Y|b_m = 0, \beta = 0) \cdot P(\beta = 0|Y) + P(Y|b_m = 0, \beta = 1) \cdot P(\beta = 1|Y)} \tag{7-38}$$

The terms $P(Y|b_m, \beta)$ can be calculated as showed in equation (7-29) using equation (7-23) for the M-ary orthogonal signaling case. The only difference is found when $\beta=1$ since the value of y_m has its sign inversed in this case.

$$P(Y|b_m = x_m, \beta = \beta_0) = \frac{1}{(M/2)} \cdot \sum_n P\left(Y|R^k{}_{n, bm=xm, \beta=\beta_0}\right) \tag{7-39}$$

Therefore, $P(Y|b_m, \beta)$ can be expressed from equations (7-34) and (7-39) as:

$$P(Y|b_m = x_m, \beta) = \frac{2 \cdot K(Y, \beta)}{M} \cdot \sum_{\substack{i \\ b_m = x_m}} \exp\left(\frac{2 \cdot (0.5 - \beta) \cdot y_i}{\sigma^2}\right) \tag{7-40}$$

$$K(Y, \beta) = \frac{1}{(2\pi\sigma)^{M/4}} \cdot \exp\left[-\frac{1}{2\sigma^2}\left(1 + \sum_{i=1}^{M/2} y_i^2\right)\right] \tag{7-41}$$

Note that $2 \cdot K(Y,\beta)/M$ has the same value for $b_m = 0$ and for $b_m = 1$. Finally, the likelihood ratio expression is:

$$LR(b_m) = \frac{\left(\sum_{i=0}^{M/2-1} \exp\left(\frac{y_i}{\sigma^2}\right)\right) \cdot \left(\sum_{\substack{i \\ b_m=1}} \exp\left(\frac{y_i}{\sigma^2}\right)\right) + \left(\sum_{i=0}^{M/2-1} \exp\left(-\frac{y_i}{\sigma^2}\right)\right) \cdot \left(\sum_{\substack{i \\ b_m=1}} \exp\left(-\frac{y_i}{\sigma^2}\right)\right)}{\left(\sum_{i=0}^{M/2-1} \exp\left(\frac{y_i}{\sigma^2}\right)\right) \cdot \left(\sum_{\substack{i \\ b_m=0}} \exp\left(\frac{y_i}{\sigma^2}\right)\right) + \left(\sum_{i=0}^{M/2-1} \exp\left(-\frac{y_i}{\sigma^2}\right)\right) \cdot \left(\sum_{\substack{i \\ b_m=0}} \exp\left(-\frac{y_i}{\sigma^2}\right)\right)} \quad (7\text{-}42)$$

7.5.2.2.2. Second Option

The second option does not use the estimated probability of the bit mapped by the symbol polarity. This option assumes that the symbol polarity is equiprobable. Therefore equation (7-37) becomes:

$$LR(b_m) = \frac{P(Y|b_m = 1, \beta = 0) + P(Y|b_m = 1, \beta = 1)}{P(Y|b_m = 0, \beta = 0) + P(Y|b_m = 0, \beta = 1)} \quad (7\text{-}43)$$

The numerator and denominator expression are simply the addition of expression (7-39) evaluated by the two possible β values.

$$P(Y|b_m = x_m, \beta = 0) + P(Y|b_m = x_m, \beta = 1) = \frac{4 \cdot K(Y, \beta)}{M} \cdot \sum_{\substack{i \\ b_m=x_m}} \cosh\left(\frac{y_i}{\sigma^2}\right) \quad (7\text{-}44)$$

Finally, the likelihood ratio expression is:

$$LR(b_m) = \frac{\sum_{\substack{i \\ b_m=1}} \cosh\left(\frac{y_i}{\sigma^2}\right)}{\sum_{\substack{i \\ b_m=0}} \cosh\left(\frac{y_i}{\sigma^2}\right)} \quad (7\text{-}45)$$

7.5.3. Likelihood ratios mathematical expressions verification

In this section, several simulations have been executed in order to simulate the transmission of CSK symbols through an AWGN channel. These simulations do not implement any channel code over the transmitted information and thus they are used to compare the demodulation performance between the hard outputs demodulators described in section 7.3, and the likelihood ratio expressions calculated in this section. Moreover, this comparison also shows whether it is interesting to use a bi-orthogonal M-ary modulation to transmit k bits instead of using an orthogonal M-ary modulation.

Two different figures are presented in order to compare the hard output and soft output demodulator performance. Each figure represents the BER as a function of E_b/N_0 for different CSK modulation orders of M. The chosen amounts of bits are 6 and 8 ($M = 2^k$). In each figure, 7 different demodulation performances are plotted. Three curves represent the demodulation performance of an orthogonal M-ary signaling and the remaining four ones represent the demodulation performance of a bi-orthogonal M-ary signaling. The three curves of the M-ary orthogonal case are: the theoretical hard output demodulation performance (see

equation (7-10)), the hard output demodulation performance calculated from Monte Carlo simulations and the soft output demodulation performance obtained from Monte Carlo simulations using equation (7-30). The four curves of the M-ary bi-orthogonal case are: the theoretical hard output demodulation performance (see equation (7-13)), the hard output demodulation performance calculated from Monte Carlo simulations, the soft output demodulation performance obtained from Monte Carlo simulations using equations (7-36) and (7-42), and the soft output demodulation performance obtained from Monte Carlo simulations using equations (7-36) and (7-45).

All the simulations have been executed assuming ideal code delay and carrier phase tracking processes.

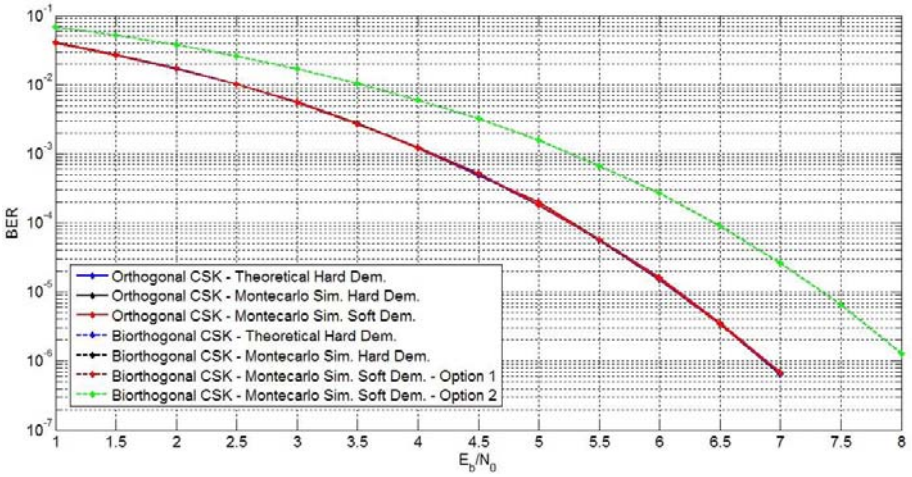

Figure 7-4: BER vs Eb/N0 for a transmission employing CSK symbols which map 6 bits

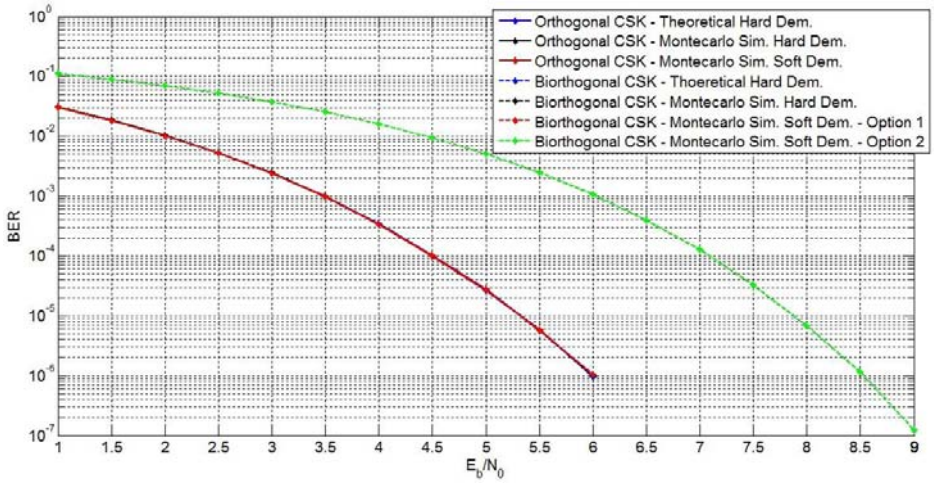

Figure 7-5: BER vs Eb/N0 for a transmission employing CSK symbols which map 8 bits

Note that from now on, the M-ary orthogonal signaling is called orthogonal CSK and the M-ary bi-orthogonal signaling is called bi-orthogonal CSK.

The first observation that can be made from the two previous figures is that the demodulation performance of an orthogonal CSK modulation using either soft or hard outputs is the same. This means that the theoretical likelihood ratio expression of the bits mapped by an orthogonal CSK given in equation (7-30) is verified as the equivalent soft output method of the hard output method defined by the literature [PROAKISc, 2001].

The second observation made from Figure 7-4 and Figure 7-5 is that a bi-orthogonal CSK signaling obtains the same demodulation performance when employing either the hard output method or the soft output method first option. Therefore, it has been verified that the soft output method first option is equivalent to the hard output method proposed by the literature [PROAKISc, 2001].

Besides, the demodulation performance of the soft output method second option is worse than the demodulation performance of the soft output method first option. The reason is the special nature of the individual terms of the numerator and denominator summation of the likelihood ratio expression in equation (7-45). These terms are hyperbolic cosines for the 2^{nd} option whereas they are exponentials for the 1^{st} option. And, since the hyperbolic cosine function grows at a slower rate than the exponential function, the first option requires input values less noised than the second option values in order to achieve the same output values. Therefore, the Y vector observation has more influence over the final likelihood ratio expressions on the first option than on the second option. Besides, the hyperbolic cosine function process equally the positive and negative terms whereas the exponential function does not.

Moreover, another observation that can be made about the second option of the soft output method is that when the number of bits mapped by a bi-orthogonal CSK symbol decrease, the demodulation performance difference between the first option and the second option of the soft output methods also decreases. In fact, taking as reference a BER of 10^{-5}, the difference in terms of E_b/N_0 is about 1.25 dB for 64-CSK (6 bits) and 2.6 dB for 256-CSK (8 bits). The reason of this phenomenon is found again on the difference of growth between the hyperbolic cosine terms and the exponential terms. The increase of the number of bits mapped by a bi-orthogonal CSK symbol is equivalent to increase the number of terms used in the likelihood ratio mathematical expression. And since the contribution of the hyperbolic terms to the likelihood ratio is more noised than the contribution of the exponential terms, an increase of the number of terms implies an increase in the difference of performance between the first and the second option.

One last remark to make about the soft output second option is the decrease of its demodulation performance when the number of bits mapped by a bi-orthogonal CSK symbol increases. In fact, the number of bits when the E_b/N_0 is fixed determines the E_s/N_0, where the E_s/N_0 determines the influence of the noise on the Y vector components. Therefore, it could be thought that the increase of the E_s/N_0 value should improve the demodulation performance as it is the case for the other curves. But, since the demodulation performance is decreased, it can be concluded that the increase of the number of hyperbolic terms used to calculate the likelihood ratios has a greater impact over the likelihood calculation than the decrease of the noise on the Y vector components.

Finally, in this section, the standard deviations of the proposed likelihood ratio expressions are calculated since the standard deviation of a likelihood ratio is an important indicator of its performance when it is used as input to a channel code decoder block. The reason is given next. The likelihood ratio of a bit is equal to 1 when the probability of receiving this bit equal to 1 is the same as the probability of receiving this bit equal to 0. Therefore, a channel code using soft inputs is interested in having either the larger or the smaller likelihood ratios in order to estimate reliable values of the received bits. In other words, a channel code has fewer difficulties in decoding a word with bits likelihood ratios far from 1 than in decoding a word with bits likelihood ratios around 1. The statement is only true if the erroneous demodulated bits have a likelihood ratio always near 1 which is normally the case.

In order to inspect this characteristic, the lognormal likelihood ratios mean and standard deviation of the previous orthogonal and bi-orthogonal CSK configurations have been calculated. Figure 7-6 shows the results.

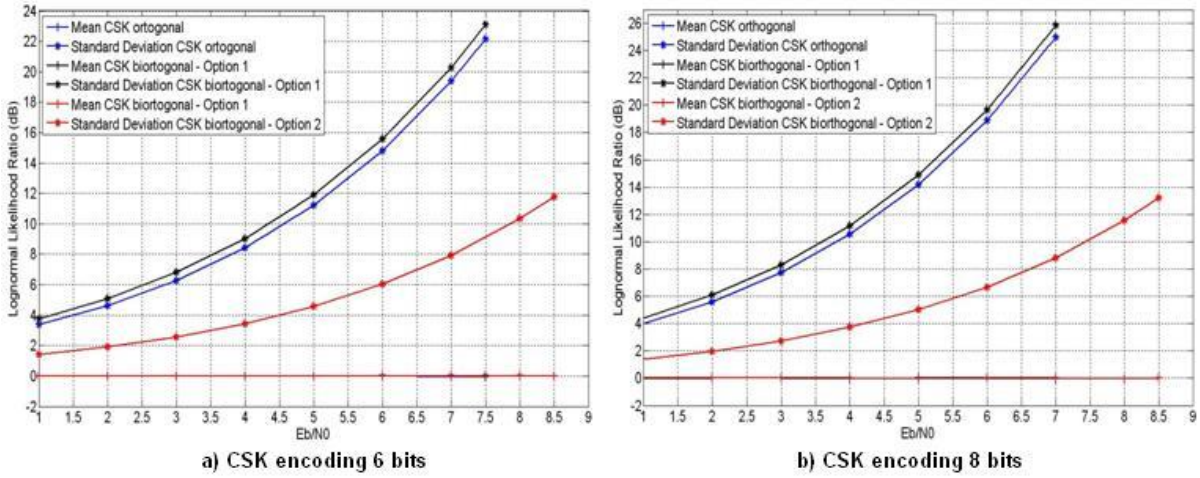

Figure 7-6: Lognormal likelihood ratio mean and standard deviation for CSK symbols encoding 6 and 8 bits

The first observation that can be made from Figure 7-6 is that the soft output method second option for a bi-orthogonal CSK has a smaller standard deviation than the standard deviation of the soft output method first option and than the standard deviation of the soft output method for an orthogonal CSK. Therefore, the difference of demodulation performance between the soft output method second option and the two other methods will increase when a channel code is implemented over the transmitted CSK bits. This statement has been verified through simulations not presented in this PhD manuscript.

Moreover, it can be seen that the more bits are mapped by a CSK symbol, the larger the lognormal likelihood ratio standard deviation is. Therefore, in addition of having a better hard demodulation performance, a CSK modulation modulating more bits will obtain a farther performance improvement when a channel code is implemented. Nevertheless, this improvement may not be significant between CSK modulations modulating almost the same number of bits. This statement has been verified in section 7.8.

Finally, the reader must note that the growth of the standard deviation of the lognormal likelihood ratio is due to the growth of E_s/N_0 for a fixed value of E_b/N_0. However, if it is the E_s/N_0 value which is fixed and thus it is the E_b/N_0 which is variable, the results are inverted: the fewer bits are encoded by a CSK symbol, the bigger the standard deviation of the likelihood ratio is.

The results of this section are used in section 7.8.

7.6. Source packets mapping of CSK modulation

The implementation of a channel code over the bits transmitted by a CSK modulation introduces a new interesting element of study: the source packet mapping.

A packet is defined as a unit of information composed by several bits. A packet of bits is the denotation used in this dissertation for a word which is transmitted through a channel. The information bits generated by the information source are grouped into information words. Each information word is coded with a channel code and the result is called code word.

Therefore, since the information unit transmitted through the channel is the codeword, a packet is the denotation used for a coded word transmitted through a channel. One characteristic of the packet denotation assumed in this Ph.D. work is that after the packet reception, the information bits carried by the packet can directly be transmitted to higher layers of the receiver. This means that all the transmitted bits inside a packet must be protected by the same channel code.

The source packet mapping is defined as the distribution between the bits mapped by each CSK symbol and the bits belonging to the same packet; or, in other words, which bits mapped into a CSK symbol belong to a given packet A, which bits belong to a given packet B, etc. Therefore, one question to be asked is which source packet mapping provides the lowest BER for a communication.

The question about the optimal source packet mapping which has the lowest BER for each transmitted packet is irrelevant if no channel code is implemented because the errors introduced by the transmission channel cannot be corrected. Therefore, since the errors of the determination of the transmitted CSK symbols are independent when the signal is transmitted through an AWGN channel and the value of each bit mapped by an erroneous estimated CSK symbol is random, the BER of any packet of any possible source packet mapping is the same. This statement is verified later. This means that there is no difference of BER between a mapping where all the bits mapped by CSK symbol belong to the same packet and a mapping where each bit mapped by a CSK symbol belongs to a different packet.

However, due to the implementation of channel codes on the transmitted bits, some packets defined by a given source packet mapping can have after their transmission through an AWGN channel a pattern of errors easier to correct than the pattern of errors presented by another packet defined by another source packet mapping. More specifically, the average number of errors is the same for any packet defined by any source packet mapping, but the variance of the number of errors can change significantly among different source packet mappings. Therefore, it can be much more interesting to have either a small or a large variance depending on the transmission channel.

On one hand, it can be interesting to have a small variance in order to keep the majority of packets with a total number of bit errors very near to the average number of bit errors. The reason is that the channel code can be able to correct any packet having a quantity of errors a bit larger than the average number of errors introduced by the channel. However, if the variance is very high, a larger amount of packets than the amount of packets when the variance is low will not be corrected since their total number of errors will very far from the average number of errors.

On the other hand, the transmission of the signal through a channel can introduce a huge average quantity of errors, where this big quantity of errors cannot be corrected by the channel code. Therefore, if the source packet mapping defines packets having a large variance, it can be that some of them have a very small number of errors in reception. This means that whereas a source packet mapping defining packets with a high variance can allow the decoding of some packets, a source packet mapping defining packets with a low variance cannot allow the decoding of any packet. Moreover, note that this last statement is also true if only a detection code is implemented and the system is only interested in recovering packets without errors.

To sum up, a source packet mapping defines the distribution between the bits of the packets and the bits mapped by a CSK symbol. Moreover, the source packet mapping determines the variance of the number of bit errors introduced by the transmission of a packet through a

channel. Therefore, if a channel code is implemented on the information bits and the source packet mapping is optimally chosen, the BER of the signal transmission can be minimized.

In this section, only the results are presented and commented. The method applied to calculate the variance of the number of bits errors of the packets defined by different source packet mapping variances can be found in annex I.2.

7.6.1. Results of the calculation of the variance of the number of bit errors of a packet defined by a CSK source packets mapping

In this section, the results of the calculations of the variance and the mean of the number of bit errors of the packets defined by a given CSK source packet mappings are presented. The variance and the mean values are calculated by applying their well known formulas, where the probabilities used in the calculations have been defined in annex I.2.

The main idea of this section is to compare for a given number of bits mapped by a single CSK symbol the variance of the number of bit errors of the packets defined by different CSK source packet mappings. Nevertheless, the only CSK source packet mappings analyzed in this research work are the configurations where all the packets have the same number of bits mapped by a single CSK symbol (see annex I.2). In other words, if a CSK symbol maps K bits, the analyzed CSK source packet mappings are the configurations where c packets are transmitted by several CSK symbols where K/c bits mapped by a CSK symbol belong to each packet. Note that K/c has to be an integer number. Therefore, since all the packets defined by a CSK source packet mapping present the same variance and the same mean of number of bits errors, only one curve if employed to represent the packets defined by the same CSK source packet mapping. Besides, all the possible combinations between the number of packets and the number of bits mapped by a CSK symbol belonging to the same packet are plotted on the same figure.

The figures of this subsection plot the mean and the variance of the number of bit errors of the packets defined by a given CSK source packet mapping as a function of the signal E_s/N_0, although their calculation is made by using the probabilities defined in annex I.2 which are expressed as a function of the CSK symbol probability of error. In fact, the different CSK symbol error probabilities have been changed by the E_s/N_0 values through equation (7-9). Therefore, the following figures are only valid for orthogonal CSK modulations. Note that the same process but for bi-orthogonal CSK modulations can be easily achieved using equation (7-12).

Moreover, the variance of the number of bit errors of a packet depends on the packet size as it has been said in annex I.2.2. The results presented here are calculated for a packet size equal to 1200 bits. This choice is justified by the new structure of the GALILEO E1 signal proposed by TAS-France. Moreover, the variance and the mean of the number of bit errors of packets having a size of 300 bits and 600 bits of size have also been analyzed.

The CSK source packet mappings analyzed for a packet of 1200 bits are the CSK source packet mappings which are possible for a CSK symbol mapping 4, 6, 8, 10 or 12 bits. The figures of the CSK symbol mapping 4 bits are shown below.

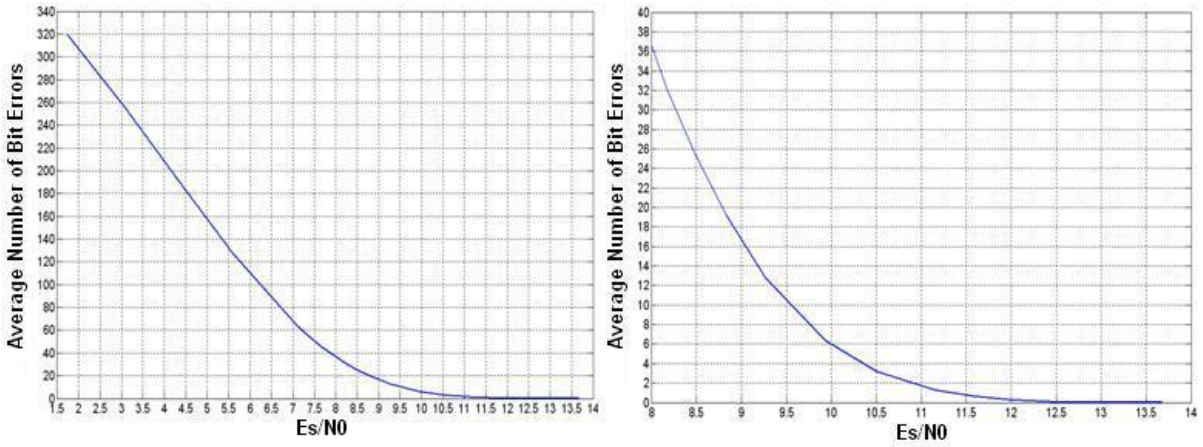

Figure 7-7: Average of the number of bit errors of a packet of 1200 bits transmitted by 16-CSK symbols with any possible CSK source packet mapping through an AWGN channel

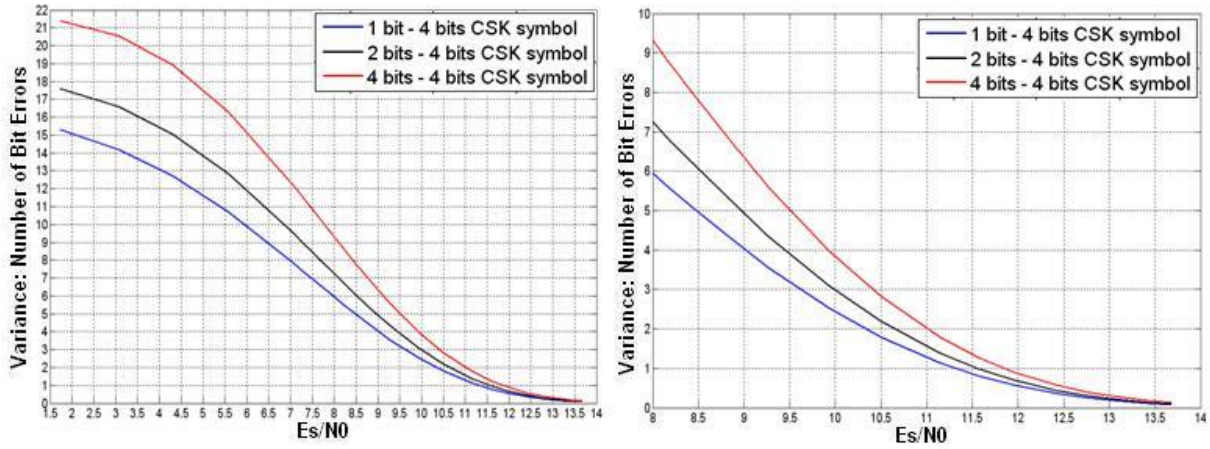

Figure 7-8: Variance of the number of bit errors of a packet of 1200 bits transmitted by 16-CSK symbols with any possible CSK source packet mapping through an AWGN channel

The first observation that can be made from Figure 7-7 is that the average of the number of bit errors of a packet transmitted by several CSK symbols each one mapping 4 bits is the same for all possible CSK source packet mappings. This observation verifies the statement proclaiming that for any possible CSK source packet mapping the BER of a packet without any implemented channel code is always the same.

The second observation is that the average of the numbers of bit errors of a packet presented in Figure 7-7 can also be calculated from equations (7-9) and (7-10): to multiple the BER associated to a given E_s/N_0 value by the number of bits of the packet. Therefore, since the average of the number of bit errors calculated using equations (7-9) and (7-10) is equal to the value presented in Figure 7-7 for any E_s/N_0 value, the bit error probabilities of the packets and the formulas applied to calculate these probabilities in annex I.2.1 are verified. Remember that annex I.2.1 probabilities, bit error probabilities of a packet, are also used to calculate the variance of the number of bit errors of a packet.

An important observation from Figure 7-8 is that the quantity of bits belonging to the same packet out of the total number of bits mapped by a single CSK symbol determines the variance of the number of bit errors of the packet. More specifically, it can be seen that decreasing the number of bits belonging to the same packet mapped by the same CSK symbol leads to a smaller the variance of the number of bit errors of the packet. For example, with an

E_s/N_0 equal to 4 dB, the variance of a packet of 1200 bits that have its bits transmitted by CSK symbols mapping 4 bits is about 19 bits when all the bits mapped by a CSK symbol belong to the same packet. But, the variance is about 13 bits when each bit mapped by a single CSK symbol belongs to a different packet. In fact, the first CSK source packet mapping requires an E_s/N_0 value of about 6.75 dB in order to obtain the same variance than the second CSK source packet mapping. However, for this E_s/N_0value, the average of the number of bit errors is much smaller for the second CSK source packet mapping. Therefore, the variance in percentage terms is not the same: for the second CSK source packet mapping, the variance of 13 bits is reached when the average is about 210 bits, and for the first CSK source packet mapping the variance of 13 bits is reached when the average is about 75 bits.

Additionally, note that the mean and variance values in Figure 7-7 and Figure 7-8 have been plotted until a CSK symbol probability of error equal to 0.5.

The figures of CSK symbols mapping 6, 8 and 10 bits are presented below.

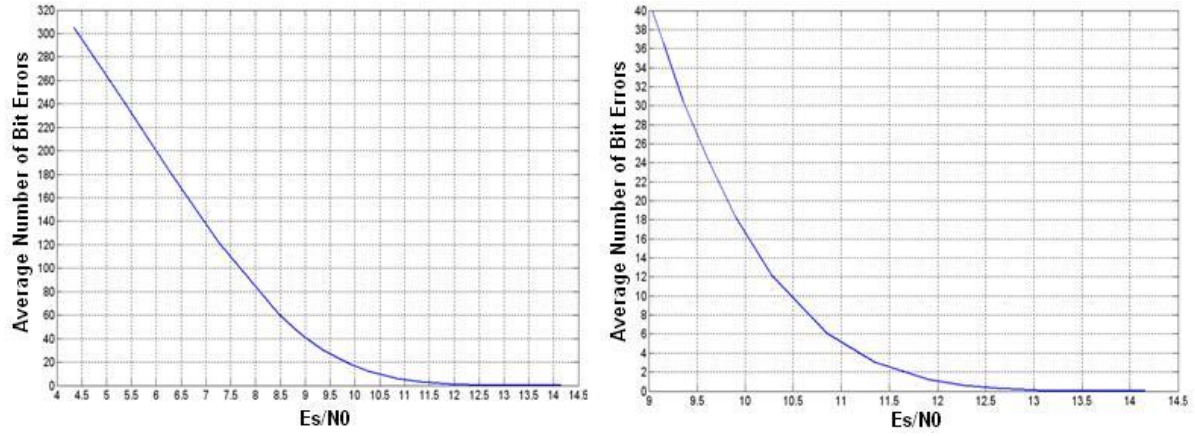

Figure 7-9: Average of the number of bit errors of a packet of 1200 bits transmitted by 64-CSK symbols with any possible CSK source packet mapping through an AWGN channel

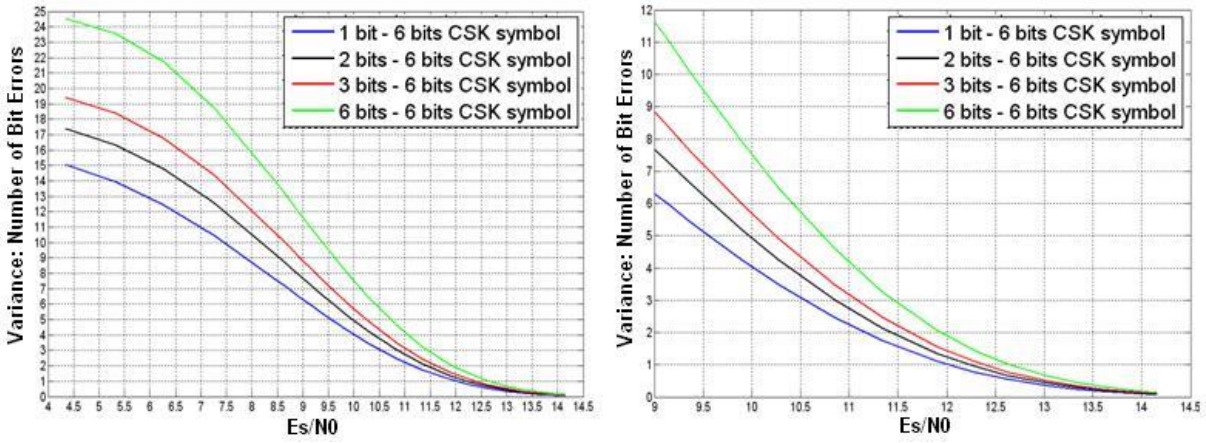

Figure 7-10: Variance of the number of bit errors of a packet of 1200 bits transmitted by 64-CSK symbols with any possible CSK source packet mapping through an AWGN channel

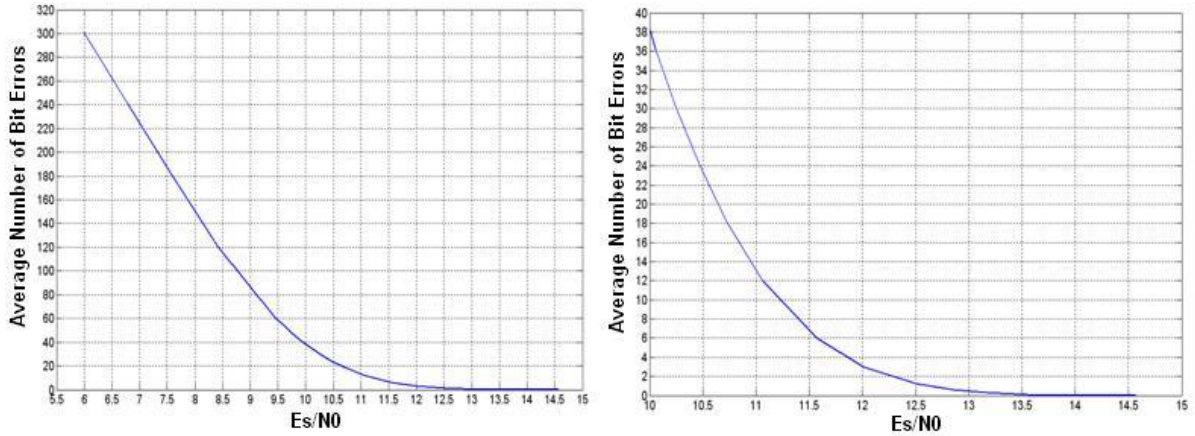

Figure 7-11: Average of the number of bit errors of a packet of 1200 bits transmitted by 256-CSK symbols with any possible CSK source packet mapping through an AWGN channel

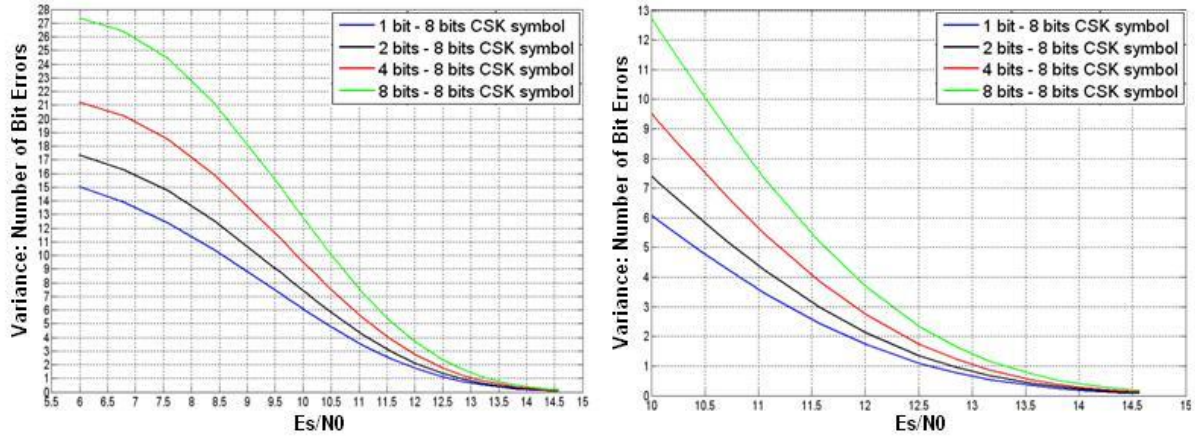

Figure 7-12: Variance of the number of bit errors of a packet of 1200 bits transmitted by 256-CSK symbols with any possible CSK source packet mapping through an AWGN channel

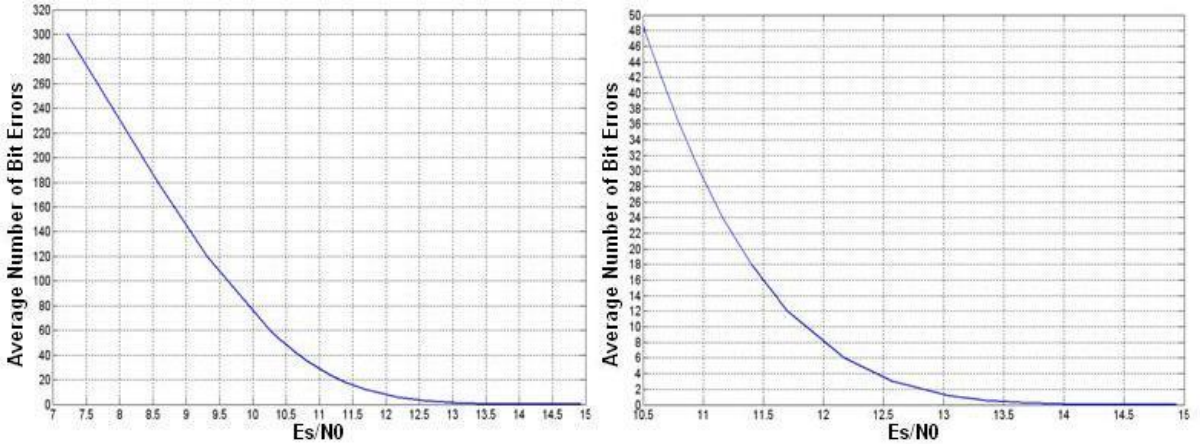

Figure 7-13: Average of the number of bit errors of a packet of 1200 bits transmitted by 1024-CSK symbols with any possible CSK source packet mapping through an AWGN channel

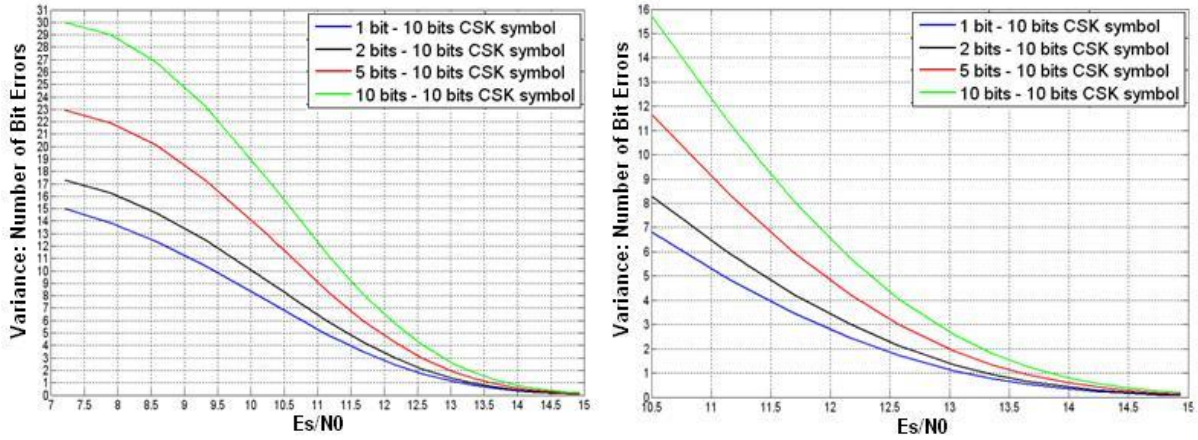

Figure 7-14: Variance of the number of bit errors of a packet of 1200 bits transmitted by 1024-CSK symbols with any possible CSK source packet mapping through an AWGN channel

First, from Figure 7-9 to Figure 7-14, it can be confirmed the first two conclusions extracted from Figure 7-7 and Figure 7-8.

Second, it can be observed that the increase of the number of bits belonging to the same packet out of the total bits mapped by a single CSK symbol leads to a larger variance of the number of bit errors of the packet.

In fact, it can be seen that for any number of bits mapped by a single CSK symbol, the variance of the number of bit errors of a packet with respect to the mean (or evolution) when each one of the bits mapped by the CSK symbol belong to a different packet is the same. Therefore, the variance of the number of bit errors of a packet with respect to the mean (or evolution) does not depend on the number of bits mapped by a single CSK symbol but rather on the number of bits belonging to the same packet out of the total number of bits mapped by the CSK symbol.

Nevertheless, note that the number of bits mapped by a single CSK symbol conditions the E_s/N_0 of the figures. In other words, the evolution of the variance of the number of bit errors as a function of the E_s/N_0 obtained by a packet having c bits transmitted by a CSK symbol mapping X bits is the same as the evolution of a packet having c bits transmitted by a CSK symbol mapping Y bits. However, the E_s/N_0 values of the CSK symbol mapping X bits case is equal to the addition between the E_s/N_0 of the CSK symbol mapping Y bits case and an offset constant for all the variance values.

The figures of a CSK symbol mapping 12 bits are presented below.

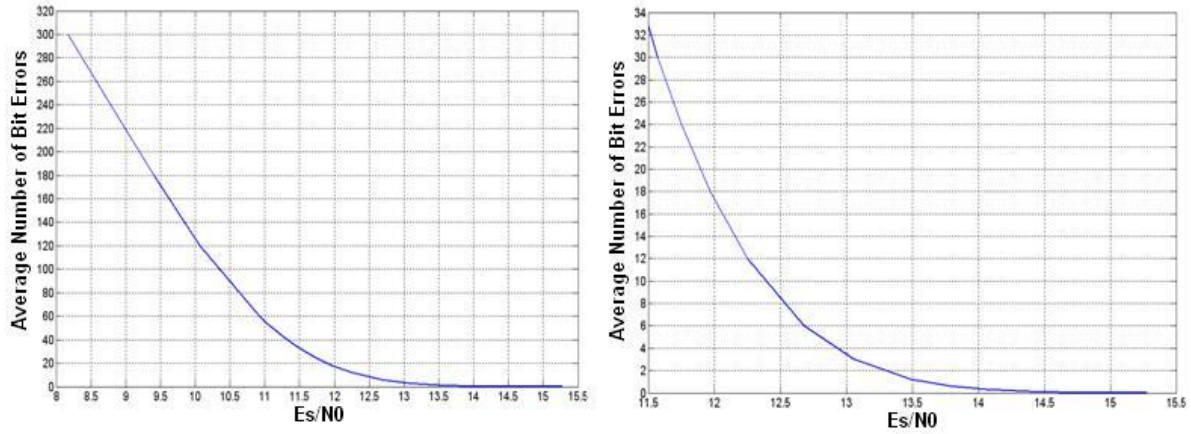

Figure 7-15: Average of the number of bit errors of a packet of 1200 bits transmitted by 4096-CSK symbols with any possible CSK source packet mapping through an AWGN channel

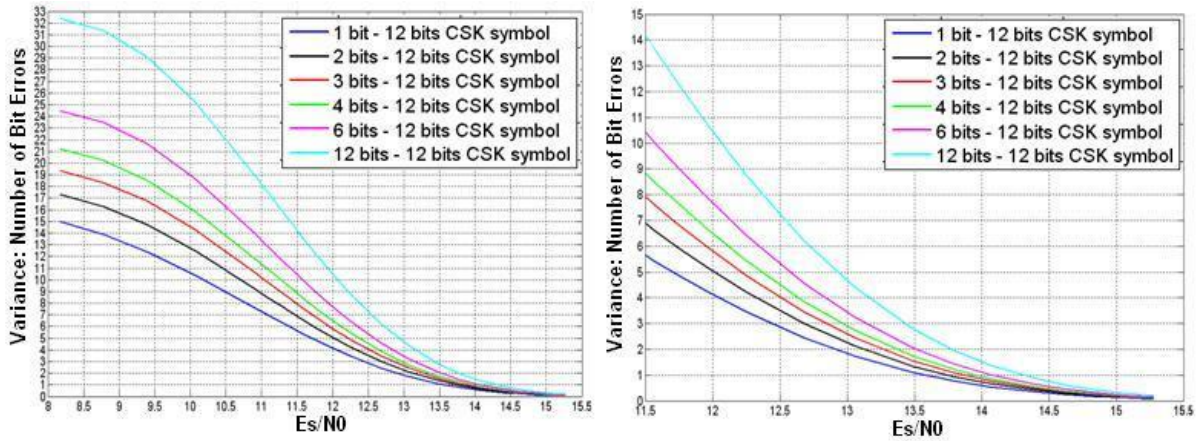

Figure 7-16: Variance of the number of bit errors of a packet of 1200 bits transmitted by 4096-CSK symbols with any possible CSK source packet mapping through an AWGN channel

The same conclusions from Figure 7-7 to Figure 7-14 can be extracted from Figure 7-15 and Figure 7-16.

In order to remark the dependence of the variance of the number of bit errors of a packet on the packet size, the variance of the number of bit errors of packets having a size of 300 bits and 600 bits when the CSK symbol maps 12 bits is shown below.

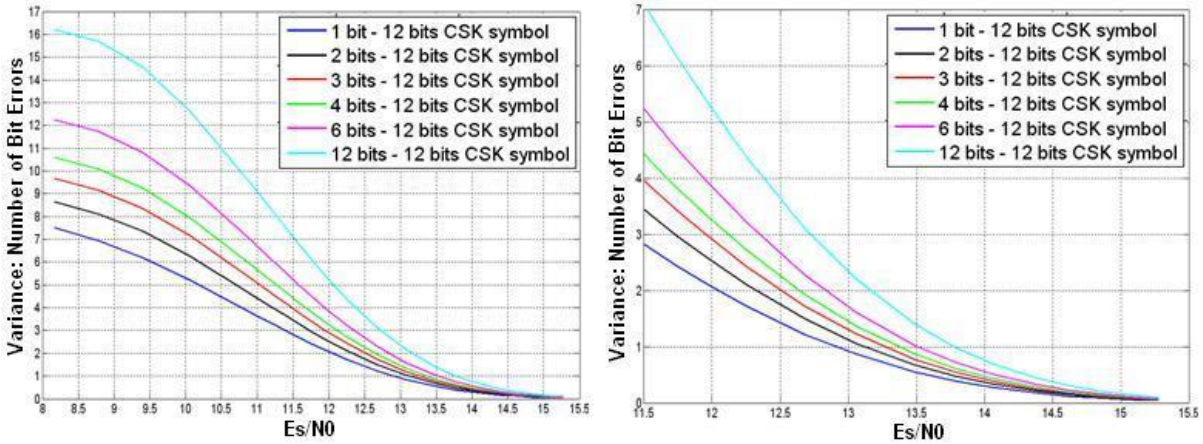

Figure 7-17: Variance of the number of bit errors of a packet of 300 bits transmitted by 4096-CSK symbols with any possible CSK source packet mapping through an AWGN channel

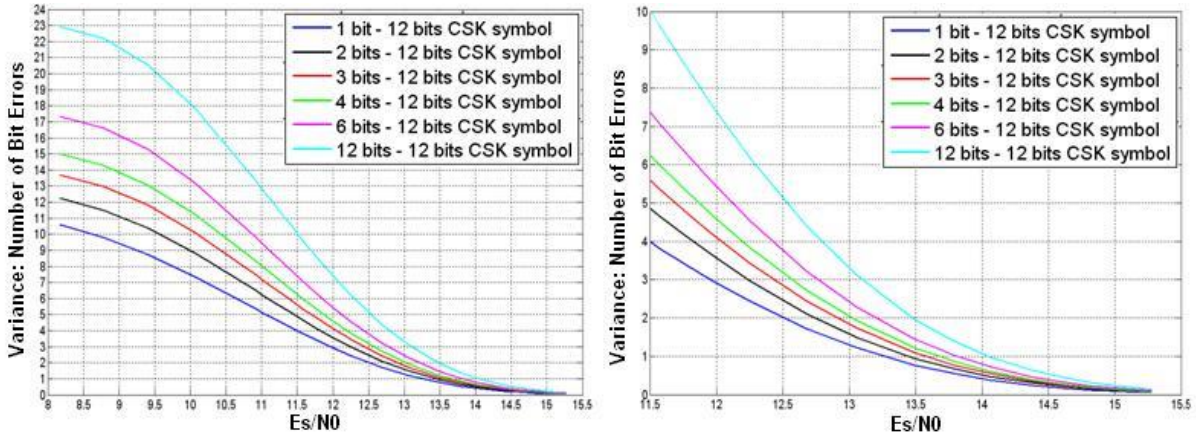

Figure 7-18: Variance of the number of bit errors of a packet of 600 bits transmitted by 4096-CSK symbols with any possible CSK source packet mapping through an AWGN channel

From Figure 7-16 to Figure 7-18, it can be observed that the variance of the number of bit errors of a packet decreases with the packet size. In fact, there is a linear relationship between the packet size and the variance values: each time that the size is reduced to half its original value, the variance is divided by $\sqrt{2}$.

Finally, the mean and the variance of the number of bit errors of packets of 1200 bits transmitted by CSK symbols mapping 12 bits when the CSK symbols probabilities of error go from 10^{-5} to 0.99 are presented. Nevertheless, remember that the CSK symbol error probability is not shown in these figures since it has been considered to be more interesting to display the variance and mean values as a function of the signal E_s/N_0.

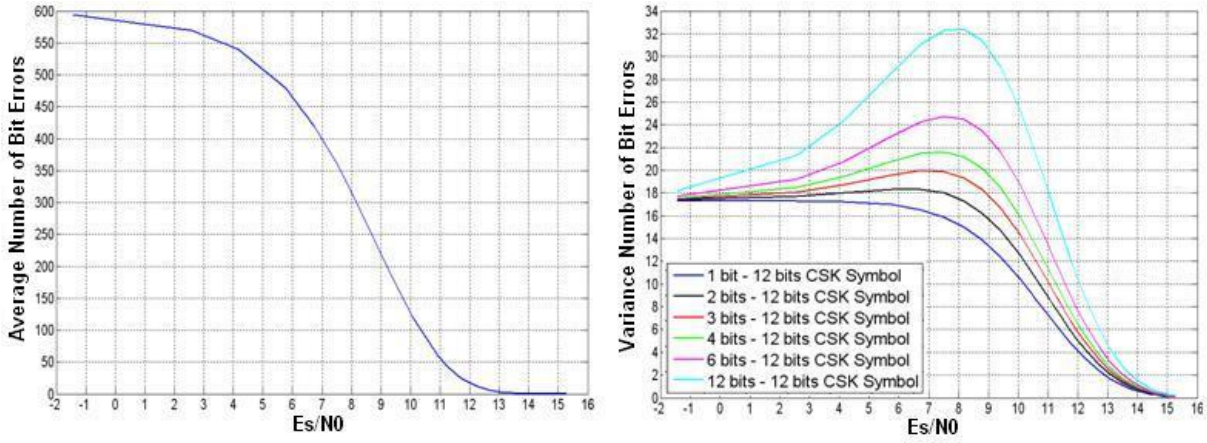

Figure 7-19: Average and variance of the number of bit errors of a packet of 1200 bits transmitted by 4096-CSK symbols with any possible CSK source packet mapping through an AWGN channel

From Figure 7-19, it can be observed that the additional E_s/N_0 values representing the new probabilities change completely the evolution of the curves representing the variance and the average. The curve representing the average is convex for E_s/N_0 values representing a P_M smaller than 0.5 and it is concave for E_s/N_0 values representing a P_M bigger than 0.5. Note that for a P_M almost equal to 1, the number of erroneous bits reaches half the size of the packet since the receiver is estimating all the CSK symbols randomly.

The curve representing the variance has its maximum about the E_s/N_0 value representing the P_M equal to 0.5. After reaching its maximum, all the variances of the packets defined by the different CSK source packet mappings decrease until converging to the same value. The

exception is the CSK source packet mapping where each one of the 12 bits mapped by a single CSK symbol belongs to a different packet since this curve always grows. Note that we are inspecting the curves backwards. The convergence of all the curves is logical since the estimation of the CSK symbol is completely random. Therefore, all the bits mapped by a CSK symbol are also random and thus have the same probability of being erroneously demodulated.

The results presented in this section are further exploited in section 7.8.

7.7. GALILEO E1 signal Code Shift Keying implementation

The implementation of the CSK signaling technique in the GALILEO E1 signal is not achieved by applying the typical CSK implementation process. In fact, there are some important differences which make the study and its posterior performance analysis a bit different.

The typical implementation of the CSK signaling technique in a telecommunications signal is achieved by following two main steps. First, the desired bit information rate is selected. Second, a trade-off between the fundamental code time period, which is the duration of a CSK symbol, and the number of M-ary orthogonal signals, which determines the number of bits mapped by a CSK symbol, is searched. This trade-off consists of obtaining the desired bit information rate whereas providing the lowest possible BER for a given E_b/N_0.

The implementation of the CSK signaling technique in the GALILEO E1 signal cannot follow any of the two previous steps. First, the fundamental code and its period time, which is equal to the duration of the traditional BPSK symbol, are fixed. Therefore, the bit information rate can only be partially controlled by the selection of the number of symbols, M. The selection of M is also limited since the fundamental code is the data channel PRN code of the GALILEO E1 signal which has already been defined. The data channel PRN code has a number of chips equal to 4092. This means that this code allows a maximum mapping of 11 bits for an orthogonal CSK symbol and 12 bits for a bi-orthogonal CSK symbol. Note that if the desired number of bits mapped by a CSK symbol is smaller than 11 or 12 bits, not all the possible circularly shifted versions of the PRN code are symbols.

Another consequence of the established definition of the fundamental code and of its time period is that the E_s of a CSK symbol of the GALILEO E1 signal is constant. This means that the number of bits mapped by a CSK symbol determines the E_b instead of conditioning the E_s. Therefore, since the E_b/N_0 will not be the same for a CSK symbol mapping X bits than for a CSK symbol mapping Y bits, where $X \neq Y$, a trade-off of the GALILEO E1 signal between the bit information rate and the desired BER has also to be found. In order to find this trade-off, the demodulation performance of the GALILEO E1 signal when the CSK signaling technique is implemented and no channel code is applied on the bits have to be calculated as a function of the signal C/N_0 or the message E_s/N_0.

The next figure illustrates the BER as a function of the part of the GALILEO E1 signal which implements the CSK signalling technique for different number of bits mapped by an orthogonal CSK symbol. The curves of the next figure have been calculated from equations (7-9), (7-10) and (6-13) when assuming a transmission through an AWGN channel and an ideal code delay and carrier phase tracking processes.

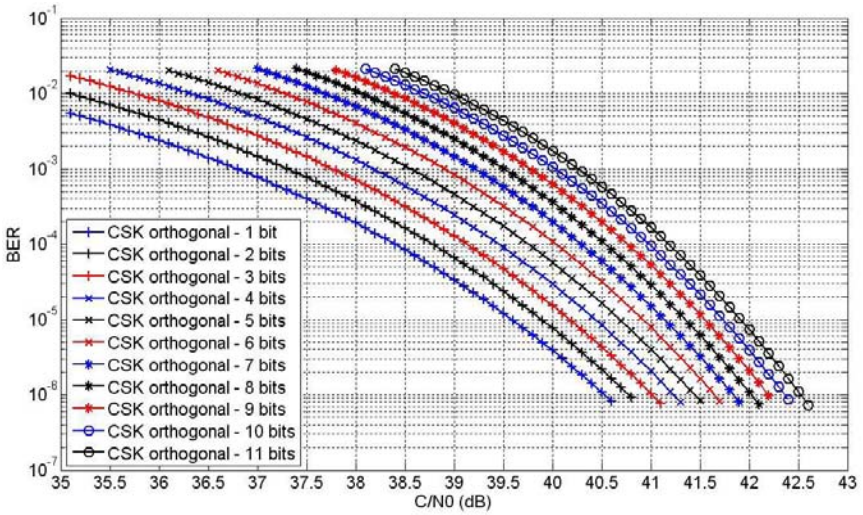

Figure 7-20: BER vs C/N_0 of the part of the GALILEO E1 signal implementing the CSK signalling technique for different number of bits mapped by an orthogonal CSK symbol

From Figure 7-20, it can be observed that due to the constant value of E_s, the more bits are mapped by a single CSK symbol, the worse demodulation performance in terms of BER is achieved by the CSK modulation in the GALILEO E1 signal. Nevertheless, studying the theoretical curves presented at [PROAKISc, 2001], it could be though that the figures should show results leading to the opposite conclusion. In fact, in the curves presented at [PROAKISc, 2001], the larger the value M is, the better the performance of the CSK modulation is. However, in the [PROAKISc, 2001] case, the E_b value is the same for all the different numbers of bits mapped by a CSK symbol which implies a different E_s for each different CSK mapping. But, in this case, the E_s value is fixed. And this means that the more bits are transmitted, the less E_b is carried by a single bit

Finally, the next figure illustrates the BER as a function of the part of the GALILEO E1 signal which implements the CSK signalling technique for different number of bits mapped by a bi-orthogonal CSK symbol. The curves of the next figure have been calculated from equations (7-12), (7-13) and (6-13) when assuming a transmission through an AWGN channel and an ideal code delay and carrier phase tracking processes.

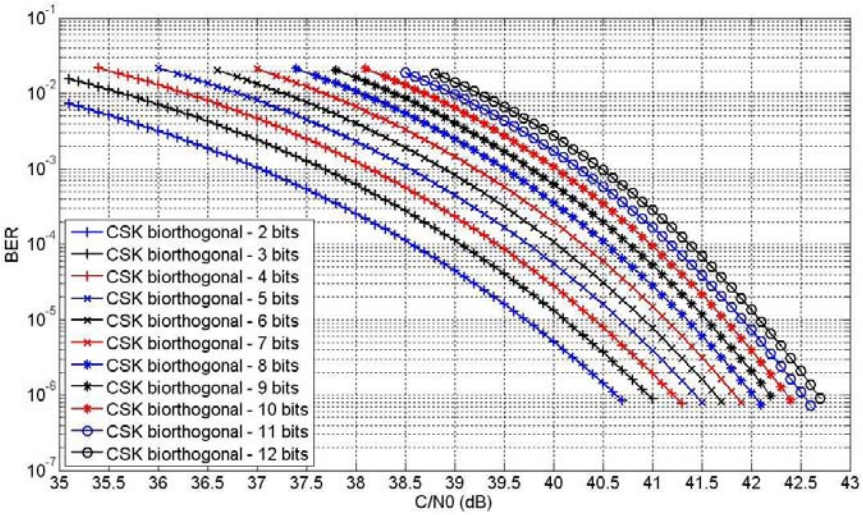

Figure 7-21: BER vs C/N0 for different CSK with polarity configurations of the GALILEO E1 signal

The same conclusions extracted from Figure 7-21 can be extracted from figure Figure 7-22.

7.8. Code Shift Keying demodulation performance for GALILEO E1 signal

In this section, the demodulation performance of the part of the proposed GALILEO E1 signal by TAS (see annex D.6) which implements the CSK signaling technique and which has a channel code applied on the information bits is presented. This section uses the likelihood expressions developed in section 7.5 and the results about the optimal CSK source packet mapping of section 7.6. Moreover, the results obtained in the previous section 7.7 are compared to the results obtained in this section.

The demodulation performance results are calculated for different CSK configurations and they are obtained through simulations. The definition of a CSK configuration is given next, before presenting the demodulation performance results of each CSK configuration.

7.8.1. CSK configurations

A CSK configuration is defined in this dissertation as the combination of 6 factors of the GALILEO E1 signal which condition the demodulation performance of the CSK modulation implemented in this signal. The 6 factors which condition the demodulation performance are the following: the number of CSK symbols transmitted each second, the number of bits mapped by a single CSK symbol, the CSK source packet mapping, the use of the CSK symbol polarity, the packet size and the channel code implemented on the word.

In this section, 12 different CSK configurations have been inspected. Each one of the studied CSK configuration has a unique combination of the number of bits mapped by a single CSK symbol, of the CSK source packet mapping and of the use of the CSK symbol polarity. The other three factors, the number of CSK symbols transmitted each second, the packet size and the channel code are the same for all the studied CSK configurations. The reason is that these last 3 factors are defined by the TAS-France proposal of the GALILEO E1 signal named ERIS. For a more detailed description of this proposal see annex D.6.

The 12 different CSK configurations are presented below.

- 12 bits mapped by a bi-orthogonal CSK symbol with each CSK bit belonging to a different packet.
- 12 bits mapped by a bi-orthogonal CSK symbol with all the CSK bits belonging to the same packet.
- 11 bits mapped by an orthogonal CSK symbol with each CSK bit belonging to a different packet.
- 11 bits mapped by an orthogonal CSK symbol with all the CSK bits belonging to the same packet.
- 8 bits mapped by a bi-orthogonal CSK symbol with each CSK bit belonging to a different packet.
- 8 bits mapped by a bi-orthogonal CSK symbol with all the CSK bits belonging to the same packet.
- 8 bits mapped by an orthogonal CSK symbol with each CSK bit belonging to a different packet.
- 8 bits mapped by an orthogonal CSK symbol with all the CSK bits belonging to the same packet.

- 6 bits mapped by a bi-orthogonal CSK symbol with each CSK bit belonging to a different packet.
- 6 bits mapped by a bi-orthogonal CSK symbol with all the CSK bits belonging to the same packet.
- 6 bits mapped by an orthogonal CSK symbol with each CSK bit belonging to a different packet.
- 6 bits mapped by an orthogonal CSK symbol with all the CSK bits belonging to the same packet.

Additionally, note that the number of transmitted packets is the same for all the CSK configurations having the same number of bits mapped by a single CSK symbol. In fact, the only difference between the CSK configurations which have a CSK symbol mapping the same number of bits is the instant where the transmitted packets are received. In the case where each bit mapped by a CSK symbol belongs to a different packet, the reception of all the packets is made at the same instant, whereas in the case where all the bits mapped by a CSK symbol belong to the same packet, the instant of reception of each packet is different. Therefore, for this latter CSK source packet mapping configuration, it is not necessary to wait until the reception of all the packets before using any of them. A figure presenting this difference is shown below.

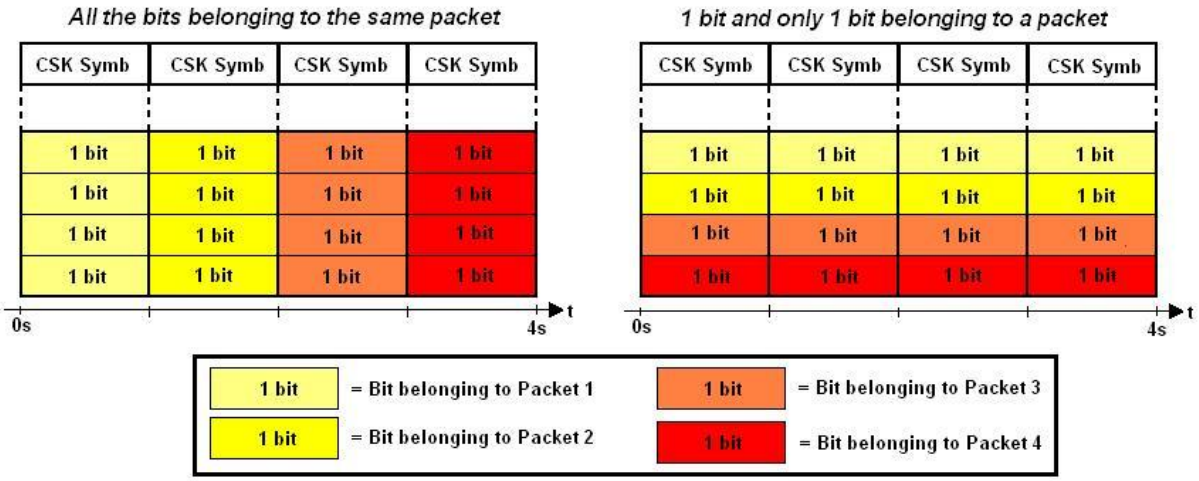

Figure 7-22: Two possible source packet mapping configurations with 4 packets of 4 bits transmitted by CSK symbols mapping 4 bits

The number of CSK symbols transmitted each second, the packet size and the channel code are presented next. These characteristics are defined by the TAS-France proposal named ERIS of the GALILEO E1 signal as it has been said before. The ERIS proposal conserves the symbol transmission rate of the original GALILEO E1 signal [ESA, 2008] but changes completely its message structure. More specifically and concerning the CSK implementation, the ERIS proposal reserves the subframe 4 of the new GALILEO E1 signal for the implementation of the CSK modulation. Therefore, since each bit spans over a complete data channel PRN code period, the subframe 4 should allow the transmission of a maximum number of 142 CSK symbols each second. Nevertheless, 4 bits of this subframe are used to configure subframe 5 which means that only a maximum of 138 CSK symbols are available. Moreover, note that, although 2 consecutive symbols could be the same circularly shifted version of the PRN code in order to create a longer CSK symbol with higher energy, E_s, it is

not possible to transmit more than 138 CSK symbols each second with the traditional CSK technique.

Finally, one of the possible packet structures defined in the ERIS proposal is the word structure employed by the GPS L1C signal subframe 2. This word structure is defined in section 2.5.2.2. This word structure determines a packet of 600 information bits that after applying a LDPC channel code results into 1200 code bits. This channel code is defined at [ARINC, 2006]. However, the number of bits to be transmitted which are necessary to completely receive the packets exceeds the selected packet size. This means that there are some extra transmitted bits without any assigned information. These bits are called padding and are not channel code protected. For example, if a CSK symbol mapping 10 bits is employed, 1380 bits will be received at the end of a frame transmission. This means an addition of 180 bits when all the bits transmitted inside the CSK symbols belong to the same packet. And, if the bits mapped by the CSK symbol do not belong to the same packet, but each bit belong to a different packet, 10 subframes are necessary to receive more than 1200 bits belonging to each of the individual packet. More exactly, 1380 bits will be accumulated and thus 180 padding bits are found again. A graphical example is shown in Figure 7-23.

Sub-Frame	T=0		T=1		T=2		T=3		T=4		T=5		T=6		T=7		T=8		T=9		Sum
	7		7		7		7		7		7		7		7		7		7		
11	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	1380
10	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	
9	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	
8	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	
7	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	
6	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	
5	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	
4	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	
3	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	
2	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	
1	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	138	52	
Sub-Frame	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	4	5	

Figure 7-23: CSK transmission with CSK symbols encoding 10 bits

Note that in Figure 7-23, subframe 5 already contains the bits reserved to indicate its configuration and which channel is active.

7.8.2. Results

In this section, the demodulation performance of each one of the previous defined CSK configurations when the signal is transmitted through an AWGN channel is presented. More specifically, these results are presented as the BER and WER as function of the signal C/N_0 at the receiver RF/IF block output.

The assumptions taken into account for this analysis are an ideal carrier phase tracking process, an ideal code delay tracking process and an ideal synchro-frame tracking process. Therefore, the results presented in this section inspect exclusively the demodulation performance of the CSK signaling technique. Nevertheless, these results can be used as a first approximation of the demodulation performance of the part of a possible future GALILEO E1 signal which implements the CSK signaling technique.

Additionally, for these simulations, an interleaver has been applied on the packets of all the CSK configurations where all the bits mapped by a CSK symbol belong to the same packet. In fact, these CSK configurations need an interleaver so that they do not have consecutive bits mapped by the same CSK symbol. The reason is that when one CSK symbol is erroneously

demodulated by a hard output demodulator, all the bits mapped by the CSK symbol are randomly estimated. The same phenomenon appears when a soft output demodulator is used; however, in this case, instead of providing random bits values, the soft output demodulator provides erroneous bits likelihood ratios. This means that the channel code implemented on the transmitted words is faced with burst of errors when a CSK symbol is erroneously demodulated. And, since the implemented LDPC channel codes have more difficulties to correct burst of errors than erroneous bits distributed uniformly over the word, the introduction of an interleaver is necessary in order to obtain the better demodulation performance of these types of CSK configurations. Note that the CSK configurations where all the bits mapped by a CSK symbol belong to different packets do not need an interleaver because they do not have consecutive bits mapped by the same CSK symbol. Nevertheless, when a signal is transmitted through a mobile channel, the application of an interleaver is mandatory for any CSK configuration in order to avoid the burst of errors introduced by the obstacles of the channel. Therefore, if the signal is conceived to be transmitted through a mobile channel, the introduction of an interleaver is required for any CSK configuration. This means that this drawback should not be taken into account when the signal is transmitted through a mobile channel.

Finally, 10 different figures are presented in this section. First, the CSK configurations are separated into different groups, where each group consists of all the CSK configurations having the same number of bits mapped by a single CSK symbol. For each group, two figures are displayed, one showing the BER and the other presenting the WER. Second, two additional figures are presented. These figures show the demodulation performance of the best configuration of each one of the previous defined groups. This comparison is made in order to decide which configuration has the better trade-off between the bit transmission rate and the signal C/N_0 at the receiver RF/IF block output necessary to demodulate the information.

The next figure illustrates the BER as a function of the signal C/N_0 of all the CSK configurations where each single CSK symbol maps 6 bits.

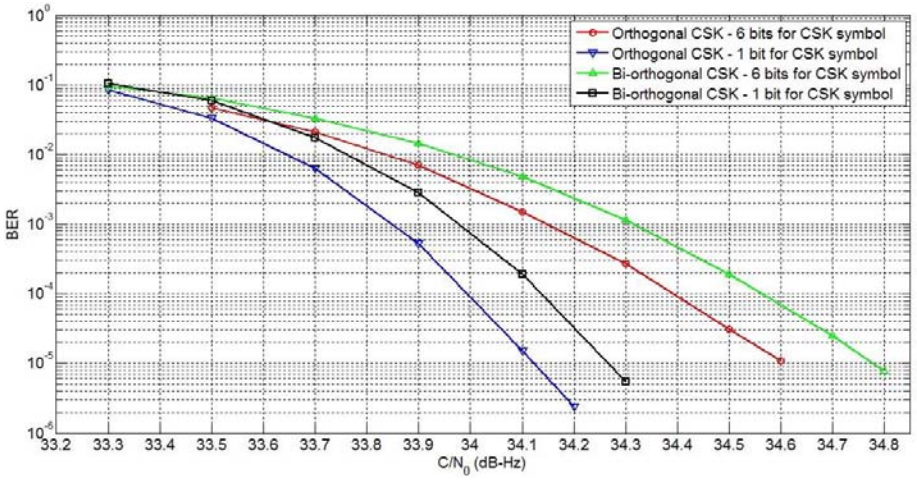

Figure 7-24: BER vs C/N_0 of different CSK configurations where each CSK symbol maps 6 bits when the signal is transmitted through an AWGN channel

From Figure 7-24, it can be observed that the CSK configurations where the bits mapped by a CSK symbol belong to different packets outperform the CSK configurations where the bits mapped by a CSK symbol belong to the same packet. More specifically, it can be seen that this difference is about 0.5dB for a BER equal to 10^{-5}. Besides, this difference should increase with C/N_0. Therefore, this difference of demodulation performance verifies the theoretical

results extracted in the CSK source packet mapping analysis, section 7.6, when it was stated that the optimal CSK source packet mapping is the CSK source packet mapping where all the bits mapped by a CSK symbol belong to different packets.

Moreover, from Figure 7-24, it can be observed that although the demodulation performance of the hard output modulators is the same for orthogonal and bi-orthogonal CSK modulations, and that although the variance of the bits likelihood ratios is almost the same for orthogonal and bi-orthogonal CSK modulations, the orthogonal CSK configurations slightly outperforms the bi-orthogonal CSK configurations. More specifically, the bi-orthogonal CSK configurations implemented in this section where the CSK symbol maps 6 bits can require between 0.1 and 0.2 extra dB to obtain the same BER values than the orthogonal CSK configurations implemented in this section where the CSK symbol maps 6 bits. This difference of performance can be caused by the non-optimal expression of the bi-orthogonal CSK configurations used to calculate the likelihood ratios of the bits mapped by the shift of the fundamental code. Remember that the likelihood ratios of the bits mapped by the shift of the fundamental code are calculated by using the estimated probability of the polarity of the bi-orthogonal CSK symbol. Nevertheless, this difference is really small and thus, the choice of the CSK configuration to implement should depend on the complexity of the demodulation process instead of on this difference. Remember that the bi-orthogonal CSK configurations require half the number of correlators needed for the orthogonal CSK configurations or they require inspecting half the number of points of the FFT transformation. See section 7.3 where the demodulation methods are explained.

Additionally, comparing Figure 7-24 to Figure 7-20 and Figure 7-21, it can be observed that, when 6 bits are mapped by a single CSK symbol, the demodulation performance gain in terms of C/N_0 introduced by the implemented LDPC channel code with respect to CSK configurations not having any channel code is very significant. More specifically, for a BER equal to 10^{-5}, the provided gain is about 6.8 dB when the orthogonal CSK configuration is implemented and is about 6.65 dB when the bi-orthogonal CSK configuration is implemented. This gain can be compared to the gain introduced by the application of the same LDPC code on a BPSK signal which does not implement the CSK signalling technique; this gain is about 7.05 dB.

The next figure illustrates the WER as a function of the signal C/N_0 of all the CSK configurations where each single CSK symbol maps 6 bits.

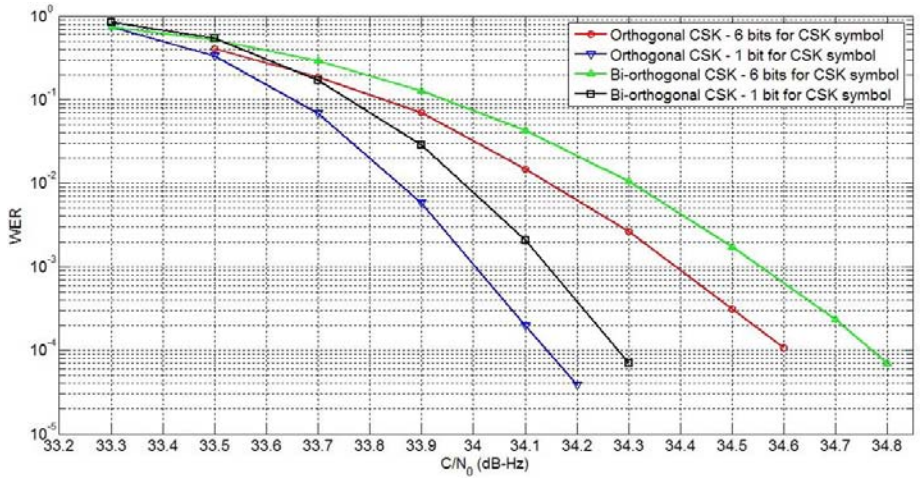

Figure 7-25: WER vs C/N_0 of different CSK configurations where each CSK symbol maps 6 bits when the signal is transmitted through an AWGN channel

The same observations extracted from Figure 7-24 can be extracted from Figure 7-25. First, the CSK source packet mapping results of section 7.6 are verified. The gain in terms of C/N_0 between the CSK configurations where all the bits mapped by a CSK symbol belong to different packets and the CSK configurations where all the bits mapped by a CSK symbol belong to the same packet are about 0.4 dB for a WER equal to 10^{-3} and about 0.45/0.5 dB for a WER equal to 10^{-4}. Second, the orthogonal CSK configurations slightly outperform the bi-orthogonal CSK configurations.

The next figure illustrates the BER as a function of the signal C/N_0 of all the CSK configurations where each single CSK symbol maps 8 bits.

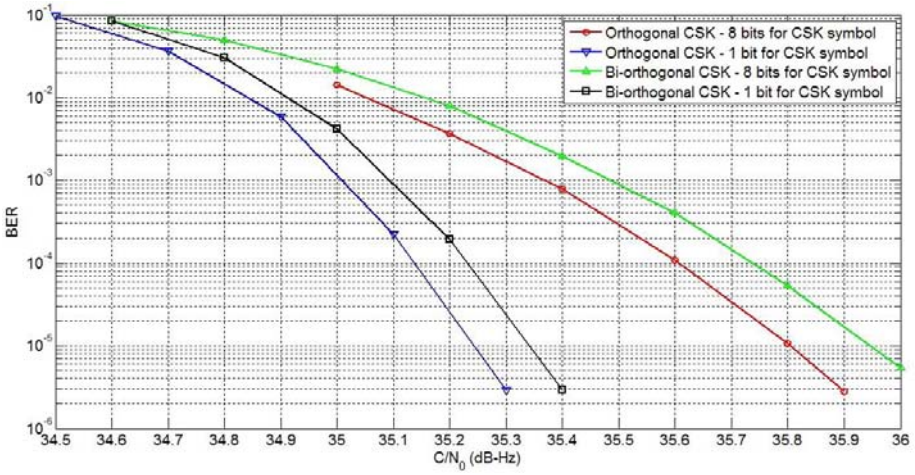

Figure 7-26: BER vs C/N_0 of different CSK configurations where each CSK symbol maps 8 bits when the signal is transmitted through an AWGN channel

From Figure 7-26, it can be verified again that the CSK configurations where all the bits mapped by a CSK symbol belong to different packets outperform the CSK configurations where all the bits mapped by a CSK symbol belong to the same packet. Moreover, Figure 7-26 also confirms another result presented in the CSK source packet mapping section. This result is that the difference of performance between the two previous commented CSK configurations grows with the quantity of bits mapped by a CSK symbol. In this case, the difference is about 0.55 dB for a BER equal to 10^{-5}.

Moreover, from Figure 7-26, it can also be observed that the orthogonal CSK configurations continue to outperform the bi-orthogonal CSK configuration. Nevertheless, the 8 bits case difference is decreased with respect to the 6 bits case difference. In Figure 7-26, this difference hardly exceeds 0.1/0.15 dB.

Additionally, comparing Figure 7-26 to Figure 7-20 and Figure 7-21, it can be observed that, when 8 bits are mapped by a single CSK symbol, the demodulation performance gain in terms of C/N_0 introduced by the implemented LDPC channel code is very significant. More specifically, for a BER equal to 10^{-5}, the provided gain is about 6.05 dB when the orthogonal CSK configuration is implemented and is about 5.95 dB when the bi-orthogonal CSK configuration is implemented. Moreover, from comparing these coding gains with those of the 6 bits case, it can be observed that the coding gain decreases with the size of the CSK alphabet. The reason of this farther improvement can be the larger variance of the bits likelihood ratios obtained by the 6 bits case with respect to the 8 bits case as it has been since in section 7.5.3, as well as the farther improvement with the increase of bits mapped by a CSK symbol of the demodulation performance for a hard demodulation than for a soft demodulation.

The next figure illustrates the WER as a function of the signal C/N_0 of all the CSK configurations where each single CSK symbol maps 8 bits.

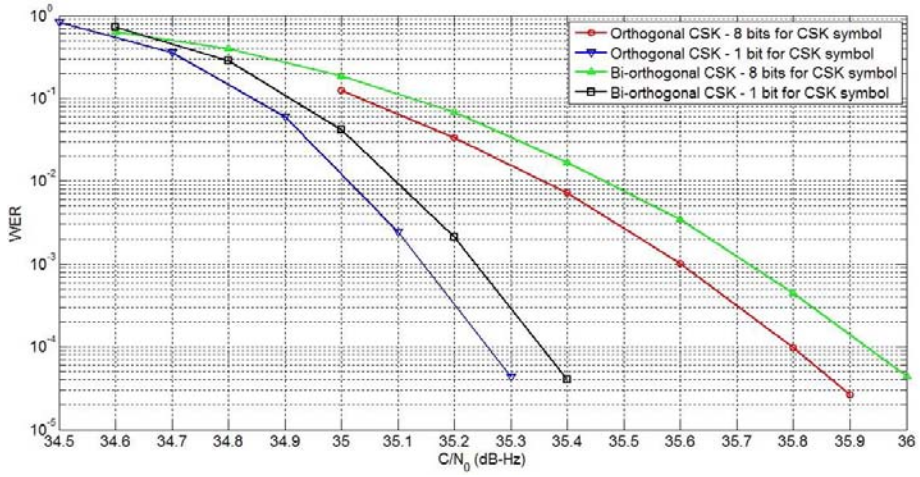

Figure 7-27: WER vs C/N_0 of different CSK configurations where each CSK symbol maps 8 bits when the signal is transmitted through an AWGN channel

The same observations extracted from Figure 7-26 can be extracted from Figure 7-27. In this case, the gain in terms of C/N_0 between the CSK configurations where all the bits mapped by a CSK symbol belong to different packets and the CSK configurations where all the bits mapped by a CSK symbol belong to the same packet are about 0.45 dB for a WER equal to 10^{-3} and about 0.55 dB for a WER equal to 10^{-4}.

The next figure illustrates the BER as a function of the signal C/N_0 of all the orthogonal CSK configurations where each single orthogonal CSK symbol maps 11 bits. The analysis of these configurations has been made in order to obtain the demodulation performance of the orthogonal CSK configuration which provides the higher information transmission rate for the proposed GALILEO E1 signal. Remember that the GALILEO E1 signal has 4092 chips; and this means that a maximum of 11 bits can be mapped by an orthogonal CSK symbol.

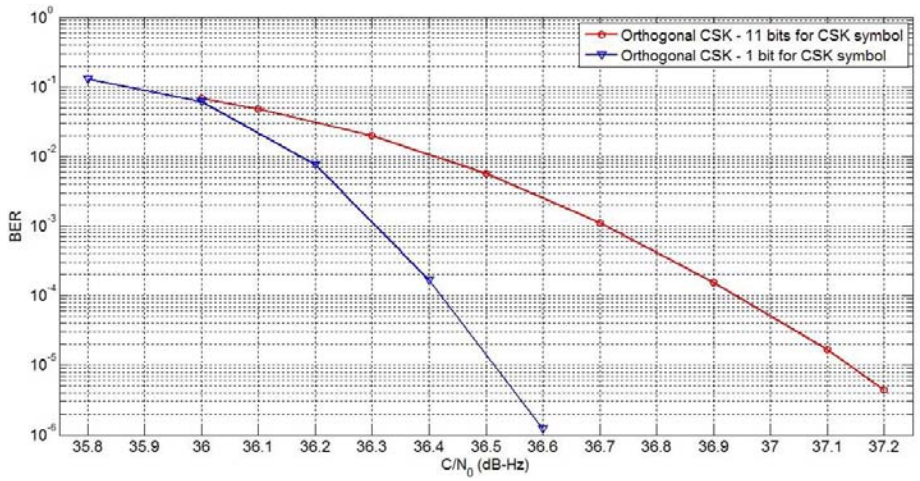

Figure 7-28: BER vs C/N_0 of different CSK configurations where each CSK symbol maps 11 bits when the signal is transmitted through an AWGN channel

From Figure 7-28, it can be verified again that a CSK configuration which has all the bits mapped by a CSK symbol belonging to different packets outperforms a CSK configuration

which has all the bits mapped by a CSK symbol belonging to the same packet. In this case, the difference is about 0.6 dB for a BER equal to 10^{-5}.

Moreover, it is confirmed from this figure, Figure 7-24 and Figure 7-26 that the previous difference grows with the increase of bits mapped by a CSK symbol. The reason can be that whereas the standard deviation of the number of bit errors of a packet defined by a CSK configuration which has all the bits mapped by a CSK symbol belonging to different packets remains constant for any number of mapped bits, the standard deviation of the number of bit errors of a packet defined by a CSK configuration which has all the mapped by a CSK symbol belonging to the same packet grows with the number of mapped bits. This phenomenon has been observed in section 7.6.

Additionally, comparing Figure 7-28 to Figure 7-20, it can be observed that, when 11 bits are mapped by a single CSK symbol, the demodulation performance gain in terms of C/N_0 introduced by the implemented LDPC channel code is very significant. More specifically, for a BER equal to 10^{-5}, the provided gain is about 5.4 dB when the orthogonal CSK configuration is implemented. Therefore, it is verified again that the coding gain between the CSK configurations implementing a LDPC channel code and the CSK configuration not implementing a channel code grows with the decrease of the number of bits mapped by a CSK symbol.

The next figure illustrates the WER as a function of the signal C/N_0 of all the CSK configurations where each single CSK symbol maps 11 bits.

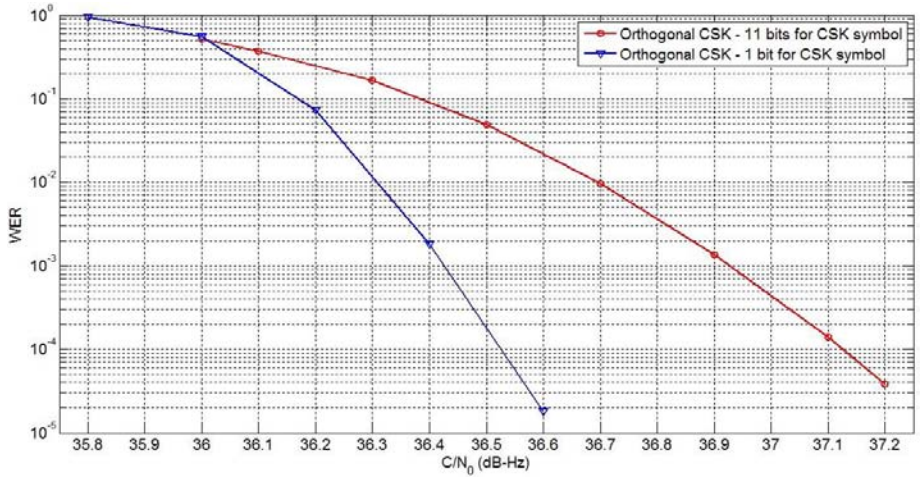

Figure 7-29: WER vs C/N_0 of different CSK configurations where each CSK symbol maps 11 bits when the signal is transmitted through an AWGN channel

The same observations extracted from Figure 7-28 can be extracted from Figure 7-29. In this case, the gain in terms of C/N_0 between the CSK configuration where all the bits mapped by a CSK symbol belong to different packets and the CSK configuration where all the bits mapped by a CSK symbol belong to the same packet are about 0.5 dB for a WER equal to 10^{-3} and about 0.6 dB for a WER equal to 10^{-4}.

The next figure illustrates the BER as a function of the signal C/N_0 of all the bi-orthogonal CSK configurations where each single bi-orthogonal CSK symbol maps 12 bits. The analysis of these configurations has been made in order to obtain the demodulation performance of the bi-orthogonal CSK configuration which provides the higher information transmission rate for the proposed GALILEO E1 signal. Remember that the GALILEO E1 signal has 4092 chips; and this means that a maximum of 12 bits can be mapped by a bi-orthogonal CSK symbol.

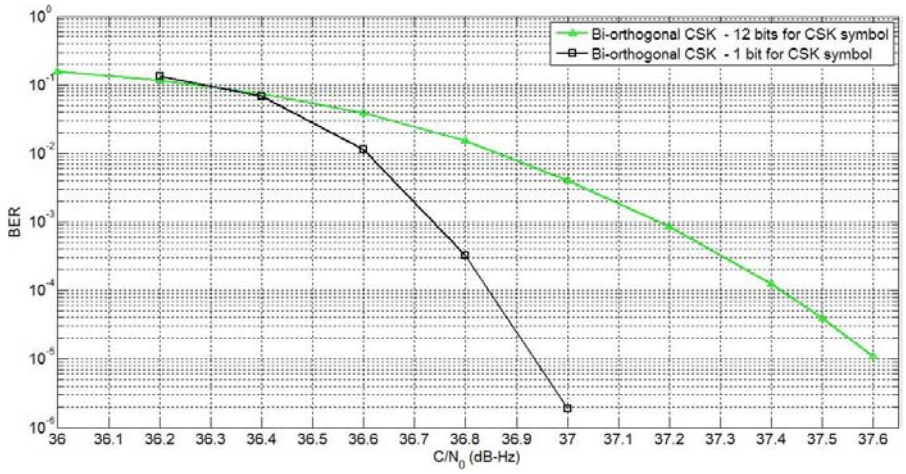

Figure 7-30: BER vs C/N_0 of different CSK configurations where each CSK symbol maps 12 bits when the signal is transmitted through an AWGN channel

The same observations extracted from the other BER figures can be seen in Figure 7-30, where the difference between the two bi-orthogonal CSK configurations shown is about 0.65 dB for a BER equal to 10^{-5}.

The next figure illustrates the WER as a function of the signal C/N_0 of all the bi-orthogonal CSK configurations where each single bi-orthogonal CSK symbol maps 12 bits.

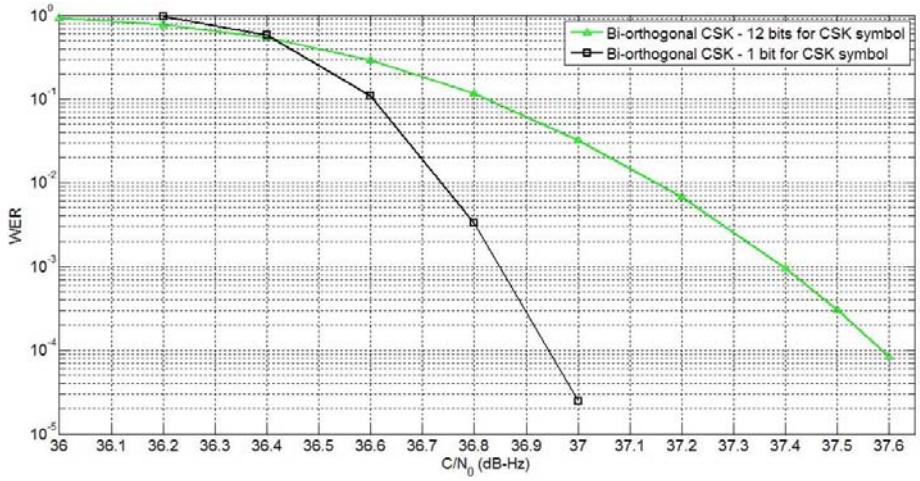

Figure 7-31: WER vs C/N_0 of different CSK configurations where each CSK symbol maps 12 bits when the signal is transmitted through an AWGN channel

The same observations extracted from Figure 7-30 can be extracted from Figure 7-31. In this case, the gain in terms of C/N_0 between the CSK configuration where all the bits mapped by a CSK symbol belong to different packets and the CSK configuration where all the bits mapped by a CSK symbol belong to the same packet are about 0.55 dB for a WER equal to 10^{-3} and about 0.65 dB for a WER equal to 10^{-4}.

Once the performance of each possible CSK configuration has been showed by groups of CSK configurations which have the same number of bits mapped by a single CSK symbol, the CSK configurations having the best demodulation performance among all the CSK configurations having the same number bits mapped by a symbol CSK symbol are displayed together. These CSK configurations are the orthogonal CSK configuration with all the bits mapped by a CSK symbol belonging to different packets. Nevertheless, for the 12 bits case

the bi-orthogonal CSK configuration has been plotted since the orthogonal CSK configuration cannot exist for the proposed GALILEO E1 signal structure.

The next figure shows the BER as a function of the C/N_0 of these CSK configurations.

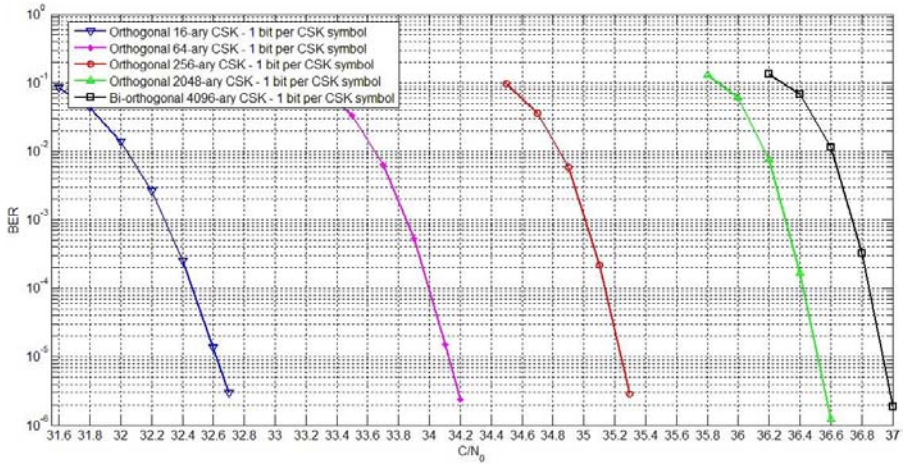

Figure 7-32: BER vs C/N_0 of the best GALILEO E1 signal CSK configurations

From Figure 7-32 and Figure 7-20 and Figure 7-21, it can be seen that the differences in terms of signal C/N_0 at the receiver RF/IF block output between the different CSK configurations when the LDPC channel code is implemented are larger than the differences between these same CSK configurations in the absence of channel code. This increase of the difference of C/N_0 is expected from the results obtained in figures Figure 7-24, Figure 7-26, Figure 7-28 and Figure 7-30.

Moreover, an additional curve of a CSK configuration mapping 4 bits per symbol has been introduced in Figure 7-32 in order to highlight that the cost in C/N_0 terms in order to increase the number of bits mapped per CSK symbol depends on the original number of bits: an increase of 2 bits mapped by a CSK symbol requires a lower increase of C/N_0 when passing from 6 bits to 8 than when passing from 4 bits to 6.

The differences of signal C/N_0 at the receiver RF/IF output block between the CSK configurations shown in Figure 7-32 are quite important. Therefore, a future GNSS signal final design implementing the CSK technique should determine how much extra information transmission rate really needs in order to design CSK symbols mapping only the required number of bits. In fact, the levels of C/N_0 which provide satisfactory BER values when using CSK symbols mapping only 6 bits are already very high. Consequently, only in the best environments conditions, the extra information carried by the CSK information could be received.

The next figure shows the WER as a function of the C/N_0 of the best GALILEO E1 signal CSK configurations. This figure provides a summary of the WER values obtained by each CSK configuration.

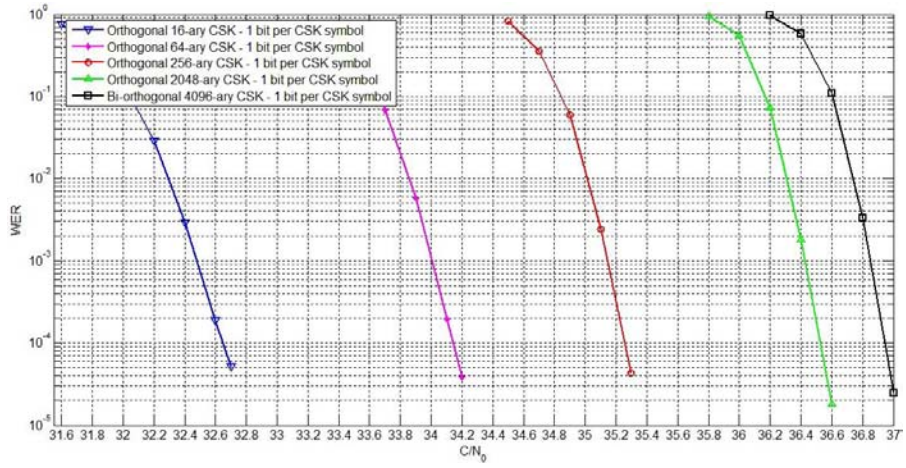

Figure 7-33: WER of the best GALILEO E1 signal CSK configurations

The same conclusions extracted from Figure 7-32 can be extracted from Figure 7-33.

7.9. Code Shift Keying Advantages

The main advantage of the code shift keying technique is clearly the increase of the signal information data rate. This increase of data rate is not easily achieved by the use of a BPSK modulated signal with a higher symbol rate because this will imply two possible undesired scenarios.

First, the increase of the symbol rate when the chip rate is maintained constant implies a shorter PRN code with its consequent loss of performance: isolation and near/far effect. Second, the increase of the symbol rate when the chip rate is increased in order to maintain the same PRN code length implies a wideband receiver with the consequent increase of number of operations and a larger spectral occupation. Therefore, the use of a CSK modulation in order to increase the bit rate is an attractive option to be explored. Moreover, this increase of data rate can be applied to already existing signals, in their full extend or only partially.

This increase of the data information rate can be used in two different manners. The first one is to use the extra rate in order to broadcast information for new commercial services. In this case, the introduction of the CSK technique provides a commercial gain but not a technical one except if new pseudo-range measurements corrections are sent.

The second one consists in broadcasting redundant information such as the ephemeris. In this case, a satellite could broadcast its own future ephemeris in order to allow the GNSS receivers to store and use them in the future instead of obliging the receivers to demodulate the navigation message in worse environment conditions where its correct reception could be impossible. Moreover, the ephemeris of other satellites could also be broadcasted by other satellites in order to benefit from their punctual good reception conditions and thus in order to decrease the acquisition and tracking C/N_0 threshold of the satellites with bad line-of-sight by means of wipe-off techniques.

7.10. Code Shift Keying Drawbacks

In this section, the drawbacks of the implementation of the Code Shift Keying technique onto a navigation signal are presented. More specifically, the drawbacks presented are the possible

decrease of the acquisition and tracking processes performance due to the lack of synchronization between the pilot and data PRN codes. Finally, the impact of the CSK on the calculation of the pseudo-range measurement and why it is not a main drawback is presented.

A brief explication of how the lack of channel PRN codes synchronization affects these processes is given next. The typical structure of the new navigation signals consist of two channels transmitted together in phase or in phase-quadrature. Each channel carries a spreading code which allows the receiver to separate them after their reception. The channels are called pilot channel and data channel whereas the spreading code is called PRN code. The method employed to separate the channels, which is the same method used to calculate the pseudo-range between the receiver and the broadcasting satellite, consists in computing the correlation between the received code and a generated local replica. In order to successfully exploit the correlation, the receiver must know the phasing of the PRN codes with respect to the other elements of the signal. The processes responsible of the synchronization of the local replica with the received code are the acquisition and tracking processes.

These processes can use the data channel PRN code and pilot channel PRN code inherent synchronization in order to enhance their estimation performance. If both PRN codes are not synchronized but the delay between them is constant, the same enhancement could be applied due to the PRN codes transmission periodicity. The only difference between two synchronized PRN codes and two PRN codes with a fixed delay is that one local replica has to be delayed in relation to the other one whereas in the non-delay case both replicas are also synchronized. However, the introduction of the CSK technique modifies the PRN code delay for every transmitted symbol. Therefore, both codes cannot be used simultaneously since the receiver does not know what the current delay between the data and pilot PRN codes is. Moreover, since the data channel has a variable delay which is not known in advance, the delay found by the n^{th} bit cannot be used as a starting value to estimate the delay of $(n+1)^{th}$ bit. Therefore, the tracking process cannot be made using the data channel signal unless the data information is known in advance.

In this section, first the loss of acquisition performance introduced by the CSK technique is analyzed, second, some preliminary conclusions are made about the possible loss of tracking performance, and third, the impact of the CSK technique on the pseudo-range calculation is explained.

7.10.1. GALILEO E1 signal acquisition performance

In this section, the acquisition performance of the GALILEO E1 signal is calculated. A definition of the acquisition process has been given in annex C.7. This acquisition performance is calculated using different channel options and configurations. More specifically, the performance is computed when the receiver only uses the pilot channels to acquire the signal, when the receiver uses the pilot and the data channel to acquire the signal and the CSK technique is not implemented, and when the receiver uses the pilot and the data channels to acquire the signal and the CSK technique is implemented. The main differences between the acquisition process conducted over a typical navigation signal and the acquisition process conducted over a signal implementing the CSK technique can be found in annex C.7.2.2.

The acquisition performance of the GALILEO E1 signal is calculated by using the acquisition parameters presented and described in annex C.7.2.1. Moreover, this performance is calculated for the optimal mapping between the CSK symbols and the shifts of the

fundamental code which reduces the CSK performance loss over the acquisition process. This optimal mapping is presented in annex C.7.2.3.

These results of the acquisition performance of the GALILEO E1 signal are expressed as a function of the signal C/N_0 at the receiver antenna output, but before presenting them, the simulator scheme is explained and its parameters are defined.

7.10.1.1. Simulator

In this section, only the simulator parameters are described in order to allow the reader to obtain a fast interpretation of the results presented in section 7.10.1.2. Nevertheless, the description of the simulator can be found in annex C.7.2.4.

The simulator parameters are equal to the acquisition parameters defined in annex C.7.2.1. Nevertheless, a quick summary is made in this section in order to outline their values. The coherent integration time (T_p) is equal to 4ms and the non-coherent integration time is extended to 100ms (MT_p), which means that 25 non-coherent integrations are used to determine if the signal is acquired. This number is called M. The total number of bins and the false alarm probability are calculated as expressed in annex C.7.2.1.1 and C.7.2.1.2. The time required to verify that the inspected bin really contains the signal is 1 second (TFAF).

The option representing the CSK data has been configured as follows: for a transmitted page part of 250 symbols, the first 60 symbols are traditional BPSK symbols and the remaining 190 symbols are CSK symbols. The number of bits transmitted by each CSK symbol is irrelevant from the simulator point of view; therefore this parameter has not been determined.

Finally, the number of correlators used to acquire the signal has been set to 2000.

7.10.1.2. Results

The results shown in this section represent the GALILEO E1 OS signal acquisition performance obtained in the worst possible scenario. This means that we assume that the true signal carrier is always located between two frequency bins and that the true signal PRN code is also located between two time bins. Therefore, the maximum signal attenuations derived in annex C.7.2.1.1 always affect the bins where the signal should be acquired.

Moreover, the results presented in this section, in addition to being calculated in the worst possible scenario, are conditioned by two acquisition requirements. These requirements condition the frequency and time uncertainties of the signal to be acquired.

The frequency requirement is imposed by the FLL block. The FLL will use the frequency estimated during the acquisition process as initial value to track the signal carrier. However, this block has an ambiguity of $1/(2 \cdot T_p)$ Hz, where T_p is the coherent integration time. This means that if the signal is acquired in a bin whose frequency is farther apart than $1/(2 \cdot T_p)$ Hz from the real signal frequency, the FLL will lock into a stable but wrong point. Therefore, the velocity estimated by the FLL will be inconsistent with the speed estimated by the DLL, and the receiver will determine that there was a problem during the acquisition process [SPILKER and ASHBYd, 1996]. To summarize, since for these simulations the signal is chosen to be always between two frequency bins which are separated by $1/(2 \cdot T_p)$ Hz, the simulator considers that the signal is well acquired in only two frequency bins, two lines of the frequency domain.

The traditional time requirement imposes that only the bins allowing the evaluation of the correlation between the generated local code replica and the received signal PRN code at the main lobe of the correlation should acquire the signal. Note that this constraint is necessary in order to ensure that the DLL is locked in the main CBOC autocorrelation peak and thus that the code delay is correctly tracked. Techniques were developed in order to allow the receiver to determine if the DLL is locked on the CBOC main peak or on the CBOC secondary peaks [WARD and LILLO, 2009] [JULIEN et al, 2007] [MARTIN et al, 2003] [FANTE, 2003]. Therefore, the allowed time uncertainty is enlarged and now, the signal is considered to be acquired in any bin having a time difference equal or smaller than one chip period, T_c seconds. Nevertheless, it is still more interesting to acquire the signal in the CBOC main peak. In order to inspect this last characteristic, two parameters are defined: the acquisition of the signal and the correct detection of the signal. A correct detection of the signal is found when a bin which fulfills the acquisition requirements has an acquisition criterion higher than the acquisition threshold. The bin of the signal causing a correct detection and having the largest acquisition criterion is the bin of the signal where the signal is acquired.

Once the scenario and the time and frequency requirements were presented, the results are presented. First, the most important result and the most relevant acquisition performance is the average acquisition time as a function of the signal C/N_0 at the receiver antenna output. The average acquisition times presented below and the remaining acquisition performances were calculated averaging 100 simulations of each acquisition option.

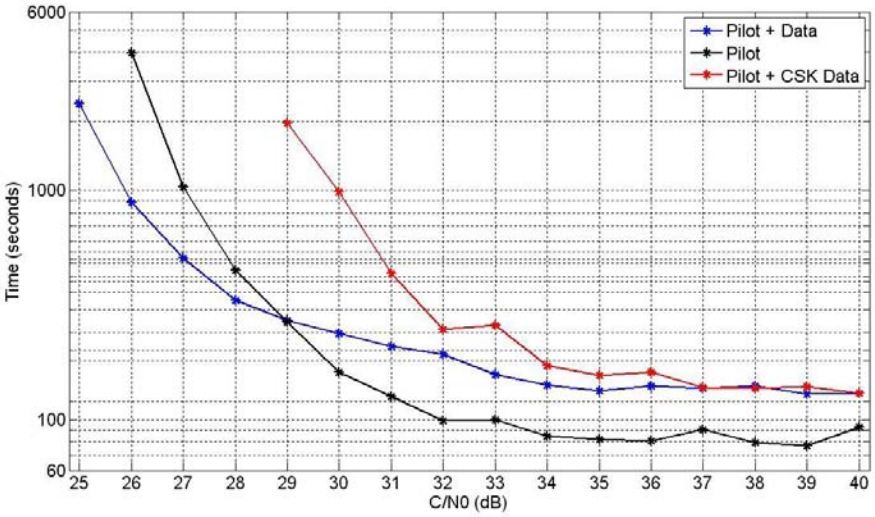

Figure 7-34: Average acquisition time for GALILEO E1 OS signal as a function of the data + pilot C/N_0

From Figure 7-34, it can be seen that the average acquisition time behavior can be divided into two zones. The first zone encloses the signal C/N_0 values at the receiver antenna output which are smaller than 29 dB-Hz. For these values, the average acquisition time is lower when the receiver uses both channels to acquire the signal than when the receiver uses only the pilot channel. Nevertheless, for an average acquisition time of about 1000 seconds, which is already a considerable amount of time for a pedestrian or a car vehicle user, the receiver using the pilot channel only requires 1 extra dB of power than the receiver using both channels.

The second zone encloses C/N_0 values larger than 29 dB-Hz and its main characteristic is that the receiver using only the pilot channel outperforms the receiver using both channels. This phenomenon can be easily explained. For high C/N_0 values, the probability of having the acquisition criterion higher than the acquisition threshold for the correct bins is quite

important. Therefore, for these values it becomes more important to inspect more bins each non-coherent integration time seconds than to increase the useful C/N_0 value employed to acquire the signal. And the option where only the pilot channel is used to acquire the signals inspects twice the number of bins tested with the option where both channels are employed each non-coherent integration time seconds. It can be observed that the option where only the pilot channel is used to acquire the signal reaches an acquisition time bound of about 80 seconds whereas the option where the pilot and data channels are used to acquire the signal reaches a bigger lower acquisition time bound, more specifically about 130/140 seconds.

Moreover, it can be seen that the average acquisition time obtained when the receiver uses both channels and the CSK technique is implemented is always larger than the time obtained when only the pilot channel is used. Therefore, it can be concluded that when the user cannot guarantee that the received data does not carry any CSK symbol, the acquisition should be made using only the pilot channel. Additionally, it can also be observed that the average acquisition time obtained for the CSK option when both channels are used to acquire the signal converges to the average acquisition time obtained for the option when both channels are used to acquire the signal but the CSK technique is not implemented.

The next two figures show different acquisition performances. Figure 7-35a displays the probability of false alarm, and Figure 7-35b shows the percentage of times that the signal was detected in the correct bins. Nevertheless, the reader must remember that different bins with different attenuation conditions fulfill the previous commented time and frequency requirements.

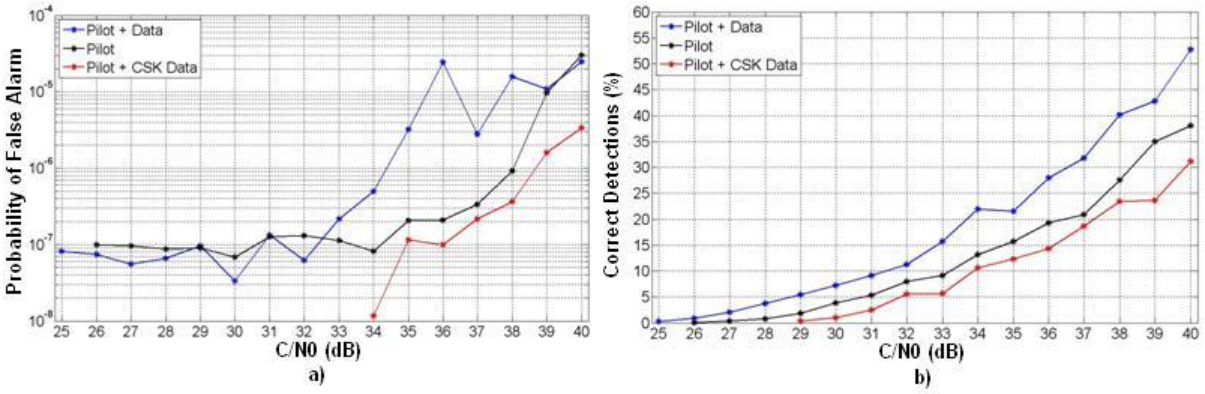

Figure 7-35: GALILEO E1 signal acquisition performance. a) Probability of false alarm. b) Locks percentage of times

From Figure 7-35a, it can be observed that the P_{fa} grows as the C/N_0 increases. From figure Figure 7-35b, it can be observed, as expected, that the probability of acquiring the signal in inspected bins where the signal should be acquired increases as the C/N_0 increases. These behaviors are explained below.

For low C/N_0 values, the distance between the acquisition threshold and the acquisition criterion (T_0) is not very large. This means that a small negative contribution of the noise in the correct inspected bin can make the acquisition criterion lower than the acquisition threshold. Moreover, this small difference also makes that the bins which have the correct PRN code delay but have an erroneous frequency value have very low probabilities of causing a false alarm. Note that these bins are the most dangerous ones since the secondary lobes of the cardinal sine can provide a significant useful signal to the total acquisition criterion. However, for high C/N_0 values, the distance between the acquisition threshold and the acquisition criterion (T_0) is large. This means that only significant negative contribution of the

noise in the correct inspected bin can make the acquisition criterion lower than the acquisition threshold. Therefore, it is justified that the percentage of correct detections is higher for high C/N_0 values. Moreover, this large difference also makes that the bins which have the correct PRN code delay but have an erroneous frequency value do not have very low probabilities of causing a false alarm, since they need a positive contribution of the noise not as large as the contribution required for low C/N_0 values. Besides, since each $M \cdot T_p$ seconds several bins are inspected and since in this simulation we sweep the entire bin domain, the bin fulfilling the frequency and time requirements is usually analyzed at the same time that the bins fulfilling the time requirements but which evaluate the frequency cardinal sine at its secondary lobes. Therefore, since these bins have a higher probability of causing a false alarm for higher C/N_0 values and since these bins are also inspected during the inspection of the bin fulfilling the frequency and time requirements, the increase of Pfa is justified.

Note that this last statement is only true when we calculate the number of false alarms before choosing among all the bins inspected during a period of $M \cdot T_p$ seconds the bin with the largest acquisition criterion (T_0). If the number of false alarms is calculated after choosing the bin with the largest T_0, Figure 7-35a will be different. The calculation of Figure 7-35b and Figure 7-36a are also made by counting first the number of detections before choosing the bin with the largest acquisition criterion.

Another observation that can be made from Figure 7-35a is that the probability of false alarm when using both channels and the CSK is not implemented is higher than the probability when both channels are used and the CSK is implemented. Nevertheless, from the explanation given in annex C.7.2.2, the probability of false alarm should be greater for the latter case. However, this fact can be explained using the same reason of the previous paragraph. The main source of false alarms is caused by the bins which have the correct PRN code delay but an erroneous frequency value. Therefore, since the case where the CSK technique is not implemented has a lower acquisition criterion but the acquisition threshold is the same for both cases, the probability of a false alarm caused by bins evaluated at the secondary lobes of the frequency cardinal sine is much lower.

Moreover, note that more than one bin fulfilling the acquisition requirements can be inspected during the same non-coherent integration time period. Therefore, it is interesting to analyze the percentage of acquisitions which are done over the CBOC autocorrelation main peak. Indeed, although there are techniques which allow tracking the signal even if starting the process over the secondary lobes, it is always more interesting to directly start at the main peak as has been said during the description of the time requirements.

Figure 7-36a shows the number of correct detections on the CBOC modulation main peak divided by the total number of correct detections. Figure 7-36b shows the percentage of times that the acquisition is done over the CBOC modulation main peak. The definitions of these parameters are given during the description of the time requirements.

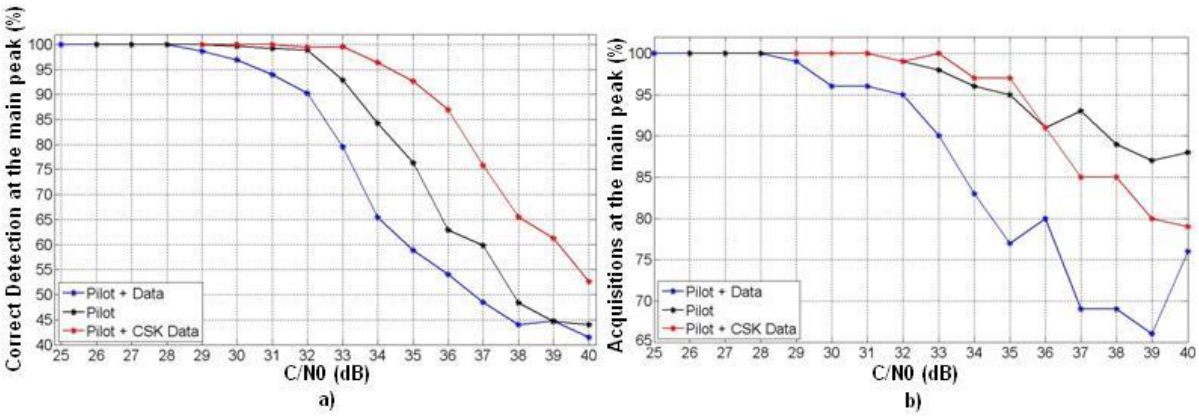

Figure 7-36: GALILEO E1 signal acquisition performance. a) Percentage of correct detections at the main peak. b) Percentage of acquisitions at the man peak

From Figure 7-36a, it can be seen that for low C/N_0 values all the correct detections are done in the CBOC modulation main peak, but this trend changes for high C/N_0 values until the point that most of the correct detections are done outside the main peak because of the emergence of the autocorrelation secondary peaks. Moreover, it can be observed that the option where the pilot and data channels are used together to acquire the signal when the CSK technique is not implemented is the option which has the lowest percentage of correct detections made at the main CBOC autocorrelation peak. This phenomenon is justified since this option provides the highest useful C/N_0 to acquire the signal with respect to the other two options, thus highlighting the secondary peaks. The same justification is true for the option which only uses the pilot channel to acquire the signal with respect to the option implementing the CSK technique.

From Figure 7-35b and Figure 7-36b, it can be seen that although the number of correct detections which are done outside the main CBOC autocorrelation peak increases as the C/N_0 increases, the bin chosen to acquire the signal, which is the bin with the largest acquisition criterion, is normally a bin where the CBOC correlation between the local code replica and the received signal PRN code is evaluated at its main peak. In fact, a bin where the CBOC correlation is not evaluated in its main peak is usually chosen because no inspected bin during the non-coherent integration time period contains the detected signal in the main CBOC autocorrelation peak.

Finally, it can be observed from Figure 7-36b that the option where only the pilot channel is used to acquire the signal obtains a higher percentage of acquisitions into the main CBOC autocorrelation peak than the option where the pilot and data channels are employed to acquire the signal and the CSK technique is not implemented. Therefore, it can be concluded that the former option, in addition to outperforming the latter option in average acquisition time for C/N_0 values higher or equal to 29dB-Hz, simplifies the tracking process by acquiring more times the signal into the main CBOC autocorrelation peak than the latter option.

7.10.2. Tracking performance preliminary conclusions

The loss of tracking performance introduced by the implementation of the CSK technique either over the GALILEO E1 OS signal or over any other navigation signal carrying two channels, a data channel and a pilot channel, is inspected in this chapter. The main difference from the analysis conducted for the acquisition performance is the fact that using the data

channel in order to track the signal is not necessary. A detailed explanation about why it is not needed to use the data channel to track the navigation signal is given below.

The most recent and future navigation signals such as GPS L2C, GPS L5, GPS L1C, GALILEO E1, etc, are formed by two channels transmitted either in phase or in quadrature-phase. These channels are the data channel, the channel responsible for broadcasting the data information, and the pilot channel, the channel responsible for improving the tracking performance. In fact, for the first GPS signal, GPS L1 C/A, the only existing channel is the data channel. And the tracking performance obtained by this signal, even if satisfactory, is limited by the data symbol duration because this value sets the maximal coherent integration time value which can be used. Therefore, in order to break this dependency, during the design of the new navigation signals, a pilot channel was implemented. The pilot channel is a dataless channel over which the tracking process can be applied without being limited by the symbol duration. Moreover, the absence of polarity in the pilot channel allows the use of more powerful PLL discriminators such as the Q and Arctan2 discriminators. To summarize, it can be concluded that the main function of the pilot channel consists in allowing the GNSS receivers to use a coherent integration time as long as allowed by clock stability and dynamics, and consists in allowing the GNSS receivers to use the Q or Arctan2 PLL discriminators.

Therefore, if the previous conclusions are applied to the case where a navigation signal has implemented the CSK technique over the data channel, it can be seen that the tracking performance should not be modified a priori. More specifically, if the tracking process wants to profit from unlimited coherent integration time and from more powerful PLL discriminators, the tracking process should be limited to the pilot channel. But, if the tracking process wants to profit from the entire power carried by the navigation signal, the tracking process should be applied over both channels but using time limited integrations and less powerful PLL discriminators. And, in this last case, the CSK technique will prevent the tracking process from obtaining its best performance since the data channel will usually provide only noise. However, the best performance is poorer than when using the pilot channel only.

Moreover, the data wipe-off technique which allows using the Q and Arctan2 discriminators and which breaks the data channel time limitations can still be perfectly executed when the navigation signal implements the CSK technique. In this case, the only difference with a signal not implementing the CSK is that instead of smartly reversing the data channel polarity before adding the channel to the pilot channel, the receiver must choose the correct correlator output to be added to the pilot channel. And the correct correlator output is the correlator which matches the code circular shifted version being currently received and thus which provides the useful signal.

To summarize, the introduction of the CSK signaling technique over a navigation signal carrying two channels, the data channel and the pilot channel, does not modify the signal tracking performance. The only introduced modification is that the receivers are obliged to track the signal only over the pilot channel, whereas without CSK the receivers can choose to track the signal over both channels but using less powerful PLL discriminators and using integrations limited by the data symbol duration.

Finally, this coherent integration time limitation is more restrictive for GALILEO E1 OS signal than for GPS L1C signal since the data symbol duration is much shorter for the GALILEO E1 OS signal.

7.10.3. Impact of the CSK technique on the pseudo-range calculation

In order to understand the impact of the CSK technique on the pseudo-range calculation, we need first to know how this calculation is made on the traditional GNSS signals such as GPS L1 C/A.

In the GPS L1 C/A signal, 3 mains operations are effectuated in order to calculate the pseudo-range measurement since time ambiguities must be solved. The first operation consists in tracking the satellite PRN code until the beginning of a received PRN period is identified. In this operation, we are able to determine the propagation time with an ambiguity equal to an integer number of PRN code periods since we do not know in which bit and in which period inside a bit the PRN code replica is locked. The second operation consists in making the synchro-bit process in order to determine in which period inside a GPS L1 C/A bit the PRN code has been tracked. In this case, we determine the time between the start of the bit and the locked PRN code. However, we still do not know how many bits have been transmitted between the transmission and the reception instants of the tracked signal, which implies an ambiguity of the calculated signal propagation time equal to an integer number of bits. The third and last operation consists in effectuating the synchro-frame process. More specifically, the TOW field is identified inside the received frame (along the TLM and inside the HOW field) and thus, the number of bits between the start of the transmitted frame and the instant when the first bit of frame is received is determined resolving then the last ambiguity. Therefore, the pseudo-range measurement is calculated from the addition of the 3 previous calculated times.

In the case of a future GNSS signal, more specifically the GALILEO E1 OS signal, the PRN code period is equal to the duration of a symbol. Therefore, the second previous operation is no longer required. Nevertheless, the other two operations must still be effectuated and thus, the impact of the implementation of the CSK technique is evaluated on them. First, GALILEO E1 OS signal has a pilot channel which allows the tracking of the PRN code and thus the first operation can be conducted as before. Second, the TOW of the navigation message is found on the data channel and, as in the GPS L1 C/A case, its value is known allowing its identification after the message demodulation. This means that in both cases the navigation message has to be demodulated since once the TOW is known, the determination of the ambiguity of the number of transmitted symbols is easily resolved. In fact, the only difference is the demodulation method which is different from a CSK modulation to a BPSK modulation and the fact that we need a higher C/N_0 value to demodulate a CSK signal.

To sum up, the implementation of the CSK technique on the GALILEO E1 OS signal has as impact on the calculation of the pseudo-range measurement a higher C/N_0. Nevertheless, if we adopt the new navigation message proposed by TAS-France, we can see that only a part of the message is CSK modulated. In fact, the TOW is encoded individually at the beginning of the frame and is BPSK modulated. This means that in order to resolve the symbol ambiguities we should effectuate a BPSK demodulation and thus the CSK implementation does not have any impact on the calculation of the pseudo-range measurement.

7.11. Conclusions

In this section, the conclusions extracted from the detailed analysis of the CSK signaling method and from the GALILEO E1 OS signal acquisition and tracking performance in the presence of the CSK signaling method are presented.

First, the conclusions extracted from the general analysis of the CSK signaling method are summarized.

The Code Shift Keying technique is a type of orthogonal M-ary signaling technique, where the orthogonal M-ary signaling technique is a well known method which is already thoroughly analyzed in the literature. Moreover, several specific studies about the CSK were conducted along the years, where the CSK hard demodulators as well as their demodulation performance (BER) are perfectly characterized. Nevertheless, no scheme for a soft demodulator was proposed. And this means that the only channel codes which can be applied over the bits transmitted by the CSK method are hard input codes. Therefore, in this chapter, the expressions of the likelihood ratios for the bits transmitted inside a CSK symbol are developed in order to allow the design of a CSK soft demodulator scheme, which, in its turn, allows the application of soft input codes over the bits transmitted by the CSK signaling method.

In this chapter, the CSK bits likelihood expressions are developed for orthogonal and bi-orthogonal CSK signaling methods, and they are verified by comparing the BER obtained from them to the BER obtained from the hard output demodulators. In fact, the obtained BER is the same for hard and soft demodulators, and thus, it is verified that the proposed soft output methods are equivalent to the hard output methods. Moreover, two expressions are developed for the bi-orthogonal CSK method, where the a priori optimal expression provides worse demodulation performance than the suboptimal expression.

Additionally, a closer inspection of the proposed orthogonal and bi-orthogonal CSK likelihood ratios expressions states that, in addition to depending on the message E_b/N_0, the demodulation performance provided by a channel code which uses as inputs the CSK bits likelihood ratios depends on the number of bits mapped by a single CSK symbol. More specifically, it is verified that if the E_b/N_0 value is constant for any number of bits mapped by a single CSK symbol, the increase of the number of bits mapped by a CSK symbol leads to a larger variance of the lognormal likelihood ratio. Therefore, the demodulation performance when a soft input channel code is implemented is better for CSK symbols mapping more bits. However, if the CSK E_s/N_0 is constant for any number of bits mapped by a CSK symbol, the decrease of the number of bits mapped by a CSK symbol leads to a larger variance of the lognormal likelihood ratio. Therefore, the demodulation performance when a soft input channel code is implemented is better for CSK symbols mapping fewer bits. Moreover, note that if the E_s/N_0 is constant, the E_b/N_0 value has to vary for different number of bits mapped by a CSK symbol; and the inverse is also true.

Once the likelihood ratio expressions of the CSK bits are developed, verified and analyzed, the best source mapping between the bits transmitted inside a CSK symbol and the bits belonging to a transmitted word is inspected in this chapter. More specifically, a study is conducted in order to search for the CSK source packet mapping which provides the best demodulation performance. This analysis is made in two steps, where the first step determines the CSK source packet mapping characteristic to be optimized, and the second step conducts the analysis to optimize this characteristic.

In the first step, it is stated that in order to obtain the maximum number of correct words after the channel code correction, the mapping has to improve the probability of the channel code to correct a wrong received word. Moreover, it is also stated that the lower the variance of the number of errors of a word is, the higher this probability of the channel code to correct a wrong word becomes. Therefore, the first step concludes that the optimal CSK source packet mapping is the mapping which provides the lowest variance of the number of errors of a

word. However, this statement is only valid when the average number of errors found in a word is lower than the channel code correction capacity.

In the second step, the conducted analyses shows that the lowest variances of the number of errors are obtained by the CSK source packet mappings which transmit the fewest bits belonging to the same packet inside a single CSK symbol. Therefore, it is concluded that the optimal mapping is the mapping where each one of the bits transmitted inside a CSK symbol belongs to a different packet. Moreover, it is stated that the evolution of the variance obtained by each possible mapping configuration does not depend on the number of bits mapped by a CSK symbol but rather on the number of bits belonging to the same packet transmitted inside the same CSK symbol. Additionally, it has also been observed the existence of a linear relationship between the word size and the variance value; when the word doubles its size, the variance value is multiplied by root of 2.

Second, the conclusions extracted from the analysis of the CSK implementation over the GALILEO E1 signal are presented. Additionally, remember that the CSK signaling technique is planned to be implemented in the LEX signal of the Japanese GNSS called QZSS as indicated in the public released interface specification [JAXA, 2010].

The most important conclusion is that due to the already defined parameters of the GALILEO E1 OS signal, the CSK implementation is very restricted and thus its properties are quite particular. More specifically, the maximum number of bits which can be mapped by a CSK symbol is determined by the data channel PRN code. And this means that a maximum of 11 bits can be transmitted with an orthogonal CSK and a maximum of 12 bits can be transmitted with a bi-orthogonal CSK for the GALILEO E1 OS signal. Moreover, since the data PRN channel code is defined and its time duration cannot be modified, the E_s/N_0 of a CSK symbol is always the same regardless of the number of bits mapped by the symbol. Therefore, it can be concluded that, with or without the implementation of a channel code, the best demodulation performance obtained by the CSK method implemented over the GALILEO E1 OS signal is obtained when fewer bits are mapped by a CSK symbol.

Other important conclusions drawn from this chapter are about the demodulation performance obtained by the different CSK configurations which could be implemented over a future GALILEO E1 OS signal. The first conclusion is that, as expected from the general study on the CSK packet source mapping, a CSK configuration where the bits transmitted inside a single CSK symbol belong to different packets outperforms a CSK configuration where all the bits transmitted inside a single CSK symbol belong to the same packet. More specifically, the difference in C/N_0 at the receiver antenna output when the desired BER is equal to 10^{-5} is about 0.5 dB for orthogonal CSK configurations mapping 6 bits, about 0.55 dB for orthogonal CSK configurations mapping 8 bits, about 0.6 dB for orthogonal CSK configurations mapping 11 bits and about 0.65 dB for bi-orthogonal CSK configurations mapping 12 bits. Moreover, this difference is also found for the WER figure of merit; for a WER equal to 10^{-4}, there is a difference of about 0.5 dB for orthogonal CSK configurations mapping 6 bits, of about 0.55 dB for orthogonal CSK configurations mapping 8 bits, of about 0.6 dB for orthogonal CSK configurations mapping 11 bits and about 0.65 dB for bi-orthogonal CSK configurations mapping 12 bits. Moreover, it is also concluded that this difference of performance between, CSK configurations where the bits transmitted inside a single CSK symbol belong to different packets, and CSK configurations where all the bits transmitted inside a single CSK symbol belong to the same packet, grows as the number of bits mapped by a single CSK symbol increases. Therefore, it is verified that the variance of the number of errors of a word grows as the number of bits mapped by a single CSK symbol increases as concluded in a previous paragraph.

Additionally, although it is verified that the CSK configurations where all the bits transmitted inside a single CSK symbol belong to the same packet obtain a worse demodulation performance than the CSK configuration where all the bits belong to different packets, some designers can choose to use the former option since the word reception for the former configuration is not subject to any additional delay. In this case, it has been stated that these former CSK configurations need to apply an interleaver over the transmitted word in order not to send consecutive bits inside the same CSK symbol. If this consideration is not respected, the demodulation performance of the CSK configuration is penalized.

Besides, it is also concluded that although the demodulation performance when no channel code is implemented is the same for the orthogonal and bi-orthogonal CSK configurations; the orthogonal configurations slightly outperform the bi-orthogonal configurations when a soft input channel code is implemented. Nevertheless, the C/N_0 difference is very small since it never exceeds 0.2 dB. Moreover, this difference decreases as the number of bits mapped by a single CSK symbol increases.

Moreover, the previous statement announcing that a better demodulation performance improvement is achieved for CSK configurations mapping fewer bits is verified. More specifically, this verification is made by observing the gain difference obtained between CSK configurations implementing a channel code and CSK configuration not implementing a channel code. Some numerical examples are given next.

If a BER value equal to 10^{-5} is searched for a CSK configuration implemented on the GALILEO E1 OS signal, an improvement of 6.8 dB is found when comparing the C/N_0 required at the receiver antenna output for an orthogonal CSK configuration mapping 6 bits implementing the GPS L1C LDPC channel code with the C/N_0 required for the same orthogonal CSK configuration but without implementing a channel code. However this difference is decreased to 6.05 dB for an orthogonal CSK configuration mapping 8 bits and it is farther decreased to 5.4 dB for an orthogonal CSK configuration mapping 11 bits. Moreover, the C/N_0 values required for the previous configurations to obtain a BER equal to 10^{-5} are respectively 34.1, 35.25 and 36.5 dB-Hz. Note that these values are valid for the CSK configurations where the bits transmitted inside a single CSK symbol belong to different packets.

Nevertheless, the previous commented C/N_0 values are very high. In fact, even for the CSK configuration mapping 6 bits, the difference of demodulation performance of the ephemeris data of the new proposed GALILEO E1 OS navigation message and a message transmitted by the CSK signaling method is about 8 dB. Therefore, it can be concluded that only at the best reception situations, the desired level of BER or WER could be obtained for a message transmitted by the CSK signaling technique implemented over the GALILEO E1 OS signal.

The advantages of the CSK applied to a GNSS signal have also been quickly summarized in this chapter. The increase of available data information rate allows the transmission of either new information of commercial services, which provide a financial gain, or redundant information such as the satellite ephemeris, which improve the signal acquisition, tracking and demodulation thresholds. In this last case, the broadcasting of future ephemeris of the satellite can avoid the necessity of the demodulation of these ephemerides in a later hostile scenario, and the broadcasting of ephemeris of other satellites can decrease the acquisition and tracking thresholds by means of wipe-off techniques.

Moreover, the increase of the bit rate is not easily achieved using a BPSK modulated signal with a higher symbol rate since it implies two possible undesired scenarios. First, increasing the symbol rate and maintaining constant the chip rate leads to a shorter PRN code with its

consequent loss of performance: isolation and near/far effect. Second, increasing the symbol rate and the chip rate in order to maintain the same PRN code length leads to a wideband receiver with the consequent increase of the number of operations and a larger spectral occupation.

Finally, the conclusions extracted from the calculation of the pseudo-range measurement, the new tracking performance and the new acquisition performance of the GALILEO E1 OS signal when the CSK signaling method is implemented over the data channel are presented. In this case, the acquisition performance has been analyzed for a cold start without any a priori information and the simulated scenario has been modeled in order to represent the worst possible attenuation conditions.

Three different acquisition modes are inspected in this chapter in order to analyze if the CSK signaling method implementation over the data channel influences negatively the acquisition performance. The first inspected mode consists in acquiring the GALILEO E1 OS signal by using both the pilot and data channels when the CSK method is not implemented. The second inspected mode consists in acquiring the signal by using only the pilot channel. The third and last inspected mode consists in acquiring the signal by using both the pilot and data channels when the CSK method is implemented.

The first conclusion extracted from the acquisition performance analysis presents the optimal mapping between the CSK symbols and the circular shift code versions which minimizes the increase of the false alarm events caused by the CSK method implementation. This mapping consists in associating each CSK symbol to the nearest available circular shift code version which is the closest to N/2, where N is the number of code chips.

The second conclusion states that whenever the CSK technique is implemented, the best acquisition performance is always obtained when only the pilot channel is used to acquire the signal. Indeed, the acquisition mode using only the pilot channel always has a better acquisition performance than the acquisition mode using the pilot and data channel when the CSK is implemented. Therefore, it only remains to inspect the difference of acquisition performance between the acquisition mode using only the pilot channel and the acquisition mode using both channels when the CSK is not implemented. This difference is the loss of acquisition performance due to the fact that the receiver cannot use both channels to acquire the GALILEO E1 OS signal since the CSK method is implemented on the data channel.

This loss of acquisition performance between the acquisition mode using both channels when the CSK is not implemented and the acquisition mode using only the pilot channel only appears for C/N_0 values at the receiver antenna output which are lower than 29 dB-Hz. However, this loss is small. In fact, for an average acquisition time of 1000 seconds -a very large time to wait for a pedestrian or car user- the acquisition mode using both channels when the CSK is not implemented needs a C/N_0 equal to 26 dB-Hz whereas the acquisition mode using only the pilot channel needs a C/N_0 equal to 27 dB-Hz. This means that there is only a difference of about 1 dB between both modes at a very big average acquisition time.

Moreover, for C/N_0 values larger than 29 dB-Hz, the acquisition mode using only the pilot channel obtains better acquisition performance than the acquisition mode using both channels when the CSK is not implemented. This means than the average acquisition time of the mode using only the pilot channel becomes smaller than the average acquisition time of the mode using the pilot and data channels. And the difference in average acquisition time between the modes keeps growing until the acquisition mode using only the pilot channel reaches a lower bound of about 80 seconds and the acquisition mode using the pilot and data channels when the CSK is not implemented reaches a lower bound of about 130/140 seconds.

Besides, the percentage of times that the signal is acquired on the main peak of the CBOC modulation is higher for the acquisition mode using only the pilot channel than for the acquisition mode using both channels when the CSK is not implemented. Therefore, the third conclusion states that the loss of acquisition performance caused by the implementation of the CSK signaling method over the GALILEO E1 OS data channel is not significant taking into account the higher information rate provided by the CSK signaling method.

The main conclusion extracted from the preliminary analysis of the tracking process states that the tracking performance of the new generation of navigation signals is not affected by the implementation of the CSK signaling method on the data channel. The reason is that the inclusion of a dataless channel into the new navigation signals answered the need of removing the time limitation imposed by the data symbol period and the limitation of the operating region of the discriminator imposed by the data sign. Therefore, the use of both channels at the same time in order to track the signal if the data is not known in advance is not expected. Moreover, the CSK signaling method also allows the application of the data wipe-off technique if the data is known in advance.

The method applied to calculate the pseudo-range measurement of the GALILEO E1 OS signal when this signal implements the CSK technique is the same method as when this signal does not implement the CSK technique. The only difference is found in the case where the TOW field is CSK modulated since the demodulation of this field is mandatory. In this case, the demodulation process of this field is different for both options and the required C/N_0 to calculate the pseudo-range measurement increases since the CSK demodulation performance is worse than the BPSK demodulation performance on the GALILEO E1 OS signal case.

Chapter 8. Conclusions and future work

First, the conclusions obtained from the results in the previous chapters are presented, second the contributions made along this dissertation are given and last the perspectives for future work are drawn in this chapter.

8.1. Conclusions

An AWGN channel mathematical model which represents the transmission of a navigation signal through an open environment is presented for purposes of analysis of demodulation performance. This proposed model simulates the loss of power of the useful signal component employed to demodulate the signal due to the carrier phase tracking process. The PLL carrier phase estimation is only affected by thermal noise and dynamic stress errors. The block scheme of a simulator representing the channel model is also proposed.

A mobile channel mathematical model which represents the transmission of a navigation signal through a urban environment is presented for purposes of analysis of demodulation performance. The proposed model is defined from the general theory of mobile channels and from the three state mathematical model proposed by Perez-Fontan [PEREZ-FONTAN et al, 1998] [PEREZ-FONTAN et al, 2001]. More specifically, the proposed model simulates the received signal at the receiver antenna output as the transmitted signal at the transmitter antenna input multiplied by a complex coefficient. The envelope of this complex coefficient follows a Loo distribution and the parameters of the Loo distribution depend on the current state of the signal among the 3 possible states defined by the mathematical model of Perez-Fontan. The Loo distribution parameters values and the three states parameters values were determined by the DLR [PEREZ-FONTAN et al, 2001]. The proposed model also takes into account the loss of power of the useful signal component employed to demodulate the signal due the carrier phase tracking process. The PLL carrier phase estimation is only affected by thermal noise and dynamic stress errors. For this proposed mathematical model to be valid, we assume that the receiver does not travel faster than a maximum speed depending on the signal symbol duration. The block scheme of a simulator representing the channel model is also proposed.

The binary prediction of the GPS L1 C/A signal ephemeris is been proven to be too hard to achieve. More specifically, neither the a priori correlation between satellite orbits, nor signal natural domain processing methods such as non-parametrical spectral estimation methods, PRONY model and neural networks, are effective in predicting the desired ephemeris data. In fact, the only method which provides some measure of binary prediction consists in predicting the ephemeris data by employing external information, such as the satellites almanacs data or long term ephemeris data predicted by external programs. Nevertheless, these methods do not reach the desired level of performance of the binary prediction.

The reasons of why the GPS L1 C/A signal ephemeris data binary prediction cannot be achieved are also identified in this dissertation. First, the accepted prediction position error is very small, about 1m, which results into a very low probability of error tolerated for each predicted bit. Second, the special relationship between the binary and the natural domain implies the necessity of an extremely accurate natural domain prediction in order to achieve poor binary domain prediction. Third and last, the ephemeris data base used to build the mentioned signal processing methods is far from ideal, which highly complicates any natural domain prediction.

Two alternative decoding methods which improve the demodulation performance of the GPS L2C mode NAV with FEC data signal, the GPS L2C mode CNAV data signal and the GPS L5 signal obtained by applying the traditional Viterbi decoding algorithm are proposed in this dissertation. This means that in addition to calculating the previous signals demodulation performance with the new proposed methods, the demodulation performance obtained by traditional Viterbi decoding is also calculated in this dissertation.

The first method consists in a modification of the traditional Viterbi algorithm so that the a priori bit probabilities can be used to improve the decoding process. However, this new proposed algorithm only introduces a slight improvement of about 0.2 dB when applied with the a priori probabilities provided by this dissertation [annex F.3]. Moreover, this gain is reduced as the C/N_0 at the receiver antenna output increases until that the traditional Viterbi algorithm and the modified Viterbi algorithm obtain the same demodulation performance. Therefore, this method is considered as a complement to the second proposed decoding method rather than a full alternative decoding method.

The second method consists in a combination of the inner and outer channel codes of the signals. More specifically, the method consists in searching with the Viterbi algorithm the most probable emitted sequence which passes the outer channel code verification. For this method, the selection of the best parameters is made and the difference of gain between the traditional Viterbi decoding and the proposed method is calculated. The main parameter and the gains are presented below.

The maximum number of sequences inspected by the outer channel code which yields to the best compromise with the receiver resources needed to apply the method is determined to be 100 sequences. And the gain obtained by this proposed method employing 100 sequences when the PLL carrier phase estimation process is noised by thermal noise is about 1.3 dB for GPS L2C mode NAV data signal, about 1.6 dB for GPS L2C mode CNAV data signal and about 1.7 for GPS L5 signal when the desired BER is equal to 10^{-6}. This gain is about 1.7 dB for GPS L2C mode CNAV data and 1.85 dB for GPS L5 signal when the desired WER is equal to 10^{-4}. And this gain is about 1.4 dB for GPS L2C mode NAV data signal when the desired EER is equal to 10^{-3}. Moreover, if the PLL carrier phase estimation process is noised by thermal noise and by a dynamic stress error of 1 jerk, the obtained gain is about 1.5 dB for GPS L2C mode NAV data signal, about 1.7 dB for GPS L2C mode CNAV data signal and about 1.75 for GPS L5 signal when the desired BER is equal to 10^{-6}. This gain is about 1.9 dB for GPS L2C mode CNAV data signal and GPS L5 signal when the desired WER is equal to 10^{-4}. And this gain is about 1.65 dB for GPS L2C mode NAV data signal when the desired EER is equal to 10^{-3}.

GPS L1C signal ephemeris data demodulation is proven to outperform GALILEO E1 OS signal ephemeris data demodulation by means of a preliminary theoretical study. The elements of the signal design which are identified as responsible of this demodulation performance difference are the signal channel relative power distribution, the symbol transmission rate, the signal data message structure (channel code and ephemeris data distribution), the interleaver and the signal data message content (repetition of the ephemeris data in consecutive frames). The combination of the GPS L1C signal channel relative power distribution with the GPS L1C symbol transmission rate provide 1 extra dB to the navigation message E_b/N_0 than the GALILEO E1 OS combination does. The GPS L1C channel code is much more powerful than the GALILEO E1 OS signal channel code and the division of the GALILEO E1 OS ephemeris data into 4 words worsens the signal EER with respect to the signal WER. The GPS L1C interleaver can stand much longer bursts of errors than the GALILEO E1 OS interleaver. And the GPS L1C ephemeris data can be coherently

accumulated in order to increase the navigation message E_b/N_0 whereas the GALILEO E1 OS ephemeris data can only be partially coherently accumulated and its accumulation process is not guaranteed.

The preliminary theoretical results are verified for a transmission through an open environment. This environment is simulated as an AWGN channel, and an ideal carrier phase tracking process is assumed. In this simulated scenario, the GALILEO E1 OS signal has proven to require 3.2 extra dB of C/N_0 at the receiver antenna output than GPS L1C signal in order to obtain a BER equal to 10^{-5}, 3.1 extra dB in order to obtain a WER equal to 10^{-3} and 3.6 extra dB in order to obtain a EER equal to 10^{-3}. Moreover, these differences grow as the C/N_0 increases. The more powerful channel code implemented for GPS L1C and its additional 1 dB obtained with respect to the GALILEO E1 OS signal are the main factors of difference of the ephemeris data demodulation performance between signals.

The preliminary theoretical results are also verified for a transmission through a urban environment. This environment is simulated as a mobile channel, and the carrier phase tracking process is assumed to be only disturbed by thermal noise. The calculated results are the best instantaneous ephemeris data demodulation performance which can be obtained in one of the worst possible reception scenarios, since the selected satellite elevation angle is equal to 30^o and the Perez-Fontan three state model is simplified to only one state, the line-of-sight state. In this simulated state, when the receiver travels at 30 km/h, GALILEO E1 OS is proven to require about 3.7 extra dB of C/N_0 at the receiver antenna output than GPS L1C in order to obtain a BER equal to 10^{-5}, 3.8 extra dB in order to obtain a WER equal to 10^{-3} and 4.2 extra dB in order to obtain an EER equal to 10^{-3}. Moreover, when the receiver travels at 5 km/h, GALILEO E1 OS is proven to require about 5.4 extra dB of C/N_0 at the receiver antenna output than GPS L1C in order to obtain a BER equal to 10^{-5}, 4.8 extra dB in order to obtain a WER equal to 10^{-3} and 5.7 extra dB in order to obtain an EER equal to 10^{-3}. The increase of the difference of the ephemeris data demodulation performance between the two signals compared to the AWGN channel case is the result of the more powerful interleaver implemented for GPS L1C signal. More specifically, the GALILEO E1 OS interleaver is proven to have an insufficient capacity to break the majority of bursts of errors generated when the receiver travels at 5 km/h.

The average ephemeris data demodulation performance obtained for GPS L1C when only one word is used to decode the ephemeris word is proven to be insufficient. More specifically, even the words only containing bits received when the PLL is locked can only obtain a BER lower than $4 \cdot 10^{-4}$ for any C/N_0 value at the receiver antenna output. Therefore, it is concluded that the word accumulation strategy is the only valid strategy to obtain the desired BER levels of performance. Moreover, it is observed that this strategy consists in accumulating the bits rather than entire words since only 4% of the total received words have all their bits received when the PLL is locked. Nevertheless, the PLL is locked only 65% of the time when the receiver travels into a urban environment.

Additionally, it is proven that no significant gain can be obtained by implementing a pilot channel estimation method instead of a PLL in order to estimate the received signal carrier phase. However, the implementation of a pilot channel estimation method can be interesting if it provides a good demodulation performance in zones where the PLL is not locked.

The example of a possible future GALILEO E1 OS navigation message structure proposed by TAS-France has been shown to increase the navigation message information transmission rate. This increase of information transmission rate is achieved by the implementation of the Code Shift Keying signaling method over the data channel.

This increase of data information rate presents a great advantage since it can be used either by broadcasting information of new commercial services, which implies a financial gain, by broadcasting redundant information such as future ephemeris of the broadcasting satellite, which improves demodulation performance, or by broadcasting other satellites ephemerides, which improves acquisition and tracking performance by means of wipe-off techniques.

Moreover, the increase of the bit rate is not easily achieved using a BPSK modulated signal with a higher symbol rate since it implies two possible undesired scenarios. First, increasing the symbol rate and maintaining constant the chip rate leads to a shorter PRN code with its consequent loss of performance: isolation and near/far effect. Second, increasing the symbol rate and the chip rate in order to maintain the same PRN code length leads to a wideband receiver with the consequent increase of the number of operations and a larger spectral occupation.

The models of the likelihood ratios of the bits for orthogonal and bi-orthogonal CSK symbols are developed and verified in this dissertation. These expressions provide the same demodulation performance than the well known hard output expressions when no channel code is implemented over the transmitted bits.

The best source mapping between the bits transmitted inside a single CSK symbol and the words to which these bits belong when a channel code is implemented over each word, is determined as the source mapping where all the bits transmitted inside a CSK symbol belong to a different word.

The special implementation of the CSK signaling method over the GALILEO E1 OS signal data channel is defined in this dissertation. Moreover, the different demodulation performance obtained for different numbers of bits mapped by a single CSK symbol when no channel code is implemented over a word is calculated. This demodulation performance shows that the increase of the number of bits are mapped by a single CSK symbol leads to higher required C/N_0 values at the receiver antenna output in order to obtain a desired BER.

The CSK demodulation performance when the transmitted word size is equal to the size of the GPS L1C word containing the ephemeris data, and when the channel code of the GPS L1C word containing the ephemeris data is applied over the transmitted word but keeping the GALILEO E1 OS symbol rate, is calculated. These results confirm that the optimal CSK source packet mapping is the mapping where all the bits transmitted inside a single CSK symbol belong to different packets. Moreover, these results also show that the orthogonal CSK methods slightly outperform the bi-orthogonal CSK methods by a maximum difference of 0.2 dB.

The signal C/N_0 at the receiver antenna output necessary to obtain a BER equal to 10^{-5}, when an orthogonal CSK signaling method is implemented over the GALILEO E1 OS signal data channel, when the CSK transmitted word and its implemented channel code are equal to the GPS L1C ephemeris word and channel code, and when all the bits transmitted inside a single CSK symbol belong to different words, is calculated. The values are 34.1 dB-Hz with a CSK symbol mapping 6 bits, 35.25 dB-Hz with a CSK symbol mapping 8 bits and 36.5 dB-Hz with a CSK symbol mapping 11 bits.

The GALILEO E1 OS acquisition process which only uses the pilot channel to acquire the signal is shown to outperform the acquisition process which uses the pilot and the data channels to acquire the signal when the CSK signaling method is implemented over the data channel for any C/N_0 value at the receiver antenna output. Nevertheless, the GALILEO E1 OS acquisition process which only employs the pilot channel to acquire the signal is shown to outperform the acquisition process which uses the pilot and the data channels to acquire the

signal when the CSK signaling method is not implemented over the data channel only for C/N_0 values at the receiver antenna output equal or higher than 29 dB-Hz. More specifically, the former acquisition process mode needs 1 extra dB, 27 dB-Hz, than the latter acquisition process mode, 26 dB-Hz, in order to obtain an average acquisition time equal to 1000 seconds. Note that this acquisition time is very long for either pedestrian or car users.

The GALILEO E1 OS signal average acquisition time obtained when employing the pilot and the data channel to acquire the signal is shown to reach a lower bound of about 130/140 seconds regardless of the CSK signaling method implementation over the data channel. And the GALILEO E1 OS signal average acquisition time obtained when employing only the pilot channel to acquire the signal is shown to reach a lower bound of about 80 seconds.

From theoretical analysis is deduced that the GALILEO E1 OS signal tracking performance is unaffected by the implementation of the CSK signaling method over the data channel. Moreover, the implementation of the CSK signaling method does not prevent the application of the data wipe-off techniques.

Finally, the impact of the CSK technique implementation on the GALILEO E1 OS signal over the calculation of the pseudo-range measurement is expected to increase the C/N_0 threshold when the TOW field is CSK modulated. However, no impact is anticipated when this field is BPSK modulated since not all the data channel has to implement the CSK technique.

8.2. Original Contributions

The main contributions of this thesis are enumerated below and are detailed all along this dissertation. The section number where each contribution can be found is also indicated in this section.

- Definition of an Additive White Gaussian Noise channel mathematical model (3.1) and definition of a mobile channel mathematical model (3.2) suited for analysis of the demodulation performance of a navigation signal. These models include the influence of the carrier phase tracking losses on the demodulation performance. The application of the mobile channel mathematical model depends on the relation between the PLL coherent integration time and the receiver speed.
- The binary prediction of the GPS L1 C/A signal ephemeris by using several signal processing methods, such as spectral estimations, PRONY parametric model and Neural Networks (4.3.1)
- A new decoding strategy proposed for the GPS L2C and GPS L5 signals (5.2.2)
- The analysis of the GALILEO E1 signal demodulation performance when the signal is transmitted through an AWGN channel (6.4.2)
- The analysis of the GPS L1C and the GALILEO E1 OS signals demodulation performance when the signals are transmitted through a mobile channel (6.4.3)
- The likelihood ratio expressions for the bits transmitted employing the Code Shift Keying (CSK) signaling method for a coherent demodulation (7.5)
- The optimal mapping between the bits transmitted inside a CSK symbol and the words where these symbols belong (7.6)
- The demodulation performance obtained by the CSK signaling method when applied over the GALILEO E1 signal data channel (7.8.2)

- The GALILEO E1 signal acquisition performance when the data channel is implemented with the CSK signaling method. This acquisition performance is compared with the GALILEO E1 OS signal acquisition performance when the CSK is not implemented with the CSK signaling method and the signal is acquired either with both channels or only with the pilot channel (7.10.1)

The articles published along this dissertation are listed below.

- A. GARCIA-PENA, M-L BOUCHERET, C. MACABIAU, A-C. ESCHER, L. RIES, J-L. DAMIDAUX, S. CORAZZA, *An Improved Method to Decode GPS L2C/L5 Navigation Message: Combination of the Inner and the Outer Channel Codes*, Proceedings of the 2009 International Technical Meeting of the Institute of Navigation, January 26 - 28, 2009, Anaheim, CA.
- A. GARCIA-PENA, C. MACABIAU, A-C. ESCHER, M-L BOUCHERET, L. RIES, *Comparison between the Future GPS L1C and GALILEO E1 OS Signals Data Message Performance*, Proceedings of the 2009 International Technical Meeting of the Institute of Navigation, January 26 - 28, 2009, Anaheim, CA.
- A. GARCIA-PENA, C. MACABIAU, A-C. ESCHER, M-L BOUCHERET, L. RIES, *Demodulation Performance of Galileo E1 OS and GPS L1C Messages in a Mobile Environment*, Proceedings of the 22nd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2009), September 22 - 25, 2009, Savannah, GA.
- A. GARCIA-PENA, M-L. BOUCHERET, C. MACABIAU, J-L. DAMIDAUX, L. RIES, S. CORAZZA, A-C. ESCHER, *Implementation of Code Shift Keying signaling technique in GALILEO E1 signal*, 5th ESA Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), December 8-10, 2010, Noordwijk (Netherlands).

8.3. Perspectives for future work

The proposed mobile channel mathematical model used to represent the transmission through a urban environment could be improved. The losses due to the code delay tracking performance could be modeled, and the model could be expanded to accept higher receiver speeds than the speeds allowed by the chosen PLL coherent integration time. In this last case, amplitude losses and interferences from other satellites should be modeled.

The ephemeris data binary prediction could be tried one last time by using neural networks which directly predict the ephemeris data bits instead of the natural domain ephemeris values. Moreover, in order to increase the chances of success, the ephemeris data source used to train the neural network should be preprocessed in order to remove all the non-perfect ephemeris data sets.

One alternative to the ephemeris data binary prediction which should be explored consists in transmitting before its application the broadcasted ephemeris. This solution would require designing a new navigation message structure able to transmit a higher information rate. This solution could be applied to any GNSS signal.

The AWGN channels demodulation performance of the GPS L2C mode NAV data with FEC signal, the GPS L2C mode CNAV data signal and the GPS L5 signal, could be obtained for the traditional Viterbi algorithm and for the combination of the inner and outer channel code algorithm when taking into account the last two PLL carrier phase estimation sources of error:

the Allan deviation noise and the oscillator vibrations. Moreover, in order to completely characterize their demodulation performance, the code delay tracking performance could also be included.

The demodulation performance of the GPS L2C mode NAV data with FEC signal, the GPS L2C mode CNAV data signal and the GPS L5 signal, could be obtained for the traditional Viterbi algorithm and for the combination of the inner and outer channel code algorithm when these signals are transmitted through a mobile channel. The mathematical model of the mobile channel employed to conduct the analysis should be, for the first study, the three states model proposed in this dissertation.

The GALILEO E1 OS and GPS L1C signals AWGN channel demodulation performance obtained when the carrier phase tracking process is noised by thermal noise, dynamic stress error, Allan deviation noise and oscillator vibrations, and when the code delay tracking process is not assumed ideal could be calculated in order to obtain a more realistic demodulation performance and a more exact demodulation performance difference between the signals.

The best punctual GALILEO E1 OS and GPS L1C signals demodulation performance obtained when the signals are transmitted through a mobile channel could be calculated for a mobile travelling at 50 km/h since this speed is the theoretical speed limit reached by a car travelling into a urban environment. Nevertheless, in order to calculate the demodulation performance for this speed case, the mathematical model of the mobile channel would have to be modified depending on the chosen PLL coherent integration time as commented in the first paragraph of this section.

The average GALILEO E1 and GPS L1C signals demodulation performance obtained when the signals are transmitted through a mobile channel should be calculated for a mobile travelling at 5 km/h, 30 km/h and 50 km/h.

Different channel estimation methods could be inspected in order to find a possible channel estimation method which could provide satisfactory demodulation performance in the zones where the PLL does not achieve a lock.

The implementation of a RAKE receiver could be studied. This receiver allows the further exploitation of a frequency-selective channel by means of coherently adding the LOS signal and the different main echoes of the signal. That results into an increase of the available signal C/N_0 at the receiver antenna input and into the introduction of a gain of diversity.

The compression of the useful data such as the ephemeris data and the clock data could be inspected. This compression of the data will allow the satellites to send more information such as the entire ephemeris data of another satellite and not only its almanac. Moreover, the ephemeris of the additional satellites could be source coded with respect to the ephemeris of the broadcasting satellite in order to reduce their size. The compression of the data should not result into an increase of the C/N_0 required to decode the ephemeris data since the size of the channel code protecting the information could be maintained constant.

The introduction of an interleaver on the information source could be analyzed in order to decrease the E_b/N_0 required to obtain a desired BER and/or EER value. The MSBs of the ephemeris and clock data have more predictable values than the LSBs. This means that if a channel code which gives priority to the value of some bits with respect to others when applying its decoding algorithm is used, the introduction of an interleaver over the information message can place the more predictable MSBs in these key positions. Therefore, their more predictable values could be used to improve the decoding performance.

A new channel code could be developed, or several channels codes and decoding strategies could be combined in order to obtain a final decoding algorithm over the ephemeris data which optimize the final EER even if it means a decrease of the BER since the final user is only interested in this first figure of merit.

Further work about the expressions of the likelihood ratio of the bits transmitted into a CSK symbol should be done in order to find more powerful expressions which provide better demodulation performance either with or without an implemented channel code over the transmitted word. On possible way could be to use the determination of one or more bit values as a starting point for the calculation of the remaining bits of the CSK symbol as is done in the bi-orthogonal CSK case.

The introduction of multidimensional channel codes over the CSK symbols and the special CSK demodulation process could be inspected in order to decrease the C/N_0 necessary to obtain a desired BER or WER level of performance. The multidimensional channel code is a combination of several channel codes protecting a transmitted word. The special CSK demodulation process is the use of several FFT points to be inspected in order to calculate the CSK bits likelihood ratios.

Finally, the GALILEO E1 OS signal acquisition process could be optimized in order to decrease the false alarms events and the average acquisition time due to consecutive bins causing false alarms.

Chapter 9. References

[ARINC, 2004] ARINC Engineering Services, *Navstar GPS space Segment/ Navigation User Interfaces, Draft IS-GPS-200D*, Dec 07, 2004

[ARINC, 2005] ARINC Engineering Services, *Navstar GPS space Segment/User segment L5 interfaces, Draft IS-GPS-705*, Sept 22, 2005

[ARINC, 2006] ARINC Engineering Services, *Navstar GPS space Segment/User segment L1C interfaces, Draft IS-GPS-800*, Aug 04, 2006

[ATIS, 2000] ATIS, *ANS T1.523-2001, Telecom Glossary 2000*, General Services Administration Information Technology Service

[BAHL et al., 1974] L. R. BAHL, J. COCKE, F. JELINEK, and J. RAVIV, *Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate*, IEEE Transactions on Information Theory, March 1974

[BASTIDE, 2004] F. BASTIDE, *Analysis of the Feasibility and Interests of Galileo E5a/E5b and GPS L5 for Use with Civil Aviation*, Dissertation, Institut National Polytechnique de Toulouse, 2004.

[BHADESHIA, 1999] H.K.D.H. BHADESHIA, *Neural Networks in Materials Science*, ISIJ International, Vol. 39, No. 10, pp. 966-979, 1999.

[BOSE and RAY-CHADAUDHURI, 1960] R.C. BOSE and D.K. RAY-CHADAUDHURI. *On A Class of Error Correcting Binary Group Codes*. Information and Control 3, pages 68-79, March 1960.

[BURZIGOTTI et al., 2008] P. BURZIGOTTI, R. PRIETO-CEDEIRA, A. BOLEA-ALAMAÑAC, F. PEREZ-FONTAN and I. SANCHEZ-LAGO, *DVB-SH Analysis Using a Multi-State Land Mobile Satellite Channel Model*, Advanced Satellite Mobile Systems, 2008. ASMS 2008. 4th

[CHEN et al, 1991] S. CHEN, C.F.N. COWAN and P.M. GRANT, *Orthogonal Least Squares Algorithm for Radial Basis Function Networks*, IEEE Transactions on Neural Networks, Vol. 2, No. 2, March 1991

[CLINTON, 2000] B. CLINTON, *Improving the Civilian Global Positioning System (GPS)*, May 1, 2000

[DILLARD et al., 2003] G.M. DILLARD, M. REUTER, J. ZEIDLER and B. ZEIDLER, *Cyclic Code Shift Keying: A Low Probability of Intercept Communication Technique*, IEEE Transactions on Aerospace and Electronic Systems, Vol. 39, No 3, July 2003.

[DAFESH et al., 2007] P.A. DAFESH, E.L. VALLES, J. HSU, D.J. SKLAR, L.F. ZAPANTE, C.R. CAHN, *Data Message Performance for the future L1C GPS Signal*, ION GNSS 20th International Technical Meeting of the Satellite Division, 25-28, September 2007, Fort Worth, TX

[ENDSLEY and DEAN, 1994] J.D. ENDSLEY and R.A. DEAN, *Multi-access properties of transform domain spread spectrum systems*, In Proceedings of the 1994 Tactical Communications Conference, Vol. 1, Digital

Technology for the Tactical Communicator, 1994.

[ESA, 2008] European Space Agency, *Galileo OS SIS ICD Draft 1*, February 2008

[ESCHER, 2003] A-C. ESCHER, *Etude de l'apport de l'Hybridation GNSS/INS au contrôle de l'Intégrité du GNSS pour des Applications Aviation Civile*, Dissertation, Institut National Polytechnique de Toulouse, 2003.

[ESCUDIER and POUILLARD, 2009] B. ESCUDIER and J-Y. POUILLARD, *Mécanique Spatiale*, Cours Supaero, 2009.

[EC, 2010] EUROPEN COMISSION, Enterprise and Industry, Space

http://ec.europa.eu/enterprise/policies/space/galileo/programme/index_en.htm

Accessed on: 21/04/2010

[FANTE, 2003] R.L. FANTE, *Unambiguous Tracker for GPS Binary-Offset-Carrier Signal,* ION 59th Annual Meeting/CIGTF 22nd Guidance test Symposium, 23-25 June 2003, Albuquerque, NM.

[FONTANA et al., 2001] R.D. FONTANA, W. CHEUNG, P.M. NOVAK and T.A. STANSELL, *The new L2 Civil Signal*, Proceedings of the 14th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS 2001), September 11 - 14, 2001, Salt Lake City, UT.

[FRIGYES et al., 2001] I. FRIGYES, B.G. MOLNAR, R. VALLET, Z. HERCZKU and Z. BODNAR, *Doppler spread characteristics of satellite personal communication channels*, International Journal of Satellite Communications, 2001.

[GALLAGER, 1963] R.G. GALLAGER, *Low Density Parity Check Codes*, no. 21 in Research Monograph Series. Cambridge, MA: MIT Press, 1963

[GORE, 1998] A. GORE, *Enhancement to the Global Positioning System that will benefit civilian users worldwide*, March 30, 1998

[GORE, 1999] A. GORE, *New Global Positioning System Modernization Initiative*, January 25, 1999

[GRASSBERGER and PROCACCIA, 1983] P. GRASSBERGER and I. PROCACCIA, *Measuring the strangeness of strange attractors*, Physica 9D (1983), pp 189-208.

[HAMMER and SCHAEFER, 1982] A. HAMMER and D.J. SCHAEFER, *Performance Analysis of M-ary Code Shift Keying in Code Division Multiple Access Systems*, ICC-1982, The digital revolution; International Conference on Communications, Philadelphia, June13-17, 1982.

[HAYKIN, 1994] S. HAYKIN, *Neural Networks: A comprehensive Foundation*, Prentice-Hall PTR, Inc. 1994.

[HECKS, 1990] K. HECKS *Bombing 1939-1945: the air offensive against land targets in World War Two.* Robert Hale Ltd., London, 1990.

[HEIN et al., 2001] G.W. HEIN, J. GODET, J-L. ISSLER, J-C. MARTIN, R. LUCAS-

RODRIGUEZ and T. PRATT, *The GALILEO Frequency Structure and Signal Design*, Proceedings of the 14th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS 2001), September 11 - 14, 2001, Salt Lake City, UT.

[IRSIGLER and EISSFELLER, 2002] M. IRSIGLER and B. EISSFELLER, *PLL tracking performance in the presence of oscillator phase noise*, GPS Solutions, Volume 5, Number 4, April 2002.

[JAXA, 2010] Japan Aerospace Exploration Agency, *Interface Specification for QZSS (IS-QZSS) Draft V1.2*, March 19, 2010

[JULIEN, et al. 2005] O. JULIEN, M.E. CANNON and G. LACHAPELLE, *Impact of Future GNSS Signals on Carrier-Phase Tracking*, CD-ROM Proceedings of European Navigation Conference, Munich, 19-22 July 2005.

[JULIEN, 2005] O. JULIEN, *Design of GALILEO L1F Receiver Tracking Loops*, Dissertation, University of Calgary, 2005

[JULIEN et al., 2006] O. JULIEN, C. MACABIAU, J-L ISSLER and L. RIES, *1-bit Processing of Composite BOC (CBOC) Signals*, First CNES Workshop on Galileo Signals and Signal Processing, Toulouse, France, 12-13 October, 2006.

[JULIEN et al, 2007] O. JULIEN, C. MACABIAU, M.E. CANNON and G. LACHAPELLE, *ASPeCT: Unambiguous Sine-BOC(n,n) Acquisition/Tracking Technique for Navigation Applications*, IEEE Transactions on Aerospace and Electronic Systems, Vol. 43, No 1, January 2007

[JULIEN, 2009] O. JULIEN, *GNSS Solutions Tutorial - CN433 Receiver Signal Processing for Future GNSS Signals – Introduction*, ION GNSS 2009.

[KAPLAN and HEGARTYa, 2006] E.D. KAPLAN and C.J. HEGARTY, *Understanding GPS principles and applications*, Second Edition, chapter 'GPS System Segments, Artech House, 2006

[KAPLAN and HEGARTYb, 2006] E.D. KAPLAN and C.J. HEGARTY, *Understanding GPS principles and applications*, Second Edition, chapter 'Satellite Signal Acquisition, Tracking and Data Demodulation, Artech House, 2006

[LEANDERSON and SUNDBERG, 2005] C.F. LEANDERSON and C-E.W. SUNDBERG, *Performance Evaluation of Sequence Map Decoding*, IEEE Transactions on Communications, Vol. 53 No. 3, March 2005.

[LEGRAND, 2002] F. LEGRAND, *Modeles de Boucle de Poursuite de Signaux a Spectre Etalé et Méthode d'Amélioration de la Précision des Mesures Brutes*, Institut National Polytechnique de Toulouse, 2002.

[LOO and BUTTERWORTH, 1998] C. LOO and J.S. BUTTERWORTH, *Land Mobile Satellite Channel Measurements and Modeling*, Proceedings of the IEEE, Vol. 86, No. 7, July 1998.

[LUTZ et al, 1991] E. LUTZ, D. CYGAN, M. DIPPOLD, F. DOLAINSKY and W. PAPKE, *The Land Mobile Satellite Communication Channel –*

Recording, Statistics and Channel Model, IEEE Transactions on vehicular technology, Vol 40, No. 2, May 1991

[MACABIAU et al, 2003] C. MACABIAU, L. RIES, F. BASTIDE, J-L. ISSLER, *GPS L5 Receiver Implementation Issues*, Proceedings of the 16th International Technique Meeting of the Satellite Division of the Institute of Navigation, September 9-12, Oregon 2003

[MACABIAU and JULIEN, 2009] C. MACABIAU and O. JULIEN. *GNSS pour l'aviation civile*. Cours ENAC 2009.

[MACABIAUa, 2009] C. MACABIAU. *Transmission par etalement de spectre*, Cours ENAC 2009.

[MACABIAUb, 2009] C. MACABIAU. *GPS signal – CNS 09 mastere course*, Cours Master ENAC 2009.

[MACKAY and NEAL, 1995] D.J.C. MACKAY and R.M. NEAL, *Good codes based on very sparse matrix*, Cryptography and Coding 5th IMA Conf., C. Boyd, Ed., Lecture Notes in Computer Science, no. 1025, 1995.

[MACKAY, 1999] D.J.C. MACKAY, *Good Error-Coding Codes Based on Very Sparse Matrices*, IEEE Transactions on information theory, Vol 45, No. 2, March 1999.

[MALLESON, 1944] COMMANDER H. ST. A. MALLESON, R.N. *The Decca Navigator System on D-Day, 6 June 1944, An Acid Test*

[MARIÑO et al, 1999] J.B. MARIÑO, F. VALLVERDU, J.A. RODRIGUEZ and A. MORENO, *Tratamiento digital de la señal*, chapter 'La Representacion frecuencial-, Edicions UPC, 1999.

[MARPLEa, 1987] S.L. MARPLE Jr., *Digital Spectral Analysis*, chapter 'Classical Spectral estimation', Prentice-Hall, Inc. 1987.

[MARPLEb, 1987] S.L. MARPLE Jr., *Digital Spectral Analysis*, chapter 'Parametric models of random processes', Prentice-Hall, Inc. 1987.

[MARPLEc, 1987] S.L. MARPLE Jr., *Digital Spectral Analysis*, chapter 'Autoregressive spectral estimation: block data algorithms', Prentice-Hall, Inc. 1987.

[MARPLEd, 1987] S.L. MARPLE Jr., *Digital Spectral Analysis*, chapter 'PRONY's method', Prentice-Hall, Inc. 1987.

[MARTIN et al, 2003] N. MARTIN, V. LEBLOND, G. GUILLOTEL, V. HEIRIES, *BOC(x, y) signal acquisition techniques and performances*, ION GPS/GNSS 2003, 9-12 September 2003, Portland, OR.

[NILL and SUNDBERG, 1995] C. NILL and C-E.W. SUNDBERG, *List and Soft Symbol Output Viterbi Algorithms: Extensions and Comparisons*, IEEE Transactions on Communications, Vol. 43, No. 2/3/4, February/ March/April 2005.

[O'DRISCOLL, 2007] C. O'DRISCOLL, *Performance Analysis of the Parallel Acquisition of Weak GPS Signals*, Dissertation, National University of Ireland, 2007.

[PARKINSON et al, B.W. PARKINSON, T. STANSELL, R. BEARD, K. GROMOV, *A*

1995]	*history of satellite navigation*, Navigation ISSN 0028-1522, 1995
[PEREZ-FONTAN et al, 1998]	F. PEREZ-FONTAN, M.A. VAZQUEZ-CASTRO, S. BUONOMO, J.P. POIARES-BAPTISTA, B. ARBESSER-RASTBURG, *S-Band LMS propagation channel behaviour for different environments, degrees of shadowing and elevation angles*, IEEE Transactions on Broadcasting, Vol. 44, No. 1, March 1998.
[PEREZ-FONTAN et al, 2001]	F. PEREZ-FONTAN, M. VAZQUEZ-CASTRO, C. ENJAMIO-CABADO, J. PITA-GARCIA, E. KUBISTA, *Statistical Modeling of the LMS Channel*, IEEE Transactions on vehicular technology, Vol. 50, NO. 6, November 2001.
[PETERSON and DAVIE, 2003]	PETERSON and DAVIE, *Computer Networks: A Systems Approach*, Third Edition, 2003
[PROAKISa, 2001]	J.G. PROAKIS. *Digital Communications*, chapter 'Introduction', McGraw-Hill, 2001
[PROAKISb, 2001]	J.G. PROAKIS. *Digital Communications*, chapter 'Characterization of Communication Signals and Systems', McGraw-Hill, 2001
[PROAKISc, 2001]	J.G. PROAKIS. *Digital Communications*, chapter 'Optimum Receivers for the Additive White Gaussian Noise Channel', McGraw-Hill, 2001
[PROAKISd, 2001]	J.G. PROAKIS. *Digital Communications*, chapter 'Channel Capacity and Coding', McGraw-Hill, 2001
[PROAKISe, 2001]	J.G. PROAKIS. *Digital Communications*, chapter 'Block and Convolutional Codes', McGraw-Hill, 2001
[PROAKISf, 2001]	J.G. PROAKIS. *Digital Communications*, chapter 'Carrier and Symbol Synchronization', McGraw-Hill, 2001
[PROAKISg, 2001]	J.G. PROAKIS. *Digital Communications*, chapter 'Digital Communications through Fading Multipath Channels', McGraw-Hill, 2001
[ROPER, 2010]	L.C. E. ROPER, *GPS Status and Modernization*, Munich Satellite Navigation Summit, March 9-11, 2010.
[RSA, 2010]	RUSSIAN SPACE AGENCY http://www.glonass-ianc.rsa.ru/pls/htmldb/f?p=202:20:853170395837923::NO Accessed on: 21/04/2010
[SESHADRI and SUNDBERG, 1994]	N. SESHADRI and C-E.W. SUNDBERG, *List Viterbi Decoding Algorithm with Applications*, IEEE Transactions on Communications, Vol. 42 No. 2/3/4, February/March/April 1994.
[SPILKER and ASHBYa, 1996]	J. SPILKER and N. ASHBY, *Global Positioning System: Theory and Applications*, volume 1, chapter 'Introduction and Heritage of NAVSTAR, the Global Positioning System'. AIAA, 1996
[SPILKER and ASHBYb, 1996]	J. SPILKER and N. ASHBY, *Global Positioning System: Theory and Applications*, volume 1, chapter 'Overview of GPS Operation

and Design'. AIAA, 1996

[SPILKER and ASHBYc, 1996] J. SPILKER and N. ASHBY, *Global Positioning System: Theory and Applications*, volume 1, chapter 'GPS signal Structure and Theoretical Performance'. AIAA, 1996

[SPILKER and ASHBYd, 1996] J. SPILKER and N. ASHBY, *Global Positioning System: Theory and Applications*, volume 1, chapter 'GPS Receivers'. AIAA, 1996

[SPILKER and ASHBYe, 1996] J. SPILKER and N. ASHBY, *Global Positioning System: Theory and Applications*, volume 1, chapter 'GPS Navigation Data'. AIAA, 1996

[STEPHENS and THOMAS, 1995] S.A. STEPHENS and J.B. THOMAS, *Controlled-Root Formulation for Digital Phase-Locked Loops*, IEEE Transactions on Aerospace and Electronic Systems, Vol. 31, No. 1, January 1995.

[TSAI, 2009] Y.-R. TSAI, *M-ary Spreading Code Phase Shift Keying Modulation for DSSS Multiple Access Systems*, IEEE Transactions on communications, Vol 57, No 11, November 2009.

[VAN DIERENDONCK et al, 2000] A.J. VAN DIERENDONCK, C. HEGARTY, W. SCALES and S. ERICSON, *Signal Specification for the future GPS Civil Signal at L5*, Proceedings of the IAIN World Congress and the 56th Annual Meeting of the Institute of Navigation, San Diego, CA, June 26 – 28, 2000.

[VITERBI, 1967] A.J. VITERBI, *Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm*, IEEE Transactions on Information Theory, Vol. IT-13, pp. 260-269, 1967.

[VITERBI and OMURA, 1979] A.J. VITERBI and J.K. OMURA, *Principles of Digital Communication and Coding*, McGraw-Hill, NY, 1979.

[WANG et al, 2008] R. WANG, W. ZHAO and G.B. GIANNAKIS, CRC-Assisted Error Correction in a Convolutionally Coded System,IEEE Transactions on Communications, Vol. 56, No.11, November 2008

[WARD and LILLO, 2009] P.W. WARD and W.E. LILLO, *Ambiguity Removal Method for any GNSS Binary Offset Carrier (BOC) Modulation*, Proceedings of the 2009 International Technical Meeting of the Institute of Navigation, 26-28 January 2009, Anaheim, CA.

[WARREN and RAQUET, 2003] D.L.M. WARREN and J.F. RAQUET, *Broadcast vs. precise GPS ephemerides: a historical perspective*, GPS Solutions, 2003

[WEI, 2002] L. WEI, *High-Performance Iterative Viterbi Algorithm for Conventional Serial Concatenated Codes*, IEEE Transactions on Information Theory, Vol. 48, No. 7, July 2002.

[WONG and LEUNG, 1997] A.Y.-C. WONG V. C. M. LEUNG, *Code-Phase-Shift Keying: A Power and Bandwidth Efficient Spread Spectrum Signaling Technique for Wireless Local Area Network Applications*, IEEE Canadian Conference on Electrical and Computer Engineering, 1997.

[YEGNANARAYAN B. YEGNANARAYANA, *Artificial Neural Networks*, chapter

Aa, 2006] 'Basics of Artificial Neural Networks', Prentice-Hall of India, 2006.

[YEGNANARAYAN Ab, 2006] B. YEGNANARAYANA, *Artificial Neural Networks*, chapter 'Activation and Synaptic Dynamics', Prentice-Hall of India, 2006.

[YEGNANARAYAN Ac, 2006] B. YEGNANARAYANA, *Artificial Neural Networks*, chapter 'Feedforward Neural Networks', Prentice-Hall of India, 2006.

Annex A. GNSS signals

In this annex, a general view of the structure of a GNSS signal is presented and the architecture of a GNSS receiver is described. Nevertheless, before presenting the general structure of a GNSS signal, the minimum information data needed to provide positioning service is explained.

A.1. Minimum information data

The main fields needed to provide positioning service can be reduced to two different types of information. On one hand, the receiver-satellite distance; on the other hand, the satellite position with respect to Earth [MACABIAU and JULIEN, 2009] [SPILKER and ASHBYa, 1996].

A.1.1. Pseudo-range measurement

A receiver needs to know the distance between a GNSS satellite and itself. This distance is calculated from the signal propagation time and it is called pseudo-range [SPILKER and ASHBYb, 1996]. The pseudo-range is measured as the time offset between the signal received from the satellite at the antenna output and a local replica of this same signal generated by the receiver taking into account that the receiver ideally knows the time at which the satellite emitted the signal. The pseudo-range is the sum of the distance and the satellite-receiver clock offset. This means that the receiver has to take into account this lack of synchronization in order to correctly estimate the distance between the satellite and the receiver and thus in order to correctly estimate the user position. More specifically, the receiver and the satellite clocks should be both synchronized with the common GNSS general time but the reality is that neither of them is. Moreover, the lack of synchronization of each clock is processed differently due to the very different clock qualities.

In fact, the clock quality is much higher for a satellite clock, which is an atomic clock, than for a receiver clock which is not very accurate [SPILKER and ASHBYa, 1996]. Therefore, whereas the satellite clock bias is modeled as en error over the pseudo-range measurement, the receiver clock bias is so important and particular for each individual clock that the receiver is obliged to estimate the clock bias in order to remove its influence over the pseudo-range measurement.

The receiver cannot estimate its position with only one satellite pseudo-range, which means that the pseudo-range of several satellites should be used. Indeed, assuming a perfect synchronization between the satellites and the receiver clocks, with one satellite pseudo-range, the user position range is a sphere centered at the satellite position. With two satellite pseudo-ranges, the user position range is reduced to the common points (intersection) of the two satellite spheres, a circumference. And with three satellites pseudo-ranges, the user position range is reduced to only two points. Nevertheless, one of the two points is directly discarded because it is not placed on the Earth surface. Therefore, with three satellites a user should be able to estimate its position [SPILKER and ASHBYb, 1996]. See Figure A-1 for a graphical explanation.

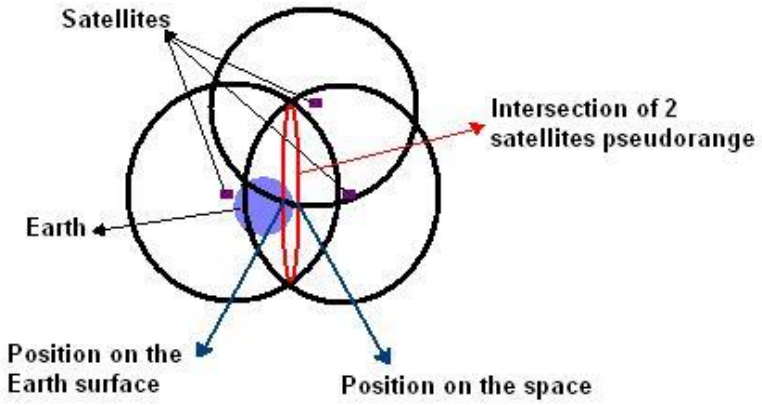

Figure A-1: Receiver positioning scheme with 3 received satellite pseudo-ranges

When assuming that the satellites and the receiver clocks are not synchronized, in order to obtain a 3-Dimensional user position, the receiver needs to measure 4 satellite pseudo-ranges; three satellites for the three user position components (x, y, z) and one for the user clock bias [SPILKER and ASHBYb, 1996].

Moreover, some additional information is required in order to correct pseudo-range measurement errors, such as the influence of the ionosphere, troposphere, etc and the previously commented satellite clock bias [MACABIAU and JULIEN, 2009].

A.1.2. Satellite ephemeris data

The receiver has to know the absolute position of the satellites, because, as it can be observed from Figure A-1, the estimated user position depends on the satellites position. Therefore, in order to obtain an absolute position instead of a relative one, the user has to know what the absolute positions of the satellites are.

A.1.3. Mathematical model of the estimation of the user position

Once the minimum information date needed to estimate a user position has been explained, the mathematical model which allows the user position estimation is given. This mathematical model relates the corrected pseudo-ranges measurements to the receiver position components and to the receiver clock bias [MACABIAU and JULIEN, 2009]:

$$\begin{cases} P^1(k) = \sqrt{(x-x_1)^2+(y-y_1)^2+(z-z_1)^2} + c\cdot\Delta t_u(k) + b^1(k) \\ P^2(k) = \sqrt{(x-x_2)^2+(y-y_2)^2+(z-z_2)^2} + c\cdot\Delta t_u(k) + b^2(k) \\ \vdots \\ P^n(k) = \sqrt{(x-x_n)^2+(y-y_n)^2+(z-z_n)^2} + c\cdot\Delta t_u(k) + b^n(k) \end{cases} \quad \text{(A-1)}$$

Where:

- $P^i(k)$: pseudo-range measurement of satellite *i* after applying the best corrections
- (x, y, z): user coordinates
- (x_i, y_i, z_i): satellite *i* coordinates

- c: velocity of light in vacuum
- Δt_u: user clock bias
- $b^i(k)$: noise plus multipath and additional residual errors after ionosphere, troposphere, satellite *i* clock bias correction, etc.

Note that an increase of the number of measured pseudo-ranges leads to a more accurate estimation of the user position [MACABIAU and JULIEN, 2009]. Moreover, the accuracy of the estimation of the user position is higher for uniform distributions of the line-of-sights of the transmitting satellites [MACABIAU and JULIEN, 2009].

A.2. GNSS signal mathematical model

In this subsection, a general view of the GNSS signal structure is presented in order to better understand the signal processing discussed in the following subsections and along this dissertation. This general view consists in a general mathematical model and in the description of the GNSS signal components providing the necessary information data specified in annex A.1.

The presented mathematical model is a generic basis for any of the existing GNSS signals although, in order to exactly represent each particular GNSS signal, some specific variations have to be introduced to this mathematical model. Nevertheless, the exact mathematical expressions of each GNSS signal have been presented in 0.

A GNSS signal has three main components, the carrier frequency, the PRN code and the navigation data [MACABIAUa, 2009] [SPILKER and ASHBYc, 1996]. Moreover, the modern GNSS signals such as GPS L1C and GALILEO E1 have an additional component, a sub-carrier. The general mathematical model of an emitted GNSS signal is given next

$$r(t) = A \cdot d(t) \cdot c(t) \cdot s_c(t) \cos(2\pi f_0 t - \theta) \tag{A-2}$$

Where:

- A: signal amplitude
- d(t): waveform representing the navigation data.
- c(t): waveform representing the PRN code.
- $s_c(t)$: sub-carrier waveform
- θ: signal carrier phase
- f_0: signal carrier frequency

First, the signal carrier frequency serves to locate the GNSS signal frequency content into its allocated frequency band. GNSS bands are situated in the L-band range frequencies.

Second, the PRN code or pseudo-random noise code is a spread spectrum code which has several functions. The first one is to spread the signal power density. The second one is to allow the transmission of several satellite signals into the same communication channel. This technique is called Code Division Multiple Access (CDMA) and is achieved by imposing that each satellite PRN code is orthogonal with any other satellite PRN code. The third function is to improve the signal resilience to the noise, to the multipath and to the interferences compared to a non-spread BPSK signal. The fourth and last function is to allow pseudo-range measurement by the receiver [SPILKER and ASHBYc, 1996], which is one of the two main

necessary information data presented in annex A.1. In order to allow the pseudo-range measurement, the PRN code autocorrelation function is designed to have a triangular form of a two-chip base width centered at τ=0 and to be about 0 outside the triangle [MACABIAUa, 2009].

$$R_c(\tau)=\begin{cases}1-\left|\frac{\tau}{T_c}\right| & |\tau|<T_c\\ 0 & |\tau|>T_c\end{cases} \quad \text{(A-3)}$$

Third, the navigation data carries the orbital parameters of the satellite broadcasting the signal with respect to the Earth and also carries the additional information necessary to calculate the pseudo-range measurement [SPILKER and ASHBYc, 1996]. Therefore, the navigation data provides the second information necessary to calculate the user position. Moreover, the navigation data also carries the satellites almanacs data, the information concerning the pseudo-range error correction, such as the ionosphere and satellite clock information, etc [SPILKER and ASHBYc, 1996].

Fourth and last, the subcarrier moves further the traditional BPSK power density spectrum around a central subcarrier frequency. Moreover, the introduction of the subcarrier is equivalent to a modification of the PRN code autocorrelation properties, where the central triangle width is reduced and additional triangles are created inside the initial two-chip base width.

Note that the mathematical model (A-3), when the subcarrier term is removed is a model of the autocorrelation of the GPS L1 C/A signal which was designed about 30 years ago [MACABIAUb, 2009]. This signal, despite providing good positioning service performance, has shown to have some limitations in terms of demodulation, tracking and acquisition performance. Among other factors, these limitations are due to the data symbol duration since it bounds the maximum coherent integration time which can be used in the FLL, PLL and DLL [SPILKER and ASHBYd, 1996]. The solution adopted by the modern GNSS signals is to transmit two signals instead of one. One signal contains the navigation data. The other signal is data free in order not to limit the FLL, PLL and PLL coherent integration time. The signal containing the data is called data channel and the signal not containing data is called dataless or pilot channel [ARINC, 2006].

The two signals are combined and transmitted together as one unique signal. Each GNSS signal is transmitted with a specific method. The signals can be transmitted in-phase [ARINC, 2006] [ESA, 2008], in phase-quadrature [ARINC, 2005] or they can be time multiplexed [ARINC, 2004]. The following equation presents the mathematical model of a signal with pilot and data channel transmitted in-phase.

$$r(t)=A\cdot\left[d(t)\cdot c_d(t)\cdot s_{cd}(t)+c_p(t)\cdot s_{cp}(t)\right]\cdot\cos(2\pi f_0 t-\theta) \quad \text{(A-4)}$$

Where:

- $c_d(t)$: data channel PRN code waveform
- $c_p(t)$: pilot channel PRN code waveform
- $s_{cd}(t)$: data channel sub-carrier waveform
- $s_{cp}(t)$: pilot channel sub-carrier waveform

Note that the pilot and the data channels have different PRN codes. The reason is that they need different orthogonal PRN codes in order to allow the receiver to separate them.

A.3. Architecture of a GNSS receiver

The general structure of GNSS receivers can differ from the general structure of traditional telecommunication receivers. Therefore for a better comprehension of the concepts and mathematical expressions given along this dissertation, the general structure of a GNSS receiver is presented in this subsection.

Moreover, since the receiver is only a part of the GNSS user segment, in this subsection, first the general structure of a GNSS user segment is presented and second the block diagram of the demodulation and signal carrier phase tracking processes are described.

A.3.1. General GNSS user segment block diagram

A GNSS user segment consists of five principal components: the antenna, the receiver, the navigator/receiver processor, the input/output (I/O) device such as a control display unit (CDU), and a power supply [KAPLAN and HEGARTYa, 2006]. The block diagram is illustrated below:

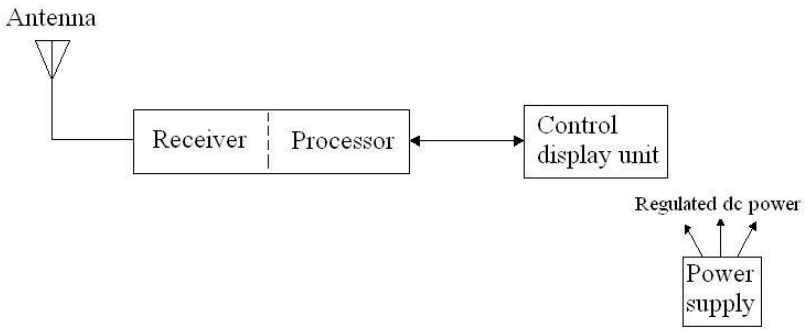

Figure A-2: Principal GNSS user segment components [KAPLAN and HEGARTYa, 2006]

The antenna is in charge of receiving the signal from the satellite. The receiver runs the three basic operations required to obtain the user position, the acquisition, the tracking and the demodulation. The navigator/receiver processor is in charge of controlling and commanding the receiver through its operational sequence, starting with the satellite signal acquisition and following by the satellite signal tracking and the satellite navigation message data demodulation. The I/O device is the interface between the GNSS and the user, and the power supply provides the power to the other user segment parts [KAPLAN and HEGARTYa, 2006].

A more detailed block diagram of a generic GNSS SPS user segment is shown in Figure A-3.

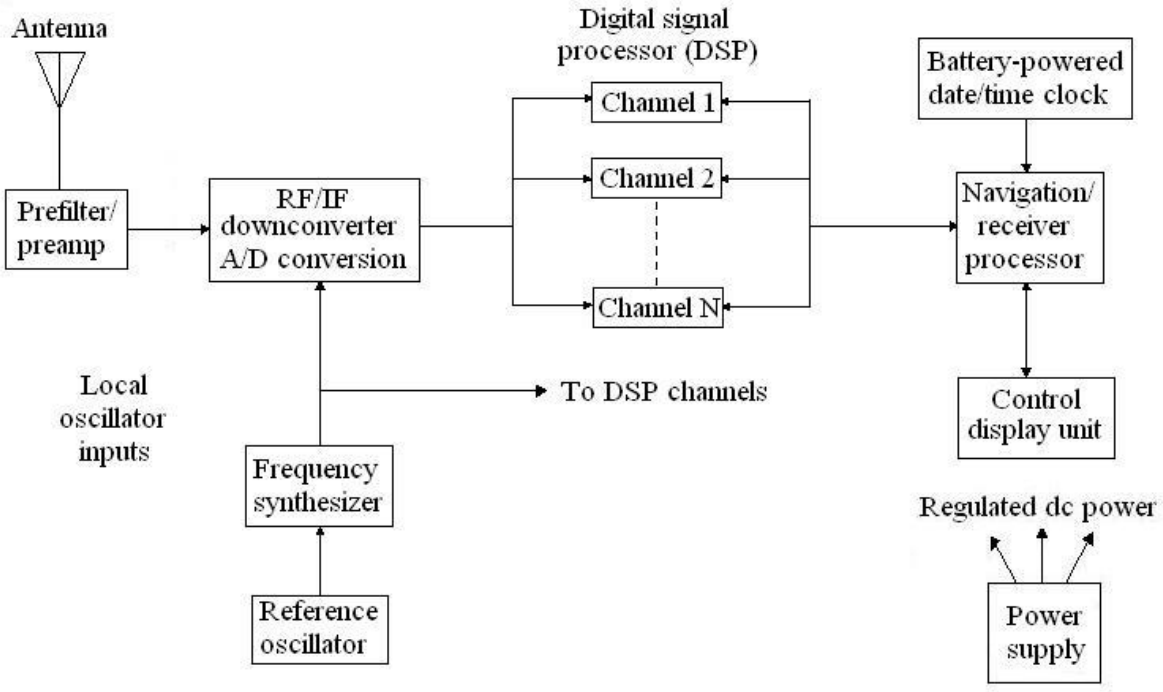

Figure A-3: Generic GNSS SPS user segment [KAPLAN and HEGARTYa, 2006]

In Figure A-3, it can be observed the separate processing performed for the different received satellite signals (marked as channels) and how the navigation/receiver processor command and control the processing on the different channels.

A.3.2. Demodulation receiver block diagram

In this subsection, the block diagram of the demodulation elements is described. However, since each GNSS signal component has a different modulation, in this section, a standard signal without any sub-carrier and BPSK modulated is assumed. More specifically, the described demodulation receiver block diagram is for a GPS L1 C/A receiver, and thus, for any other GNSS signal, the block diagram should be modified.

In this example, the received signal is simplified as we only consider the data channel as in the GPS L1 C/A case. However, this simplification does not affect the demodulator scheme and its associated mathematical model since the pilot channel contribution on the demodulation process is negligible. This statement is justified because the data and the pilot PRN codes are orthogonal, and thus the pilot channel does not influence the data channel.

The block diagram of the demodulator is shown below.

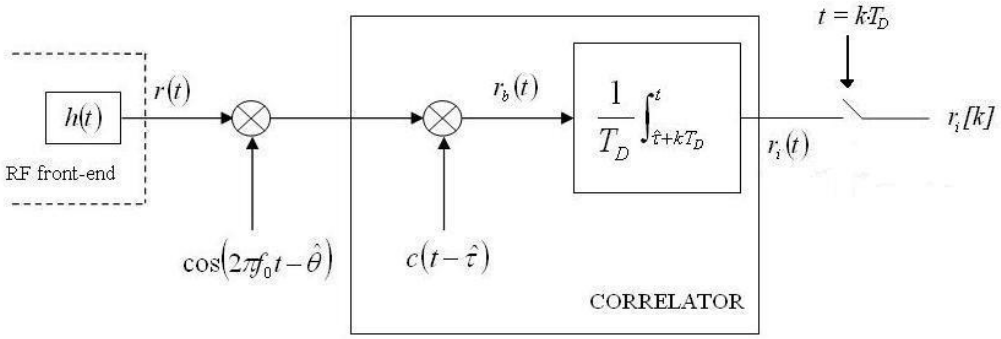

Figure A-4: Block Diagram of the Demodulator [MACABIAUb, 2009]

The elements of the previous diagram block are commented below:

- r(t): received signal after the RF front-end processing
- $\hat{\theta}$: estimated signal carrier phase
- $\hat{\tau}$: estimated code delay
- T_D: symbol duration
- $r_i(t)$: correlator output signal
- $r_i[k]$: numeric correlator output signal (after sampling)

The input signal after the front-end filter is assumed to be [MACABIAUb, 2009]:

$$r(t) = A \cdot d(t-\tau) \cdot c_f(t-\tau)\cos(2\pi f_0 t - \theta) + b(t) \tag{A-5}$$

Where:

- A: signal amplitude
- d(t-τ): navigation data
- c_f(t-τ): PRN code after the RF front-end filtering
- τ: code delay due to the propagation time
- θ: signal carrier phase
- f_0: signal carrier frequency
- b(t): thermal noise after the RF front-end filtering

Therefore, the signal at correlator input is modeled as:

$$r_b(t) = A \cdot d(t-\tau) \cdot c_f(t-\tau) \cdot c(t-\hat{\tau}) \cdot \cos(2\pi f_0 t - \theta) \cdot \cos(2\pi f_0 t - \hat{\theta}) + n(t) \tag{A-6}$$

And the noise at the correlator input is likewise modeled as:

$$n(t) = b(t) \cdot c(t-\hat{\tau}) \cdot \cos(2\pi f_0 t - \hat{\theta}) \tag{A-7}$$

Therefore, at the correlator output the signal and the noise are equal to:

$$r_i[k] = \frac{A}{2} \cdot d[k] \cdot R_{cf}\left(\tau[k] - \hat{\tau}[k]\right) \cdot \cos\left(\theta[k] - \hat{\theta}[k]\right) + n_i[k] \tag{A-8}$$

$$n_i(t) = \frac{1}{T_D} \int_{\hat{\tau}+k \cdot T_D}^{t} n(u)du \tag{A-9}$$

Where:

- $R_{cf}(x)$: PRN code autocorrelation function after RF front-end filtering

And the noise power (P_{ni}) is modeled as:

$$P_{n_i} = \frac{N_0}{4 \cdot T_D} \tag{A-10}$$

Therefore, if a perfect estimation of the code delay and of the signal phase carrier is assumed, equation (A-8) is equivalent to:

$$r_i[k] = \frac{A}{2} \cdot d[k] + n_i[k] \tag{A-11}$$

Finally, from $r_i[k]$ the symbol estimation is immediately made if no channel code is implemented. But, if a channel code is implemented, $r_i[k]$ is used to enter the detector block of Figure C-1.

A.3.3. Signal carrier phase tracking receiver block diagram

In this subsection, the block diagram of the tracking elements is described. More specifically, this tracking block diagram presents the traditional case where the received signal is tracked with a Phase Locked Loop (PLL) [KAPLAN and HEGARTYb, 2006].

The received signal model is simplified as done in the previous section. However, in this case, only the pilot channel is considered because the contribution of the data channel is negligible in comparison with the contribution of the pilot channel correlator output for the carrier phase tracking process. The justification is that the pilot and data PRN codes are orthogonal, and thus the data channel does not affect the pilot channel correlator output. Moreover, the signal sub-carrier is also removed for simplification purposes. The mathematical model of the signal after the front-end filter is presented below:

$$r(t) = A \cdot c_f(t - \tau)\cos(2\pi f_0 t - \theta) + b(t) \tag{A-12}$$

If the signal does not have a pilot channel, as for example the GPS L1 C/A signal, the signal used to track the carrier phase is the data channel.

The PLL block diagram is presented below:

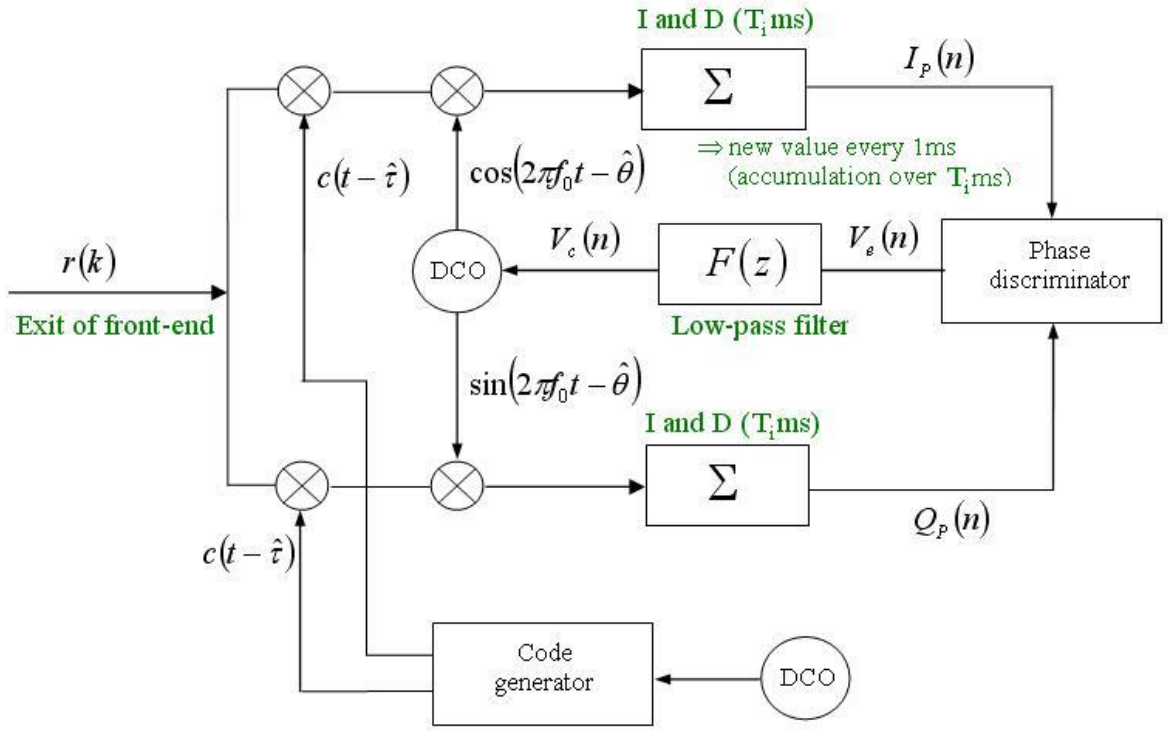

Figure A-5: Phase Locked Loop block diagram [MACABIAUb, 2009]

Where:

- r(k): sampled received signal after the RF front-end processing
- $\hat{\theta}$: estimated signal carrier phase
- $\hat{\tau}$: estimated code delay
- T_i: coherent integration time
- $I_p(n)$: Prompt I channel
- $Q_p(n)$: Prompt Q channel
- $V_e(n)$: Phase discriminator output
- $V_c(n)$: DCO input

In Figure A-5, the output of the front-end block is directly a sampled signal. Therefore, all the blocks presented in Figure A-5 are digital blocks.

Another observation that can be made from the PLL block diagram is that since the signal phase is estimated, it cannot be assured that all the useful signal power is contained in the in-phase channel (I). Therefore, in order to use all the useful signal power during the signal carrier phase tracking process, the phase-quadrature channel (Q) is also employed. Nevertheless, note that if the signal carrier phase is well estimated, all the power should be in the I channel.

The mathematical model of the I and Q channels and their noises are [MACABIAUa, 2009]:

$$I_p(n) = \frac{A}{2} \cdot R_{cf}(\tau - \hat{\tau}) \cdot \cos(\theta - \hat{\theta}) + n_I(n) \tag{A-13}$$

$$Q_p(n) = \frac{A}{2} \cdot R_{cf}(\tau - \hat{\tau}) \cdot \sin(\theta - \hat{\theta}) + n_Q(n) \tag{A-14}$$

$$n_I(n) = \frac{1}{T_i} \int_{\hat{\tau}+k\cdot T_i}^{t} b(u) \cdot c(u - \hat{\tau}) \cdot \cos(2\pi f_0 u - \hat{\theta}) du \Big|_{t=nT_i} \tag{A-15}$$

$$n_Q(n) = \frac{1}{T_i} \int_{\hat{\tau}+k\cdot T_i}^{t} b(u) \cdot c(u - \hat{\tau}) \cdot \sin(2\pi f_0 u - \hat{\theta}) du \Big|_{t=nT_i} \tag{A-16}$$

The description of the blocks forming a PLL is presented next.

The discriminator block is the part of the PLL in charge of measuring the phase estimation error. Two main groups can be distinguished depending on their sensibility to the phase shifts introduced by the data bits [MACABIAUb, 2009].

The Dot Product, or Costas, and the ArcTangent discriminators are not sensitive to the phase changes introduced by the data bits [MACABIAUb, 2009]. These discriminators are used with the GPS L1 C/A signal because this signal does not have a dataless channel and thus, the carrier phase has to be estimated with the signal containing the data. Consequently, the PLL has to remove the phase of the information data ($\pm\pi$) before measuring the error between the received signal carrier phase and the local signal carrier phase.

The Q (or Coherent) and ArcTangent2 discriminators are sensitive to the phase changes introduced by the data bits [MACABIAUb, 2009]. These discriminators are affected by the sign of the data and thus force the PLL to make a change of π on the signal estimated phase each time that the data sign varies. Therefore, these discriminators can only be used on pilot (dataless) channels. This means that they can only be applied on the GPS L2C, GPS L5, GPS L1C and GALILEO E1 signals.

Moreover, the discriminators of the second group are normally used when a pilot channel exist because they have better tracking performance than the discriminators of the first group [JULIEN, 2005]. For example, the loss of lock threshold of the Q and ArcTangent2 discriminators is quite lower in terms of dB-Hz than the loss of lock threshold of the Dot product or the ArcTangent discriminators.

The PLL filter, F(z), has two main functions. First, the filter has to reduce the noise affecting the discriminator output. Second, the filter has to allow the PLL to track the signal phase dynamics [KAPLAN and HEGARTYb, 2006]. Therefore, in order to accomplish these functions, three main parameters are defined: the filter bandwidth (B_L), the filter order (k) and the integration time (T_i). First, in order to eliminate the maximum possible noise, the filter has to have a small bandwidth. However, at the same time, the bandwidth has to be large enough so that the filter does not distort the phase signal measurement. Second, the order of the filter has to be high enough to follow the dynamics of the phase signal but without imposing an excessively complex PLL. Third and last, the filter has to be adapted to the integration time (T_i). From these 3 parameters, the filter coefficients are defined [STEPHENS and THOMAS, 1995].

The DCO or Digitally controlled oscillator is equivalent to digital VCO, which is an electronic oscillator designed to be controlled in oscillation frequency by a voltage input, in this case a sampled input.

Finally two of the sources of error affecting the PLL performance are presented, the thermal noise and the dynamic stress error. Moreover, the definition of PLL loss of lock is given and the choice of the PLL discriminator used for the simulations conducted in this dissertation is justified.

A.3.3.1. PLL Thermal Noise

The error source called thermal noise results from the impossibility of removing all the narrow-band noise existent at the RF/IF output block. Its influence on the carrier phase estimation is explained next.

First, the I and Q channels are used as inputs to the PLL discriminator in order to obtain a measurement of the carrier signal phase estimation error. Second, the PLL discriminator output is filtered and, third and last, the filtered result is used as input to the DCO in order to generate the local signal carrier phase. Nevertheless, since the I and Q channels are corrupted by the RF/IF filtered AWG noise introduced by the transmission channel and the filter cannot remove all the existent noise on the PLL discriminator output, the DCO cannot generate the exact signal phase.

Once the theoretical explanation of the thermal noise source of error has been given, a numerical quantification is presented. It is demonstrated [MACABIAUa, 2009] that the signal carrier phase estimation when only the thermal noise presence is considered can be calculated as:

$$\hat{\theta}[z] = H[z] \cdot \theta[z] + H[z] \cdot N_e[z] \tag{A-17}$$

Where:

- $\hat{\theta}$: signal phase estimation
- θ: signal phase
- H[z]: closed loop PLL transfer function
- N_e[z]: Equivalent noise at the input of the closed loop

$$N_e[z] = n \cdot N_{ne}[z] \tag{A-18}$$

Where:

- N_{ne}[z] : Noise at the discriminator output
- n: Real number which depends on the applied discriminator

From equation (A-17), it can be observed that the contribution of the thermal noise on the phase estimation error is modelled as a noise filtered by the PLL closed loop transfer function. Therefore, the general expression of the power of the carrier phase estimation error depends on the general structure of the PLL and on the thermal noise power. In fact, depending on the type of discriminator used the mathematical model of the carrier phase estimation error changes.

The mathematical expression of the variance of the carrier phase estimation error is particularized for each different discriminator. The error variance for the product discriminator also known as (generic) Costas or product discriminator is:

$$\sigma_\varepsilon^2 = \frac{B_L}{C/N_0} \cdot \left(1 + \frac{1}{2 \cdot T_I \cdot (C/N_0)} \right) \quad \text{(A-19)}$$

Where:

- σ_ε^2: Variance of the carrier phase estimation error due to thermal noise
- B_L: PLL filter Bandwidth
- T_I: Coherent integration Time

The expression for the Q discriminator is:

$$\sigma_\varepsilon^2 = \frac{B_L}{C/N_0} \quad \text{(A-20)}$$

The expressions for the ArcTangent and ArcTangent2 discriminators are very difficult to obtain mathematically; however it has been shown through Monte Carlo simulations that their expression can be approximated by the same expression as the product one [SPILKER and ASHBYd, 1996].

To sum up, the phase estimation error due to the thermal noise at the PLL output is modelled as a noise with a power defined in equation (A-19) or (A-20), with flat power density spectrum bounded by the bandwidth of the closed PLL transfer function.

A.3.3.2. Dynamic stress error

The dynamic stress error is phase jitter caused by the permanent motion of the satellites and the possible receiver motion [IRSIGLER and EISSFELLER, 2002], or in other words, it is phase jitter caused by the signal dynamics. The incoming carrier phase or signal dynamics is modelled as [LEGRAND, 2002]:

$$\theta(t) = \theta_0 + a_1(t) \cdot t + a_2(t) \cdot t^2 + a_3(t) \cdot t^3 + \ldots \quad \text{(A-21)}$$

Where:

- $\theta(t)$: Incoming signal phase
- θ_0: Initial signal phase
- $a_1(t)$: Radial satellite-receiver velocity.
- $a_2(t)$: Radial satellite-receiver acceleration.
- $a_3(t)$: Radial satellite-receiver jerk.

Along this dissertation, the phase jitter is reduced to a simple bias on the estimated signal carrier phase since the majority of signal dynamics can be perfectly tracked by the PLL filter. Indeed, the highest signal dynamics order that can be tracked by the PLL depends on the PLL filter order. This means that a PLL with a 3rd order filter can track up to 2nd order phase variations. We assume in this dissertation that the PLL has a 3rd order filter, so the only signal

dynamics which can affect the signal carrier phase estimation are the satellite and/or receiver acceleration variations, and higher order dynamics. The acceleration variations, called jerk, are the highest order signal dynamics considered in this dissertation.

The mathematical model of the dynamic stress error of a 3rd order PLL is given below [STEPHENS and THOMAS, 1995]. This error represents in meters the bias of phase introduced by the signal dynamics in the generated local carrier phase.

$$\theta_e = \frac{T_I^{\,3}}{K_3} \cdot \frac{dR^3}{dt^3}(m) \tag{A-22}$$

Where:

- θ_e: Dynamic stress error
- R: signal phase delay between the satellite and the receiver (in metres)
- T_I: Integration Time
- K_3: coefficient given by [STEPHENS and THOMAS, 1995], in their description of discrete-update PLL.

This expression is reduced to a constant bias if the signal dynamics generating the dynamic stress error is a constant jerk. We select the worst possible and realistic jerk introduced by the signal dynamics. Therefore, the results obtained by this Ph.D. when this worst jerk value is assumed represent a lower-bound of the tracking and demodulation performance.

For a constant jerk, that is $a_3(t)$, as the only signal dynamics causing the dynamic stress error, the previous expression is equal to:

$$\frac{dR^3}{dt^3} = m \ \ jerks \quad m \in \Re \tag{A-23}$$

$$1\ jerk = 1\ g/s = 9.8\ m/s^3 \tag{A-24}$$

$$\theta_e = 2\pi \frac{T_I^{\,3}}{K_3} \cdot \frac{m \cdot g}{\lambda}(rad) \tag{A-25}$$

Where:

- θ_e: Dynamic stress error
- m: Constant number of jerks
- g: gravity acceleration
- λ: Wavelength of the carrier signal frequency
- T_I: Integration Time
- K_3: coefficient given by [STEPHENS and THOMAS, 1995], in their description of discrete-update PLL.

Note that the final expression (A-25) is expressed in radians; therefore, in order to obtain equation (A-25) from equation (A-22), equation (A-22) is multiplied by (2π radians/1 λ) resulting into the substitution of the meters by the radians.

A.3.3.3. Definition of the PLL loss of lock threshold

Each of one of the discriminator defined in annex A.3.3 measures the difference between the received signal carrier phase and the generated local carrier phase in a different way. However, this measurement is not ideal due to the different sources of error commented in section 3.1.3.2 and due to the discriminator itself. Indeed, the output of a discriminator is a perfect phase difference, or in its default, a value proportional to the phase difference. In doing so, the DCO will be allowed to generate a new local carrier phase which will compensate the previous measured difference [KAPLAN and HEGARTYb, 2006]. The proportional term of the measurement is called discriminator normalization factor (KD), and it has to be removed in order to allow the correct functioning of the DCO. The action to remove this term is called normalize the discriminator output.

However, the proportional measurement of the difference of phases is only possible when the discriminator works in its linear zone; in this zone, its output is proportional to its input. If this is not the case, the discriminator works outside its linear zone and the measurement is distorted. And, although using the discriminator at the edges of this linear zone is allowed but infrequent, an allowed maximum measurement distortion has been set in order to guarantee a minimum degree of accuracy. The linear zone range and the possibility to work outside this zone depend on the signal C/N_0. Therefore, for a determined discriminator, there exists a C/N_0 value under which the discriminator does not work in the linear zone, the maximum allowed measurement distortion is reached and thus the discriminator cannot properly measure the phase difference. This value is called the PLL loss of lock threshold since it is considered that under that value the local carrier phase generated by the PLL is no longer accurate enough.

A graphical example of the variation of the discriminator linear zone is presented in Figure A-6 [JULIEN, et al. 2005]. This figure shows the slopes of two discriminators, Arctangent and Q, assuming that the discriminator output has been normalized. It can be observed that whereas the arctangent linear zone varies as a function of the pilot channel C/N_0, the Q linear zone remains constant. These curves have been calculated assuming the thermal noise presence only.

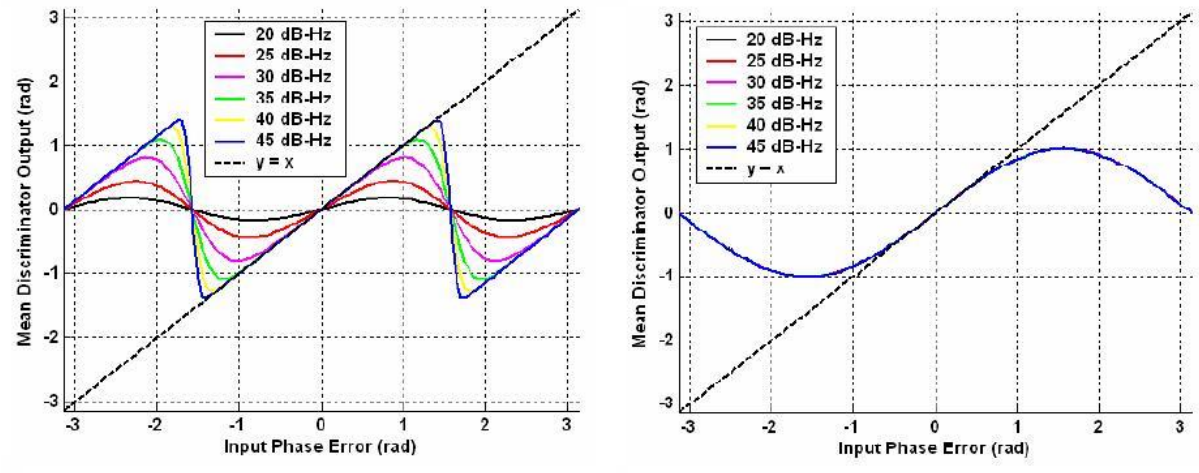

Figure A-6: Mean discriminator output (rad) as a function of the input phase error (rad). Left - Arctangent, Right – Q Discriminator. Coherent integration time equal = 4ms. Each of the curves represents a different value of pilot C/N_0 [JULIEN, et al. 2005].

Therefore, from the determination of the linear zone (depends on the accepted degradation of linearity), from the determination of a maximum accepted percentage of phase errors that can

fall outside the linear zone and from the knowledge that the phase error variance of the thermal noise depends on the pilot channel C/N_0 value, it is possible to calculate the pilot channel C/N_0 value at which the PLL loses its lock.

More specifically, this maximum accepted percentage of phase errors that can fall outside the linear zone has been defined through the phase standard deviation error (σ). Therefore, it has been defined that the PLL loses its lock when 3σ is larger than half the linear zone range [KAPLAN and HEGARTYb, 2006]. In other words, assuming that the tracking error is Gaussian, it is theoretically defined that the PLL conserves its lock for a given C/N_0 (PLL threshold lock) 99% of the time (99% of the time the phase errors fall into the linear zone). Consequently, the pilot channel C/N_0 threshold is the pilot channel C/N_0 value at which half the discriminator linear zone is equal to 3σ.

Therefore, the mathematical expression allowing the calculation of the PLL loss of lock threshold when the 4 sources of noise are taken into account is presented below. The mathematical expression is derived from equations (3-5) and (3-6) [KAPLAN and HEGARTYb, 2006].

$$3\sigma_{\varphi} + \theta_e \leq \frac{L_{\varphi}}{2} \tag{A-26}$$

Where:

- $\sigma_{\varphi} = \sqrt{\sigma_{th}^{\ 2} + \sigma_{Vib}^{\ 2} + \sigma_{Osc}^{\ 2}}$
  - σ_{th}: Thermal noise standard deviation error
  - σ_{Vib}: Oscillator vibrations standard deviation error
  - σ_{Osc}: Allan deviation noise standard deviation error
- θ_e: Dynamic stress error
- L_{φ}: Two-sided discriminator linear tracking region

From equation (A-26), the PLL loss of lock threshold can be directly calculated.

Table A-1 and Table A-2 [JULIEN, et al. 2005] summarize the threshold values for the different discriminators. These tables are calculated for a TCXO oscillator, for a level of signal dynamics jerk equal to 0 and 1 g/s, and for an effective B_L the nearest to 10Hz as possible. The integration time is 20ms because all the signals have a pilot channel.

Note that the values found in the reference [JULIEN, et al. 2005] are expressed in pilot C/N_0 values. Therefore, these values have to be converted to total signal C/N_0 values at the receiver antenna output. The PLL loss of lock threshold expressed in total C/N_0 values is given below.

	Signal	**GPS L2C**	**GPS L5**	**GPS L1C**	**GALILEO E1 OS**	***B_L (Hz)***
Threshold (dB-Hz)	**DOT Product**	29	29	27.25	29	7-10
	Q	23	23	21.25	23	5-6
	Atan	27.5	27.5	25.75	27.5	4-30
	Atan2	26	26	24.25	26	2-30

Table A-1: PLL tracking loss thresholds with a TCXO oscillator and a jerk = 0 g/s

	Signal	GPS L2C	GPS L5	GPS L1C	GALILEO E1 OS	B_L *(Hz)*
Threshold (dB-Hz)	**DOT Product**	35	35	33.25	35	16-17
	Q	29	29	27.25	29	20-30
	Atan	27.5	27.5	25.75	27.5	14-30
	Atan2	26	26	24.25	26	9-30

Table A-2: PLL tracking loss thresholds with a TCXO oscillator and a jerk = 1 g/s

The first observation to make is that the thresholds values of the Arctangent and Arctangent2 discriminators do not change from Table A-1 and Table A-2. However, this is not entirely true. In fact, each presented threshold is calculated from the B_L value inside the specified range of values which provides the lower PLL loss of lock threshold. Therefore, the B_L used for a determined discriminator is different for each analyzed jerk value. And these B_L values can also be different for each different discriminator. Therefore, in reality, for a given B_L, the PLL loss of lock threshold values of the Arctangent and Arctangent2 discriminators increase when a jerk of 1 g/s is applied.

Another observation is that the product and Arctangent discriminators are only applied over the pilot channel although they could also be used on the data channel. This means that the additional data channel power could also be used to track the signal and thus the threshold should be lower. However, the coherent integration time should be adapted to each GNSS signal since its value should become equal to the signal symbol duration. Therefore, it is not possible to directly add the power of the data channel which has not been used to the pilot channel power in order to calculate the new PLL loss of lock threshold.

The last observation to make is that the thresholds presented in the Table A-2 are calculated assuming that the PLL is always working in the linear region. In fact the bias introduced by the dynamic stress error, expressed in equation (A-25), is only valid when the discriminator is in the linear region. However, there is the possibility that the change in the propagation time between the satellite and the receiver induces a phase change greater than the discriminator linear tracking range during one coherent integration period [JULIEN, et al. 2005]. In other words, the dynamic stress error can provoke by itself a loss of lock. Nevertheless, in this dissertation, it is assumed that the PLL receiver is well dimensioned and that the PLL never loses its lock due to a dynamic stress error bigger than expected. In fact, the assumed value of 1 g/s is very large and hardly a user will receive a signal with a larger jerk; therefore, when the jerk is 1 g/s, the only element which could cause a PLL loss of lock is the thermal noise.

Table A-1 and Table A-2 constitute a good reference of PLL tracking threshold values. Nevertheless, these values are calculated from the theoretical noise standard deviation expressions resulting from models where some assumptions were made that are not always fulfilled during the signal tracking process. For example, the B_L used in their calculation is not the 10Hz adopted for this research work.

A.3.3.4. PLL discriminator selection for the simulations

In this subsection, the PLL discriminator selected to conduct all the simulations of the dissertation is presented. This selection was made from the discriminator performance since one of the objectives of this dissertation is to find the performance bounds of the demodulation of the different GNSS signals. The chosen discriminator is the Q or coherent discriminator and the reasons for its selection are given next.

There are two main reasons which justify the Q discriminator selection. The first reason is that since the majority of the GNSS signals analyzed all along this dissertation have a pilot channel (a dataless channel), it is allowed the utilization of a discriminator which is sensitive to the phase jumps of π introduced by the data because there is no data in the pilot channel. Therefore, since these types of discriminators outperform the discriminators which are insensitive to the phase jumps of π, either the Q or the Arctangent 2 discriminators are chosen. The second reason is that the Q discriminator has a better performance than the Arctangent2 discriminator. This justification is given next.

The main differences between the Costas and Q discriminator with respect to the Arctangent and Arctangent2 discriminators are the following. First, the discriminator normalization factor (K_D) of the product and Q discriminators is estimated from the I and Q channels, whereas for the Arctangent and Arctangent2 discriminators, K_D is always 1. Note that this estimation gets worse with low pilot channel C/N_0 levels. Second, whereas the two-sided discriminator linear region varies as a function of the pilot channel C/N_0 for the Arctangent and Arctangent2 discriminators, its value remains constant for the product and Q discriminators.

Therefore, from the previous differences, it is assumed that for low C/N_0 levels, the Costas and Q discriminators have a better performance than the Arctangent and Arctangent2 discriminators. The justification is given next.

For low C/N_0 values, the K_D estimation of the product and Q discriminators is quite affected by the thermal noise. Nevertheless, this estimation can be significantly improved when the K_D estimation is made using long periods of time. Obviously, this assumption is only valid when the signal C/N_0 does not vary significantly during the K_D estimation time. This assumption is valid for an AWGN channel.

For low C/N_0 values, the linear region of the Arctangent and Arctangent2 discriminators is quite reduced. This region length can be enlarged applying longer coherent integration times. However, the coherent integration time cannot be enlarged indefinitely due to the signal phase variations. In fact, the larger the coherent integration time is, the more the received signal carrier phase can vary between the coherent integration time beginning and the coherent integration time end. Indeed, since the PLL can only provide a linearly varying phase during a coherent integration time period, the integration time size is bounded. And this means that the improvement of the tracking performance of the Arctangent and Arctangent2 discriminators is limited by the coherent integration time.

Therefore, since the tracking performance of the ArcTangent2 discriminator is limited by the coherent integration time, and this time cannot be so easily enlarged as the time used to estimate the K_D term, which is the term limiting the tracking performance of the Q discriminator, the Q discriminator outperforms the ArcTangent2 discriminator from the tracking performance point of view in low C/N_0 transmission channels. This statement can be seen in Table A-1 in spite of the Table A-2 values. Note that the threshold values of Table A-1 and Table A-2 for the different discriminators are calculated for different B_L values.

Therefore, due to the limited improvement of the tracking performance of the Arctangent and Arctangent2 discriminators for low C/N_0 levels and the possibility to use long time periods for the estimation of the K_D normalization factor of the products and Q discriminators, we assume that the product and Q discriminators have better tracking performance for low C/N_0 levels than the Arctangent and Arctangent2 discriminators. And thus, we choose to implement in the simulations of this dissertation the Q discriminator.

Annex B. Figures of merit

In this subsection, three figures of merit are defined. They are the SNR, the E_b/N_0 and the C/N_0 and they are used along the entire dissertation. Therefore a good understanding of what these figures represent, what are their units and what are their inter-relationships is necessary to correctly follow the analyses conducted along the dissertation.

B.1. Signal-to-noise ratio (SNR)

The signal-to-noise ratio expresses the relationship between the signal power and the noise power [ATIS, 2000].

$$SNR = \frac{P_{signal}}{P_{noise}} \tag{B-1}$$

More specifically, this figure of merit compares the level of a desired signal to the level of background noise. In other words, the SNR represents how much the noise distorts the useful signal. The higher the ratio, the less disturbing the background noise is. The SNR is normally expressed in dB [ATIS, 2000].

$$SNR\ (dB) = 10 \cdot \log_{10}\left(\frac{P_{signal}}{P_{noise}}\right) \tag{B-2}$$

Finally, this figure of merit is widely used in the telecommunications field.

B.2. Energy per bit to noise density ratio (E_b/N_0)

The E_b/N_0 is an important parameter in digital communication or data transmission. It is a normalized signal-to-noise ratio (SNR) figure, also known as the "SNR per bit" [ATIS, 2000].

In fact, the relationship between E_b/N_0 and SNR is easily calculated. Nevertheless, in order to establish this relationship, some other figures of merit have to be defined. First, the energy per symbol to noise density ratio (E_s/N_0) can be approximated as an equivalence to the signal-to-noise ratio (SNR), where a symbol is the physical representation of a bit or of a group of bits. Indeed, the bits are never directly transmitted through a channel since they belong to the digital domain whereas the channel belongs to the continuous one. Therefore, each bit or group of bits is assigned to a physical waveform in order to adapt the digital signal to its transmission through a channel. This physical waveform is called symbol [PROAKISb, 2001].

The following equation shows the equivalence between the E_s/N_0 and SNR, an equivalence which is only true when a matched filter is used to process the received signal [PROAKISc, 2001]. However, the use of the matched filter is the normal strategy followed by any receiver:

$$\frac{E_s}{N_0} \approx \frac{E_s \cdot R_s}{N_0 \cdot B} = \frac{P_{signal}}{P_{noise}} = SNR \tag{B-3}$$

Where:

- R_s: Symbol transmission rate
- B: Channel bandwidth

Note that the use of the matched filter sets the channel bandwidth equal to the symbol transmission rate [PROAKISc, 2001]. In fact, the channel bandwidth is the bandwidth of the noise where this noise is the signal noise after the signal processing (filters, etc).

Therefore, since it has been defined that a symbol represents a bit or a group of bits, we can directly relate the E_s/N_0 with the E_b/N_0:

$$\frac{E_b}{N_0} = \frac{E_s}{N_0} \cdot \frac{1}{b} \tag{B-4}$$

Where:

- b: Number of bits represented by each symbol

Nevertheless, the bits transmitted through the channel are not the signal information bits. In fact, the information bits are usually encoded by a channel code. The definition of encoding and decoding is given in annex C.3 and C.4. Therefore, E_s/N_0 is normally related to the energy per coded bit to noise density ratio (E_c/N_0) when a channel code is applied on the message. Consequently, equation (B-4) may be expressed as:

$$\frac{E_c}{N_0} = \frac{E_s}{N_0} \cdot \frac{1}{b} \tag{B-5}$$

Finally, the E_c/N_0 can be related to the E_b/N_0 using the code rate, r.

$$\frac{E_b}{N_0} = \frac{E_c}{N_0} \cdot \frac{1}{r} \tag{B-6}$$

This code rate, r, determines the quantity *Y* of coded bits needed to represent the quantity *X* of information bits [PROAKISe, 2001].

$$r = X / Y \tag{B-7}$$

To summarize, the final relationship between the SNR and the E_b/N_0 is expressed in equation (B-8) where the E_b/N_0 unit is dB [ATIS, 2000]:

$$\frac{E_b}{N_0}(dB) = SNR(dB) - 10 \cdot \log_{10}(b) - 10 \cdot \log_{10}(r) \tag{B-8}$$

Note that, in case that the signal message does not have any channel code, the code rate is equal to 1 and thus the influence of the channel code is cancelled.

This figure of merit is also widely used in the telecommunication field.

B.3. Carrier-to-noise density ratio (C/N_0)

The C/N_0 is the ratio between the received carrier power and the receiver noise density [ATIS, 2000]. This figure of merit compares the level of a desired signal to the level of the background noise power density. The advantage of the C/N_0 in comparison to the SNR is that this former figure of merit does not take into account the channel bandwidth (B), and thus it is valid for any signal processing. Note that the signal characteristics and the signal processing inside the receiver determine the channel bandwidth. The relationship between the C/N_0 and the SNR is shown in equation (B-9).

$$\frac{C}{N_0} = SNR \cdot B \tag{B-9}$$

Therefore, the C/N_0 is expressed in dB-Hz [ATIS, 2000]. Moreover, the E_b/N_0 and C/N_0 can be related as shown below:

$$\frac{C}{N_0}(dB-Hz) = \frac{E_b}{N_0}(dB) + 10 \cdot \log_{10}(R_s) + 10 \cdot \log_{10}(b) + 10 \cdot \log_{10}(r) \tag{B-10}$$

Finally, this figure of merit is widely used in the satellite navigation field.

Annex C. GNSS fundamental processes

In this annex, the definitions of the three main processes conducted by a GNSS receiver are described. These processes are the demodulation of the navigation message, the tracking of the signal and the acquisition of the signal. Additionally, since the demodulation of the navigation message normally implies the decoding of the channel code implemented over the message, the encoding and decoding process of a message are also described.

The descriptions given in this annex are generalized for any type of signal, not only the GNSS signals.

C.1. Signal Demodulation

The demodulation process of a digital communication of a signal consists in estimating the transmitted signal bits values from the antenna output received signal. In other words, the receiver has to decide from the received signal values which is the more probable bit being transmitted [PROAKISc, 2001].

The communication of a digital signal is achieved by transmitting either a waveform or a specific combination of waveforms, where each waveform or specific combination of waveforms represents either a bit or group of bits. This association between bits and waveforms is necessary in order to adapt the digital nature of the bits to the continuous nature of the channel [PROAKISc, 2001].

Therefore, due to the adaptation of the bits to the channel, the receiver must measure the received signal waveforms and/or the received signal waveform amplitudes in order to estimate which waveforms or waveforms amplitudes have been transmitted. And from the estimation of the received signal waveforms and the received signal waveform amplitudes, the transmitted bits can be estimated since the assignation between waveforms and bits at the emission is known [PROAKISc, 2001]. This dissertation calls symbol the set of waveforms and/or the waveform amplitudes assigned to a bit or a group of bits. And the action of assigning a bit or group of bits to a determined signal waveform is called to modulate [PROAKISb, 2001].

Therefore, for an ideal channel, a channel that transmits the signal between the transmitter and the receiver without any kind of distortion or any kind of noise, the estimated symbols at the receiver are identical to the transmitted ones. This means that the transmitted bits are perfectly recovered at reception.

However, in reality, even the simplest channel, the AWGN channel, distorts the received signal after the communication. Therefore, the received signal waveforms and/or the received signal waveform amplitudes no longer correspond to a signal waveform and/or a signal waveform amplitude assigned to a bit or group of bits. Consequently, the receiver has to decide from the received signal waveforms or received signal waveform amplitudes which were the original transmitted symbols. The received signal waveforms or received signal waveform amplitudes are called the channel observations and play an important role into the demodulation of the signal [PROAKISc, 2001].

There are several criterions that can be used to decide which symbol has been transmitted and the most common is the Maximum a Priori (MAP) criterion [PROAKISc, 2001]. This criterion searches the probability of transmitting the symbol $s_i(t)$, associated to a bit or group

of bits, when the signal r(t) has been received. Equation (C-1) represents its mathematical expression:

$$P\left(s_i(t)\middle|r(t)\right) \tag{C-1}$$

Where:

- P(x|y): Probability of *x* when *y* occurs

Therefore, the receiver decides that the symbol *i* has been transmitted if the symbol *i* has the largest MAP probability among all the possible symbols [PROAKISc, 2001]:

$$s_i(t)\ transmitted \Leftrightarrow \quad P\left(s_i(t)\middle|r(t)\right) > P\left(s_j(t)\middle|r(t)\right) \quad j = 0..L-1, j \neq i \tag{C-2}$$

Where:

- L: Number of symbols

However, the probability in (C-1) cannot be directly obtained and thus, this criterion is related to the Maximum Likelihood (ML) criterion by the Bayes theorem [PROAKISc, 2001]. The ML criterion searches the probability of receiving the signal r(t) when the symbol $s_i(t)$ has been transmitted. This theorem has an equivalent expression for the probability density function.

$$f\left(r(t)\middle|s_i(t)\right) \tag{C-3}$$

Where:

- f(x): probability density function of x

Therefore, the final MAP expression is [PROAKISc, 2001]:

$$\begin{array}{c} s_i(t)\ transmitted \\ \updownarrow \\ f\left(r(t)\middle|s_i(t)\right)\cdot P\left(s_i(t)\right) > f\left(r(t)\middle|s_j(t)\right)\cdot P\left(s_j(t)\right) \quad j = 0..L-1, j \neq i \end{array} \tag{C-4}$$

In equation (C-4), it can be observed that apart from the observation of the channel, r(t), another factor influences the final symbol decision, the term $P(s_i(t))$. This factor is the a priori information of the signal communication, and altogether with the channel observation, they are the major actors of the symbol estimation process [PROAKISc, 2001].

Finally, one last intuitive concept can be extracted from equations (C-3) and (C-4). These equations show that the detected symbol will be the symbol which is the most similar to the received signal waveforms but weighted by the a priori information.

C.2. Demodulation Performance

The demodulation performance expresses how well a transmitted signal is demodulated: it indicates the quantity of errors made when demodulating the received signal [PROAKISc, 2001]. Obviously, the demodulation performance depends on the transmission channel, the employed demodulation technique, the signal characteristics, etc. All these factors define a scenario with fixed characteristics except for the signal power and the noise power. Therefore, the demodulation performance of a given scenario is given by the quantity of decision errors as a function of the signal power and the noise power [PROAKISc, 2001].

The relationship between the signal power and the noise power is provided by the three different figures of merit already defined in 0: the SNR, the E_b/N_0 and the C/N_0. The E_b/N_0 is the most used figure in the telecommunications field. However, the C/N_0 is also used along this dissertation since it is the reference figure of merit in the satellite navigation field.

The quantity of errors made when demodulating the received signal can be expressed with three different parameters. Each parameter defines the percentage of errors of a different information unit. The parameters are the Bit Error Rate (BER), the Word Error Rate (WER) and the Ephemeris Error Rate (EER).

To sum up, the demodulation performance of the different GNSS signals analyzed in this study is expressed as the BER, the WER and the EER as a function of either the E_b/N_0 or the C/N_0.

The BER, the WER and the EER are detailed and their influence is commented below.

C.2.1. Bit Error Rate (BER)

The BER is equal to the number of wrong demodulated bits divided by the number of total transmitted bits [ATIS, 2000]. Therefore, the BER expresses the percentage of bits that would be wrong demodulated during the communication of a specific signal through a determined channel.

$$BER = \frac{Nmmber\ of\ Erroneous\ Bits}{Total\ Number\ of\ Bits} \tag{C-5}$$

Besides, the BER can be calculated either over the information bits or over the coded bits. The most significant figure of merit is the BER of the information bits since the final user is only concerned by the information, and not on how this information is transmitted [PROAKISe, 2001]. Moreover, the BER of the information bits provides more information than the BER of the coded bits as detailed hereafter.

The BER of the coded bits shows the influence of the transmission channel and the influence of the signal adaptation to the channel (modulation, etc). Moreover, this BER indicates the influence of the energy per bit and, depending on the figure of merit associated to the BER (C/N_0 for example), the symbol transmission rate influence. But, in addition to all these influences, the BER of the information bits also shows the additional protection provided by the channel code implemented over the message.

To sum up, the BER of the information symbols provides the percentage of error on transmitted information bits, which is the only interesting value to the final user, and shows the improvement of the signal demodulation due to the introduction of a channel code. Remember that each GNSS signal has a distinct channel code.

Finally, the BER is a great indicator of the signal design quality in terms of demodulation performance. However, the BER does not specify the level of E_b/N_0 or C/N_0 necessary to demodulate a determined information message with a given percentage of success.

C.2.2. Word Error Rate (WER)

The information transmitted during a communication is not usually a random stream of bits, but rather a structured unit. In fact, all the bits representing an information field, such as the satellite inclination, are usually transmitted together. And, normally, the information fields are

grouped depending on their importance, on their type, etc, forming different units of information. These information units have been called packets or words along this dissertation and a different channel code can be implemented on each different word. Moreover, these words are grouped again forming new structures such as subframes or frames. Then, the process of structuring the information in words or subframes is applied over and over again until a periodic repetition of the structure information is achieved.

Additionally, each word usually carries some information which enables the receiver to detect whether any of the bits forming the word are wrong. Therefore, the receiver is usually able to know if the demodulated word information contains any error. This means that it is possible to measure the percentage of times that a word is correctly demodulated. And, even if a word did not carry any information allowing the verification of its integrity, it is always interesting to know the percentage of times that a word can be demodulated without error.

Therefore, the WER is a very interesting figure of merit in terms of percentage of times that an information field can be demodulated without errors. Moreover, the WER shows the influence of the word size because when a word is larger, a better channel code can be implemented. This statement was demonstrated by Shannon [PROAKISd, 2001].

More specifically, the WER is equal to the number of wrong demodulated words divided by the number of total transmitted words [ATIS, 2000].

$$WER = \frac{Nmmber\ of\ Erroneous\ Words}{Total\ \ Number\ of\ Words} \tag{C-6}$$

Therefore, the WER expresses the percentage of words that would be wrong demodulated during the communication of a specific signal though a determined channel.

Note that the WER can be calculated for each different word transmitted by the signal. However, if the only difference among words is their position inside the signal structure, their WER is the same.

C.2.3. Ephemeris Error Rate (EER)

During the description of the WER, it was said that the information fields are usually grouped into words. Moreover, it was commented that this regrouping is done among information fields providing the same type of information. This is the case of the different GNSS signals, where the information fields containing the satellite broadcasting ephemeris set or clock error corrections are transmitted into the same words. The entire ephemeris data set can be divided into several words as it is done for GPS L1 C/A, GPS L2C [ARINC, 2004], GPS L5 [ARINC, 2005] and GALILEO E1 [ESA, 2008] signals, or all the information can be grouped into only one word as it is done for GPS L1C [ARINC, 2006].

In any case, since the ephemeris and clock data are the only information necessary to obtain the satellite position and thus the final user position, the only important figure of merit from the point of view of the final user is the EER.

Consequently, in this dissertation, the EER is defined as the percentage of times that a receiver can correctly demodulate the signal ephemeris data set. And the EER is calculated by dividing the number of wrong demodulated ephemeris data sets by the number of total transmitted ephemeris data sets.

$$WER = \frac{Number\ of\ Erroneous\ Ephemeris\ Data\ Sets}{Total\ Number\ of\ Ephemeris\ Data\ Sets} \quad \text{(C-7)}$$

Note that the Ephemeris Error Rate and the Word Error Rate are equal for the GNSS signals transmitting the ephemeris set and clock data into only one word. Therefore, for the signals transmitting the satellite ephemeris data set into more than one word the only interest of the WER is the following: to compare the value of either E_b/N_0 or C/N_0 necessary to obtain a given WER value with the value of either E_b/N_0 or C/N_0 necessary to obtain a EER value equal to the WER value. From this comparison, it can be concluded whether the ephemeris data set division into several words has improved the signal demodulation performance.

C.3. Signal Encoding

The encoding of the signal message is the application of a channel code over the signal message. This process consists in introducing, in a controlled manner, some redundancy in the binary information sequence of the signal. This redundancy can be used at the receiver level to overcome the effects of noise and interference encountered in the transmission of the signal through the channel [PROAKISd, 2001]. In other words, from the information bits point of view, the encoding process generates some new bits which represent the same content as the information bits. Therefore, these new bits are redundant since the original information content is already transmitted by the information bits. And the inclusion of these new binary bits into the transmitted word means that if some of the original information bits are corrupted or lost at the reception, the receiver will still have the possibility to recover the message since the lost information could be reconstructed from these new generated bits.

Moreover, in addition to generating the new bits and introducing them to the transmitted word, the information bits can be modified during the encoding process. Therefore, the encoding process can be seen as a generation of a new binary word from the original binary information word, where the original word can be preserved or not. These bits of the new binary word are called coded bits and the new binary word is called coded word [PROAKISd, 2001]. Besides, note that the coded binary word is the binary word used to modulate the signal. Therefore, when a channel code is applied, equation (B-5) is used instead of equation (B-4).

Additionally, the ratio between the size of the original information word and the size of the coded word is called code rate (r). Its expression has already been given in equation (B-6).

C.4. Signal Decoding

The decoding of the received signal message is the action of exploiting the available redundancy contained into the received message in order to correct the perturbations introduced by the channel [PROAKISd, 2001]. Therefore, due to this correction, the receiver can try to recover the original binary information word.

Nevertheless, there are two types of decoding process depending on the decoder block input. In fact, the received signal passes through a demodulator and a detector before entering the decoder [PROAKISd, 2001], as depicted below.

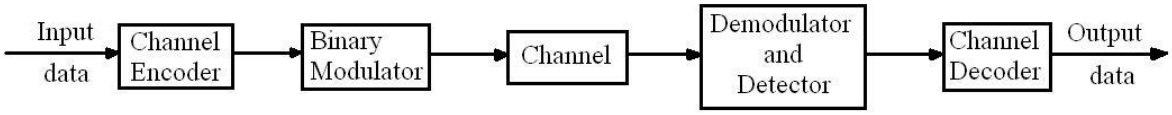

Figure C-1: Digital Communication block scheme

The demodulator block processes the received channel-corrupted waveform and reduces each waveform to a scalar or to a vector that represents an estimate of the transmitted data symbol. The detector block quantizes the output of the demodulator block and from this quantization the two possible decoding types are defined [PROAKISd, 2001].

If the number of quantization steps is equal to the number of possible symbols, the detection process has already chosen the symbol and thus has set the binary value of the coded bits. Therefore, since these coded bits are directly fed to the decoder block, this type of decoding is called *hard decision*. If the level of quantization is infinite, or much larger than the number of symbols, the value of the decoded bits is not binary but real. Therefore, this type of decoding is called *soft decision* [PROAKISd, 2001].

C.4.1. Strategies of channel codes use

There are two different strategies of channel code use. The first one is called Forward Error Correction (FEC) and consists in using the redundancy transmitted into the signal message in order to correct any possible errors introduced by the channel [PROAKISe, 2001].

The other use is called Automatic Repeat-Request (ARQ). It consists in detecting if any error has been introduced by the transmission channel into the received signal and in asking the transmitter to resend the information if some error was detected [PETERSON and DAVIE, 2003]. Along this Ph.D. manuscript, the action of using the word channel code to detect if the word contains any erroneous bit has been called verification, and the verification process is called verify the channel code. Therefore, if a word fails the channel code verification, it means that the word contain erroneous bits. And, if a word passes the channel code verification, it means that the word is error free.

In satellite navigation systems, the ARQ is not possible since the transmitters are only a few satellites whereas the number of receivers is much bigger. Therefore, a single receiver cannot request the retransmission of the navigation message. Nevertheless, the ephemeris data set is continuously repeated during a period of time, and when another set replaces the previous one, the user no longer needs the first ephemeris data set. Therefore, even if the user does not correctly decode an ephemeris data set, a retransmission request is unnecessary since the needed information is continuously broadcasted by the satellite [SPILKER and ASHBYb, 1996]. However, even if the ARQ strategy is not implemented, each GNSS signal has a channel code allowing the verification of the received words as seen in 0. And normally, if the word verification fails, the word is discarded.

Finally, in addition to the channel code used to detect whether the word is error free or not, the majority of GNSS signals have another channel code used to implement the FEC as it has also been seen in 0.

C.5. Decoding Performance

Two different decoding performance can be analyzed, one for the FEC strategy and the other for the ARQ strategy.

The decoding performance of the ARQ mode consists in measuring the number of erroneous words which verify the implemented channel code in relation to the total number of received words [PETERSON and DAVIE, 2003].

ARQ Perfo = Wrong words passing the code verification / Total Words (C-8)

This decoding performance is ignored in this dissertation mainly since the performance of the channel code implemented on the GNSS signals has already been analyzed in the literature.

The decoding performance of the FEC mode indicates how well the received signal is decoded [PROAKISe, 2001]. Therefore, this performance is represented with the BER, the WER and/or the EER of the received signal as a function of the E_b/N_0 [PROAKISe, 2001].

In order to completely analyze the decoding performance, the BER of the signal when a channel code is implemented has to be compared with the BER of the signal when no channel code is implemented [PROAKISe, 2001]. More specifically, from the plot of the two BER curves, the gain of E_b/N_0 for the signal implementing a channel code to reach the same BER value as the signal not implementing a channel code can be quantified. Note that this comparison only analyzes the improvement introduced by the code and does not reflect any other signal characteristic.

Finally, the decoding performance of the FEC mode can also be expressed as a function of the C/N_0. However, in this case, the performance not only shows the influence of the channel code, but also the symbol transmission rate, etc. Therefore, the decoding performance of the FEC mode when expressed as a function of the C/N_0 is considered equivalent to the demodulation performance all along this dissertation.

C.6. Signal Carrier Tracking

The carrier tracking process of a digital signal consists in estimating the instantaneous carrier frequency and the carrier phase of the received signal [KAPLAN and HEGARTYb, 2006]. There are two main methods employed to achieve this estimation, and both methods have been used along this dissertation. The first one consists in estimating the carrier frequency and carrier phase signal using a FLL and a PLL [PROAKISf, 2001], and the second one consists in estimating the transmission channel [PROAKISg, 2001]. Nevertheless, the satellite navigation systems traditionally use FLLs and PLLs [KAPLAN and HEGARTYb, 2006]. Therefore this method is more widely presented and studied along this dissertation than the channel estimation technique. A brief explanation of both methods is given below.

The FLL and PLL methods consist in the following. The FLL (Frequency Locked Loop) has to find the signal carrier frequency in order to down-convert the signal to baseband, and the PLL (Phase Locked Loop) has to find and subtract the instantaneous carrier phase of the signal in order to recover the maximum useful signal power [PROAKISf, 2001].

Nevertheless, in this thesis, the influence of the carrier tracking process is reduced to the study of the PLL. The reason is that the FLL is able to recover the signal carrier frequency with a sufficient accuracy to neglect its impact over the demodulation performance for the analysed levels of C/N_0 of this dissertation [MACABIAU et al, 2003]. However, the phase tracking errors must be taken into account.

The channel estimation consists first in estimating the phase and the amplitude introduced by the channel on the received signal. Second, once the estimation is made, the method consists in equalizing these introduced phase and amplitude in order to maximize the received signal SNR [PROAKISg, 2001]. Different techniques with different performance are used to estimate the signal phase and amplitude.

The different performance of phase tracking methods is presented in the following subsections.

C.6.1. PLL tracking performance

The performance of the PLL in the satellite navigation domain is measured by three different parameters: the PLL tracking loss threshold, the tracking accuracy and the cycle slip rate [KAPLAN and HEGARTYb, 2006].

The first one is the PLL tracking loss of lock threshold and it is expressed in terms of C/N_0. This threshold indicates the level of C/N_0 at which the PLL is no longer able to estimate the signal phase. This parameter is detailed in subsection annex A.3.3.3.

The second parameter defining the PLL tracking performance is the tracking precision: the bias and the variance of the carrier phase estimation error. This error is introduced by several noise sources which are presented in subsection 3.1.3.2.

The third parameter is the cycle slip rate. The cycle slip rate is the percentages of times that a phase cycle slip occurs during a second. However, this part of the tracking performance, although it affects the demodulation performance, does not have any relationship with the objective of this research work and the conducted studies. Therefore, this parameter is not analyzed and is not commented along this dissertation.

Finally, the influence of the PLL tracking performance is analyzed along this dissertation through its impact on the demodulation performance since the demodulation performance study is the main objective of this research work.

C.6.2. Channel estimation performance

The channel estimation performance is measured with one of the PLL tracking performance parameters. This parameter is the phase accuracy, which is represented by the bias and variance of the signal carrier phase estimation error and of the signal amplitude estimation error. The bias and the variance are presented as a function of the C/N_0. Nevertheless, in this dissertation, the channel estimation performance is also shown through its impact on the demodulation performance as done for the PLL tracking performance.

C.7. Signal Acquisition

The acquisition process of a GNSS signal is a detection process carried out as a search. This search process consists in finding at which frequency and at which code delay the GNSS signal is being received [KAPLAN and HEGARTYb, 2006].

In order to do so, the frequency domain is divided into bins and the same is done for the code delay domain: the combination of one code bin and of one frequency or Doppler bin is called a cell. Therefore, the acquisition process consists in detecting which cell contains the received signal carrier frequency and code delay. And thus, for each cell a signal replica with the

determined carrier frequency and code delay is generated and used to calculate the correlation with the received signal. Finally, the decision of which cell represents better the signal carrier frequency and code phase depends on the cell correlation results.

Several strategies can be used to select the cell. The receiver can explore the entire two-dimensional domain and choose the cell having the biggest correlation output [O'DRISCOLL, 2007]. The receiver can also explore the two-dimensional domain beginning from an arbitrary cell and ending with the first cell having a correlation output larger than a given threshold [KAPLAN and HEGARTYb, 2006]. This cell is then tested for tracking. Figure C-2 illustrates a graphical scheme of the GPS L1 C/A acquisition process:

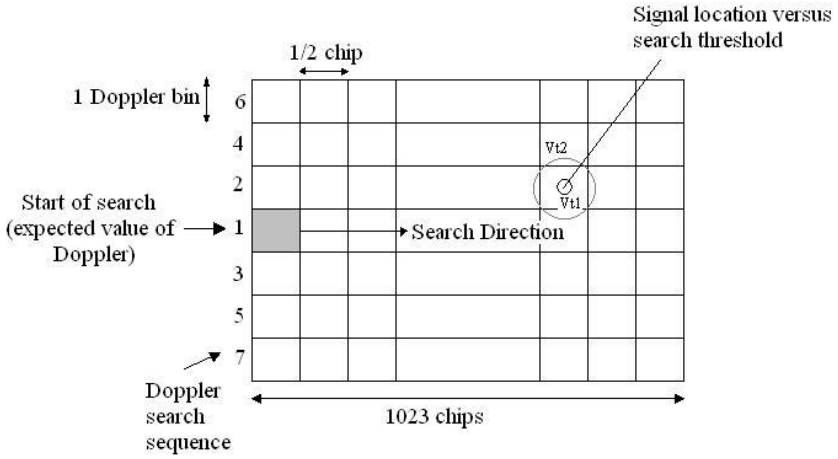

Figure C-2: GPS L1 C/A acquisition process

Once an overview of the acquisition process is presented, the frequency and time domain bin division is described next. The carrier frequency search domain is required because the received signal carrier frequency can be different from the expected nominal signal carrier frequency due to the Doppler Effect. Therefore, the frequency search bounds are determined by the maximum Doppler shift that can affect the received signal [KAPLAN and HEGARTYb, 2006]. The code delay search domain or time search domain is required because the receiver does not know a priori which is the received signal delay, and thus cannot know the received code delay. Consequently, the code delay domain bounds are all the possible code delays, e.g. 1023 code delays for GPS L1 C/A. Nevertheless, the number of code delay bins is not equal to the number of code chips and the reason is presented next.

Since the receiver can only obtain a correlation value for a determined delay and not for a determined period, each bin represents a point of time rather than an interval of time. In other words, the code delay division is in reality a sampling process of the time delay domain. Therefore, it can be that none of the sampled code delay points is exactly equal to the received signal code delay but rather that the received signal code delay is between two sampled code delay points [KAPLAN and HEGARTYb, 2006]. This means that for this latter case, the bins where the received signal has its largest power are the bins having the code delay nearest to the received signal code delay. Nevertheless, even for these best bins, the useful signal power is lower than the useful signal power when the code delay is perfectly estimated since the signal PRN code autocorrelation is not evaluated at its peak. Therefore, the sampling criterion of the code delay domain is defined so that, in the worst case, the useful signal power at the correlator output loses a given proportion of correlation power [BASTIDE, 2004], normally 2.5 dB. Note that since the code delay sampling depends on the PRN code autocorrelation

function and on the signal modulation, the GPS L1 C/A signal has a different code delay sampling than the GPS L1C signal.

Moreover, the same principle can be applied to the Doppler search, where the frequency sampling is set in order to guarantee a fixed loss of signal power at most, normally 0.9 dB [BASTIDE, 2004].

The code and frequency bounds defined in the previous paragraphs correspond to the case where no previous information is known. However, if for some reason the receiver already has some signal information, the search bounds are reduced to 3σ of the previous known frequency and code delay tracked values uncertainties. This can be the case where the receiver is trying to acquire the signal only moments ago after losing the PLL lock [KAPLAN and HEGARTYb, 2006].

Finally, one last important element of the acquisition process is the false alarm rate. A false alarm occurs whenever a cell not representing the received signal carrier frequency and code delay is selected as the cell containing the useful signal [KAPLAN and HEGARTYb, 2006]. Until now, the only considered source of error was the sampling limitation since the receiver cannot inspect an infinite number of different code delays for example. However, there are other error sources which can influence the cell selection. These sources are, for example, the thermal noise, the interferences and the signal dynamics. These sources of error can change the cell correlation level. This means that cells not containing the useful signal can have their amplitude level increased and cells containing the useful signal can have their amplitude level decreased. In both cases the final cell choice can be corrupted and thus a false alarm can appear. Therefore, due to these sources of errors, the acquisition process has to set a decision threshold on the correlation values to guarantee a given maximum level of false alarms. Moreover, note that due to this introduction of the decision, a cell containing the useful signal can be missed.

C.7.1. Acquisition performance

The acquisition performance is measured by the statistics of the time required to acquire the signal as a function of the C/N_0 at the antenna output. This time is calculated for a determined false alarm probability in order to compare different acquisition parameters which are the cell selection criteria and the acquisition techniques.

Note that the difference among the acquisition parameters is really translated into different detection probability values. This detection probability refers to the probability of choosing the cell which contains the effective signal carrier frequency and the effective code properly delay.

C.7.2. Acquisition of the GALILEO E1 signal

In this section, the GALILEO E1 acquisition parameters are presented. These parameters are defined in order to calculate the GALILEO E1 signal acquisition performance case where the receiver is in cold start and no a-priori information is available. Therefore, the entire PRN code and the full range of possible Doppler frequencies have to be explored. Moreover, in this section, the particular case where a part of the data channel is modulated with the CSK signaling technique is also presented. Additionally, the simulator used to calculate the acquisition performance is described.

C.7.2.1. Acquisition parameters of the GALILEO E1 signal

Four different parameters can define the acquisition search of the GALILEO E1 OS signal. These parameters are the total number of bins required to sweep an entire PRN code period and entire range of Doppler frequencies, the maximal allowed false alarm probability (Pfa), the acquisition threshold, and the number of bins inspected each non-coherent integration time.

C.7.2.1.1. Total number of bins

The total number of bins is the number of cells into which the two dimensional search of the signal (time/frequency) is divided in order to cover an entire PRN code chip period and the entire Doppler frequency spectrum. The total number of bins is selected in order to maximize the chances that the signal appears in at least one of the inspected bins.

The received signal is attenuated at the correlator output because the generated code local replica and the generated local carrier have a delay and a frequency fixed by each bin, which is different from the true values. Therefore, the division of the two dimensional search space must be made with a step in frequency and a step in time small enough to ensure that, in the correct bin, the difference between the code local replica and the received PRN code, and the difference between the local carrier and the signal carrier is small enough so that the signal loss of power does not exceed a given maximum attenuation. Note that, the more the signal is attenuated, the more the signal is masked by the noise, and thus, the more it is difficult to identify it.

The model of the received signal at the correlator output is [BASTIDE, 2004].

$$I[k] = \frac{A}{2} \cdot \frac{\sin(\pi \Delta f T p)}{\pi \Delta f T p} \cdot \cos(\varepsilon_\theta[k]) \cdot R_{code,f}(\varepsilon_\tau[k]) + n_I[k] \quad \text{(C-9)}$$

$$Q[k] = \frac{A}{2} \cdot \frac{\sin(\pi \Delta f T p)}{\pi \Delta f T p} \cdot \sin(\varepsilon_\theta[k]) \cdot R_{code,f}(\varepsilon_\tau[k]) + n_Q[k] \quad \text{(C-10)}$$

Where:

- I[k], Q[k]: Channel I and channel Q signal respectively.
- A: Useful signal amplitude at the ADC output
- Δ_f: Frequency offset between the received and the local carriers
- ε_θ : Phase offset between the received and the local carriers
- ε_τ : Phase offset between the received and the local PRN codes
- $R_{code,f}$: Cross-correlation between the filtered received code and the unfiltered local code
- T_p: Coherent integration time
- $n_I[k]$: I channel noise at epoch k
- $n_Q[n]$: Q channel noise at epoch k

Note that this expression is valid for the pilot channel since a data term should be multiplied to the first term of equations (C-9) and (C-10) in order to obtain the data channel expression.

From expressions (C-9) and (C-10), it can be seen that the signal evaluated on a bin has 3 different kinds of attenuations. The first attenuation is the sinc term and is related to the difference of frequency between the received signal carrier and the generated local carrier. The worst attenuation introduced by this term over the signal is found when the received signal falls exactly in the middle of two frequency bins. In this case, the attenuation is calculated by evaluating the term at half the length of the frequency domain step. The expression is given below:

$$A_{freq\_\max} = \left| \frac{\sin\left(\pi \cdot L_f \cdot Tp/2\right)}{\pi \cdot L_f \cdot Tp/2} \right| \tag{C-11}$$

Where:

- L_f: Frequency domain step
- A_{freq_max}: Maximum attenuation caused by the frequency domain step

The second attenuation is the cosine and sine terms which depend on the phase offset, ε_θ, between the received and the local carrier. Nevertheless, this factor disappears since the I and Q channels are powered by two and are added to form the acquisition test criterion. Therefore, the attenuation introduced by the phase is neglected.

The third attenuation is introduced by the cross-correlation between the filtered received code and the unfiltered local one. In this case, the worst attenuation is found following the same principle as the frequency attenuation one. This means that the worst attenuation is calculated by evaluating this term at half the length of the time domain step.

$$A_{time\_\max} = \left| R_{code,f}\left(L_t/2\right) \right| \tag{C-12}$$

Where:

- L_t: Time domain step
- A_{time_max}: Maximum attenuation caused by the time domain step

Once the attenuation expressions have been presented, the total number of bins can be calculated.

The total number of bins is found by multiplying 3 factors: the number of chips forming a PRN code period, the number of different frequency bins which have to be inspected in order to cover the maximum Doppler frequency, and the number of parts into which a chip is divided.

$$Num\_total\_bins = Num\_chip \text{ x } Num\_freqs \text{ x } Num\_parts\_chip \tag{C-13}$$

The number of chips of each PRN code is defined by the signal itself [ESA, 2008]. In this case, each channel PRN code is formed by 4092 chips.

The number of different inspected frequencies can be calculated by dividing the total range of inspected frequencies by the length of the frequency domain step. The frequency range has to cover the entire Doppler frequency spectrum at which the signal can be received. Therefore, the acquisition has to sweep from the nominal frequency minus the maximal Doppler frequency to the nominal frequency plus the maximal Doppler frequency. In this case and for a cold start, the Doppler frequency range is set to +/- 1.5 kHz. The length of the frequency domain step is calculated so that the maximum allowed frequency attenuation is equal to 0.9 dB (A_{freq_max} = 0.8128). In this case, the frequency step is equal to 125Hz.

Finally, the number of parts into which a chip is divided is found by imposing that the maximum allowed time attenuation is equal to 0.75 (A_{time_max} = 0.75). Therefore, in order to find the time step length, the autocorrelation of both channels have to be presented. The expressions of GALILEO E1 OS pilot and data channels autocorrelations before taking into account the front-end filter is shown below [JULIEN et al., 2006].

$$R_{pilot}(\tau) = \frac{10}{11} \cdot R_X(\tau) + \frac{1}{11} \cdot R_X(\tau) - 2 \cdot \frac{\sqrt{10}}{11} R_{X/Y}(\tau) \quad \text{(C-14)}$$

$$R_{pilot}(\tau) = \frac{10}{11} \cdot R_X(\tau) + \frac{1}{11} \cdot R_X(\tau) + 2 \cdot \frac{\sqrt{10}}{11} R_{X/Y}(\tau) \quad \text{(C-15)}$$

Where:

- $R_X(\tau)$: BOC(1, 1) autocorrelation function.
- $R_Y(\tau)$: BOC(6, 1) autocorrelation function.
- $R_{X/Y}(\tau)$: cross-correlation function between the BOC(6, 1) and BOC(1, 1) waveforms.

From expressions (C-14) and (C-15), and taking a front-end filter of 14 MHz of bandwidth double-sided, it is calculated that the length of the time domain step is about $(1/11)\cdot T_{chip}$. And this means that each chip is evaluated on 11 points.

Finally, multiplying the previous 3 factors, the number of total bins is equal to 1080288 bins.

C.7.2.1.2. False alarm probability

The false alarm probability is calculated in order to guarantee a maximum rate of 1 false alarm per code search space sweep. Therefore, the false alarm value is the inverse of the total number of bins.

$$P_{fa} = 1/Num\_total\_bins \quad \text{(C-16)}$$

The false alarm value is equal to 9.2568e-007 for the previous number of total bins value. Finally, the time required to verify that the signal has been falsely acquired (TFAF) is set to 1 second.

C.7.2.1.3. Acquisition threshold

The acquisition threshold is computed assuming that the useful signal is not present and that only thermal noise contributes to the total signal power. This means that the acquisition threshold value has to be calculated in order to guarantee that the presence of thermal noise only is interpreted as the presence of useful signal with a probability equal to the false alarm probability.

Therefore, in order to calculate the acquisition threshold, the noise probability density has to be determined. This expression has already been calculated in [BASTIDE, 2004]; nevertheless, in this dissertation the noises appearing in each inspected bin are considered independent when they are not. However, for the analysis conducted in this dissertation, the [BASTIDE, 2004] expression is considered accurate enough for a preliminary study.

The noise to be compared to the acquisition threshold is the sum of squared Gaussian noises:

$$T_0 = \sum_{i=1}^{M} \left[n_{I,data}^2 + n_{Q,data}^2 + n_{I,pilot}^2 + n_{Q,pilot}^2 \right] \quad \text{(C-17)}$$

Where:

- T_0: Acquisition criteria without the useful signal presence
- M: Number of non-coherent accumulations
- $n_{I,data}$, $n_{Q,data}$, $n_{I,pilot}$, $n_{Q,pilot}$: Gaussian noises from the I and Q data channels and from the I and Q pilot channels respectively with $N(0, \sigma_n^2)$.

Therefore, if the noise samples are considered independent, T_0/σ_n^2 has a chi-square distribution with $4 \cdot M$ degrees of freedom when the data and pilot channels are used to make the acquisition, and T_0/σ_n^2 has a chi-square distribution with $2 \cdot M$ degrees of freedom when only the pilot channels is used to make the acquisition.

Finally, the acquisition threshold (T_h) is calculated by imposing:

$$P_{fa} = P(T_h > T_0) \text{ in presence of noise only} \quad \text{(C-18)}$$

Where:

- P(X): Chi-square distribution with either $2 \cdot M$ or $4 \cdot M$ degrees of freedom.

C.7.2.1.4. Number of inspected bins each non-coherent integration time

The number of bins which are inspected each non-coherent integration time seconds is the main parameter which determines the total amount of time required to acquire the GALILEO E1 signal. More specifically, the receiver needs non-coherent integration time seconds to analyze a bin and to decide whether the useful signal is contained into the bin. Therefore, the acquisition search is limited by the non-coherent integration time and the number of bins which can be inspected at the same time.

On one hand, a long non-coherent integration time is necessary in order to reduce the noise influence over very noised signals, which means that this value is normally fixed. In this simulation, the coherent integration time is fixed to 4ms and the non-coherent integration time is fixed to 100ms, which implies a value of M equal to 25.

On the other hand, the number of bins inspected each non-coherent integration time depends on the number of available correlators and on the number of channels used to inspect the bin. More specifically, 4 correlators are necessary to inspected a bin when the data and pilot channels are used, 1 for the I data channel, 1 for the Q data channel, 1 for the I pilot channel and 1 for the Q pilot channel. And thus, only 2 correlators are used when the pilot channel is employed alone, 1 for the I pilot channel and 1 for the Q pilot channel.

To sum up, in this analysis the number of bins inspected each non-coherent integration time depends on the initial available number of correlators and on the selected acquisition option; if the signal is acquired using the pilot and data channels or if the signal is acquired using only the pilot channel.

C.7.2.2. Acquisition differences with a signal implementing CSK

The introduction of the CSK technique implies that two consecutives symbols carry data channel PRN codes with a different delay. In other words, each symbol carries a different shift of the data channel PRN code compared to the symbols transmitted before and after it. This means that a blind non-coherent integration technique cannot be used over the data channel since all the added coherent integrations can have different code shifted versions. Therefore, if a navigation signal implements the CSK technique over the entire data channel, this channel cannot be used to acquire the signal, since the non-coherent integration is necessary in order to acquire the signal without too high C/N_0 values.

However, the use of the data channel can be considered when the CSK technique is implemented over only a part of the channel. Indeed, it has been shown that the use of pilot and data channels altogether enhance the acquisition performance [BASTIDE, 2004]. However, there are three main problems in using both channels two acquire the signal when the data channel has implemented the CSK technique.

First, the acquisition threshold calculated when using both channels is higher than the acquisition threshold calculated when using only one channel since the number of noise samples for the former case is twice larger than in the latter case. However, the increase of this threshold is compensated by the useful signals of the pilot and data channel. Nevertheless, this cannot be true when the CSK is implemented on the data channel. In this case, when the CSK data channel is used to acquire the signal, only the pilot channel is providing useful signal although the receiver is using both channels. Therefore, since the threshold has been computed assuming the two channels case, the probability of acquiring the signal is low when the analyzed part of the data channel carries the CSK.

Second, since the signal is in process to be acquired, the receiver cannot know if the analyzed part of the data channel carries or does not carry the CSK signaling. In fact, if the receiver had this information, it could choose to change the acquisition threshold in order to adapt itself to the situation. However, this is not normally the case and thus the receiver has to use the threshold computed for the two channels. This is normally the bad option.

Third and last, the probability of false alarm is increased due to the CSK technique. Normally, for a navigation signal without implementing the CSK technique, only the thermal noise term appears into the inspected bins which do not contain the pilot PRN code. However, if the data channel signal implements the CSK signaling technique, in a inspected bin which does not contain the pilot channel PRN code, the correlation between the received circular shifted code version of the data channel PRN code and the generated replica can be evaluated at its peak and thus an additional term is added to the acquisition criteria T_0. Therefore, since the acquisition threshold is calculated assuming the presence of only thermal noise in the acquisition criteria T_0, the inspected bins where, in addition to the thermal noise terms, the correlation between the generated code replica and the received shifted version of the data channel PRN code contributes to the acquisition criteria T_0, have more probabilities of surpassing the threshold and thus have more probabilities to cause a false alarm.

A graphical example is shown in Figure C-3. In this figure, the green and blue areas represent the part of the correlation function between the generated local data channel PRN code replica and the received data channel PRN code which is not zero and thus produces undesired useful signal.

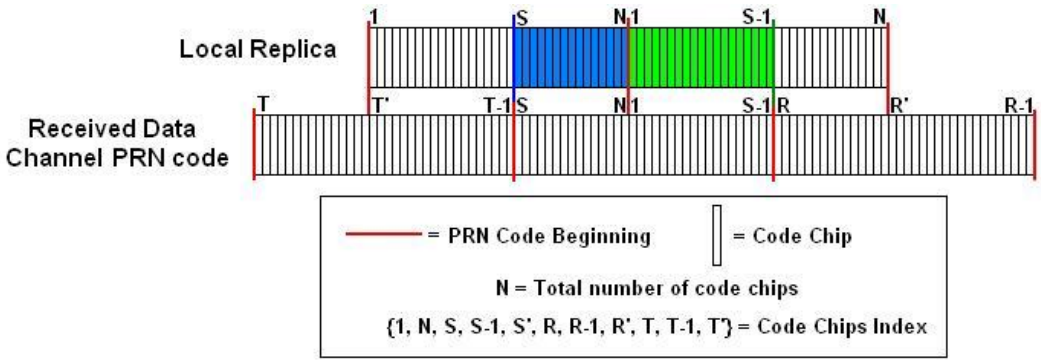

Figure C-3: CSK acquisition undesired correlation between the data channel PRN and the generated local data channel PRN code replica

This study neglected what we call the border effects. In fact, as seen in Figure C-4, the different circular code shifts between two consecutives symbols create a new PRN code. In this case, the original autocorrelations properties of the PRN code are lost since the original data channel PRN code is correlated with a new created code. Therefore, it can no longer be assured that the correlation is about 0 when the codes are delayed by more than one chip.

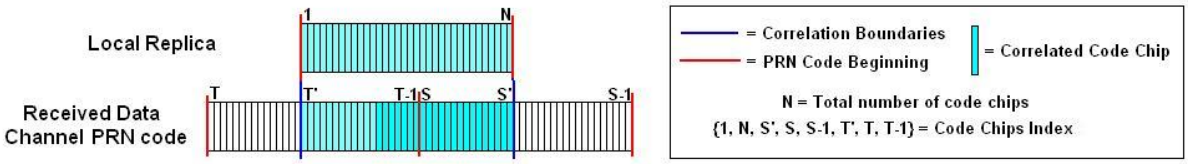

Figure C-4: CSK acquisition border effect

To summarize, due to these different problems, the acquisition process for the GALILEO E1 OS signal proposed by TAS-France must be analyzed in order to examine how the acquisition performance is perturbed when using the pilot and data channel to acquire the signal.

C.7.2.3. Optimal CSK symbol shift source mapping

In this section, the optimal CSK symbol shift mapping source is presented. The optimal CSK symbol shift mapping source is the optimal association between code circular shifts and CSK symbols.

In order to make that choice, this analysis searches the circular codes shifts which provide the lowest level of false alarm probability; or, in other words, this analysis searches the lowest contribution of the correlation between a shifted data channel PRN code and the generated data channel PRN code replica to the acquisition criteria T_0. The assumptions taken into account are that the border effects are negligible and that the partial data channel PRN code correlations are about 0. Therefore, inspecting Figure C-3, two cases can be observed.

The first case is that the same two CSK symbols are transmitted consecutively, which means that the coherent integration of one PRN code period of the local replica achieves the maximal autocorrelation value. In fact, in this case, the transmission of two consecutive identical CSK symbols means that the original unshifted PRN code is found along the transmission of these two symbols. And note that, regardless of the selected circular shift, the correlation always reaches its maximum value which means that this case does not condition the CSK symbol shift mapping.

The second case is found when two consecutives transmitted CSK symbols are not equal. In this situation, it can be observed from Figure C-3 that two consecutives coherent integrations appear at the same time, where each one correlates a part of the useful signal depending on the transmitted code circular shift. Moreover, note that since no identical CSK symbols are transmitted consecutively, each individual coherent integration is only affected by one received CSK symbol. Therefore, the circular shift selection is done in order to minimize the useful signal contribution made by two consecutives coherent integrations. The function defining the contribution is shown below:

$$f(n) = \left(\frac{N - n + 1}{N} \right)^2 + \left(\frac{n}{N} \right)^2 \quad \text{(C-19)}$$

Where:

- N: Total number of chips forming the PRN code
- n: Circular code shift

Searching the minimum of the previous function, the optimal circular code shift can be found. And this shift is equal to (N+1)/2, or equivalently, the code is shifted circularly half its length.

Finally, since the number of circular shifts has to be equal to the number of CSK symbols, the optimal shift mapping is obtained by associating each CSK symbol to the nearest available shift to (N+1)/2.

C.7.2.4. Description of the simulator used to calculate the GALILEO E1 signal acquisition performance

The simulator implemented in order to analyze the GALILEO E1 OS signal performance is described next. The main principle of the simulator is explained first.

Each MT_P seconds, where T_p is the coherent integration time and M is the number of non-coherent accumulations, the simulator analyzes a number of bins determined by the number of correlators and by the acquisition configuration. Each bin is analyzed using the part of the signal received during its inspection. This means that the received signal used to inspect two bins which are analyzed in different MT_P periods of time is not the same. In fact, during MT_P seconds, the receiver is blocked analyzing a set of bins and cannot make any other action. But, once the inspection of the set of bins is finished, the receiver begins to inspect a new set of bins using the signal being received at the new epoch. This process is repeated until one inspected bin obtains a power value which is higher than the acquisition threshold. And, this means that the acquisition process can sweep several times the entire bin time/frequency domain before acquiring the signal or may not sweep the entire domain even once.

Once the fundamental principle is explained, a more detailed description of the simulator is given. First, the acquisition threshold is calculated from the assumed N_0 at the receiver antenna output and the number of channels used to acquire the signal, either the pilot and the data channel or only the pilot. This acquisition threshold is calculated by modeling the noise with a central chi-square function. Therefore, the value obtained has to be multiplied by the noise variance at the correlator output, σ_n^2, in order to obtain the real threshold value. For this simulation, the noise variance at the correlator output is calculated without taking into account the front-end filter.

Second, once the acquisition threshold is determined, the generation of the acquisition criteria is simulated. Normally, this process is simulated as a non-central chi-square function. Nevertheless, since in this simulation the noise correlation between consecutives bins is taken into account, another solution has to be used. The solution chosen is first to generate all the correlated noise samples of all the bins inspected during a non-coherent integration time. Then, we generate the useful signal correlator outputs associated to each bin. Note that since the received signal used to inspect the bins analyzed during different MT_P periods of time is not the same, the noise of a bin can only be correlated with noises of bins analyzed during the same MT_P period.

Third and last, the acquisition criterion is compared to the acquisition threshold in order to determine if the signal is detected. Therefore, depending on the acquisition criterion, on the acquisition threshold and on the true portion of the peak in the search space, the false alarm and detection counters are updated. Moreover, each time that a group of bins has been inspected, the acquisition time is increased by MT_P. This time is also increased by TFAF seconds each time that the signal is acquired when it should not. Finally, if the acquisition is made, the simulation is ended.

Annex D. Advanced description of the GNSS signals

In this annex, a more detailed description of the GNSS signals analyzed in this research work is presented. Besides, a navigation message structure for the GALILEO E1 signal proposed by THALES ALENIA SPACE France is described and its demodulation performance is commented.

D.1. GPS L1 C/A signal structure

The GPS L1 C/A signal is contained within one band of 20.46 MHz centered on L1. The nominal carrier frequency (f_0) for L1 is 1575.42 MHz.

The spreading code of the GPS L1 C/A signal is a pseudo-random noise sequence used to spread the signal spectrum. The PRN code is different for each satellite and it allows the GPS L1 C/A receiver to differentiate among the different satellites transmitting at the same frequency carrier L1. Each L1 C/A PRN code has a duration of 1 ms at a chipping rate of 1023 kbps, meaning that each code has a length of 1023 chips.

Moreover, each PRN code waveform is labeled as $G_i(t)$ and it is the physic materialization of a 1023 bit Gold Code. Therefore, each PRN code is generated from the modulo-2 sum of two 1023 bit m-sequences, G1 and $G2_i$. $G2_i$ is constructed by delaying the nominal sequence G2 by an integer number of chips. Finally, G1 and G2 m-sequences are generated by 10 stage shift registers.

D.2. GPS L2C signal structure

The GPS L2C signal is contained within one band of 20.46 MHz centered on L2. The nominal carrier frequency (f_0) for L2 is 1227.6 MHz.

Following the same logic as the GPS L1 C/A signal conception, the PRN codes of the GPS L2C signal, CL and CM, are different for each transmitting satellite and they are different from each other. This property allows the differentiation among satellites in addition to allow the separation between the part of the signal carrying the L2 CL code and the part of the signal carrying the L2 CM code. Therefore, due to the differentiation among satellites and the chip-by-chip time multiplex, the GPS L2C signal can be divided into two channels: the pilot channel, the part of the signal containing the L2 CL code, and the data channel, the part of the signal containing the L2 CM code plus the navigation data message.

This separation into two channels allows a better performance in terms of acquisition and tracking process since the accumulation time on the pilot channel is no longer restricted by the duration of a data symbol [JULIEN, 2005]. The power distribution between the two channels is 50% for the pilot channel and 50% for the data channel of the total amount of power.

The L2 CL code has a duration of 1.5 seconds at a chipping rate of 511.5 Kbps, meaning a length of 767250 chips. This code is generated with a modular-type shift register generator with a determined code generator polynomial [ARINC, 2004] and it is reinitialized after 767250 chips. Each different code $CL_i(t)$ is created from a different configuration of the register's initial state while having the same code generator polynomial.

The L2 CM code has a duration of 20 milliseconds at a chipping rate of 511.5 Kbps, meaning a length of 10230 chips. This code is generated in the same way as the CL code, using the same modular-type shift register generator and the same code generator polynomial, but it is reinitialized only after the generation of 10230 chips. Each different code $CM_i(t)$ is created from a different configuration of the register's initial state, and this configuration is different from any $CL_i(t)$ code.

D.3. GPS L5 signal

The GPS L5 signal is contained within a band of 24 MHz centered on L5. The nominal carrier frequency (f_0) for L5 is 1176.45 MHz.

Following the same logic as the previous GPS signal conception, the PRN codes of the GPS L5 signal, I5-code and Q5-code, are different for each transmitting satellite and they are different from each other. This property allows the differentiation among satellites in addition to the separation between the part of the signal carrying the I5-code and the part of the signal carrying the Q5-code. Therefore, due to the differentiation among satellites and the in phase-quadrature modulation, the GPS L5 signal can be divided into two channels: the pilot channel, part of the signal containing the Q5-code, and the data channel, part of the signal containing the I5-code plus the navigation data message.

This separation into two channels allows a better performance in terms of acquisition and tracking process since the accumulation time on the pilot channel is no longer restricted by the duration of a data symbol [JULIEN, 2005]. The power distribution between the two channels is 50% for the pilot channel and 50% for the data channel of the total amount of power.

The I5-code has a duration of 1 millisecond at a chipping rate of 10.23 Mbps, meaning a length of 10230 chips. This code is generated by a 2-modulo sum of two extended patterns, XA and XBI_i, also clocked at 10.23 Mbps. XA is a 8190 chip length code which is 1-bit short cycled before its natural period end, and XBI_i is a 8191 chip length code which is not short cycled. Each XBI_i code has a different initial condition but all of them are generated by the same generator polynomial.

The Q5-code has a duration of 1 millisecond at a chipping rate of 10.23 Mbps, meaning a length of 10230 chips. This code is generated by a 2-modulo sum of two extended patterns, XA and XBQ_i, also clocked at 10.23 Mbps. XA is a 8190 chip length code which is 1-bit short cycled before its natural period end, and XQI_i is an 8191 chip length code which is not short cycled. Each XQI_i code has a different initial condition but all of them are generated by the same generator polynomial, which is also the XBI_i generator polynomial. The initial conditions of XBI_i and XQI_i are different.

The synchronization sequence of the data channel is a 10-bit Neuman-Hofman code clocked at rate of 1 kHz, and thus has a duration equal to 10 ms, equal to 1 navigation data symbol. This code is synchronized with the data information symbols.

The synchronization sequence of the pilot channel is a 20-bit Neuman-Hofman code clocked at rate of 1 kHz, and thus has a duration equal to 20ms, equal to 2 navigation data symbols. This code is synchronized with one out of two data information symbols.

D.4. GPS L1C signal

The GPS L1C signal is contained within a band of 30.69 MHz centered on L1. The nominal carrier frequency (f_0) for L1 is 1575.42 MHz.

Following the same logic of the previous GPS signals conception, the PRN codes of the GPS L1 signal, $L1C_P$ and $L1C_D$, are different for each transmitting satellite and they are different from each other. This property allows the differentiation among satellites in addition to the separation between the part of the signal carrying the L1 C_P code and the part of the signal carrying the L1 C_D code. Therefore, due to the differentiation among satellites and the differentiation among L1 codes, the GPS L1C signal can be divided into two channels: the pilot channel, part of the signal containing the C_P code, and the data channel, part of the signal containing the C_D code plus the navigation data message.

This separation into two channels allows a better performance in terms of acquisition and tracking process since the accumulation time on the pilot channel is no longer restricted by the duration of a data symbol [JULIEN, 2005]. The power distribution between the two channels is 75% for the pilot channel and 25% for the data channel of the total amount of power.

The $L1C_{Pi}(t)$ and the $L1C_{Di}(t)$ codes have a duration of 10ms at a chipping rate of 1.023 Mbps, meaning a length of 10230 chips. Both codes are the result of inserting into a unique 10233-length sequence a fixed 7-bit length sequence. The insertion point of the 7-bit length sequence varies for each satellite and is not the same for the $L1C_{Pi}(t)$ codes than for the $L1C_{Di}(t)$ codes. This unique 10233-length sequence is a Weil code generated from the 2-modulo sum of two 10233 Legendre sequences. These two Legendre sequences are the same but one is shifted in relation to the other. This shift is different between the $L1C_{Pi}(t)$ codes and the $L1C_{Di}(t)$ codes, and it also varies among satellites.

The overlay codes $L1C_{Oi}(t)$ are constructed using the Linear Feedback Shift Register (LFSR) method. These codes are 2047 bits long sequences truncated to 1800-bits long sequences at a rate of 100 bit/sec, and thus their duration is 18s. These codes are constructed using 11-stage LFSR generators. More precisely, each code is generated by 2-modulo adding two sequences S1 and S2 with different polynomial generators for each satellite. Nevertheless, the first 64 sequences are generated by only using S1.

Finally, the interleaver applied to the subframe 2 and subframe 3 of the GPS L1C navigation message is presented.

D.4.1. Interleaver and Desinterleaver

In this section the interleaving block and the way it works are described. The function of the desinterleaver which is supposed to be the inverse process of the interleaver is not detailed here.

The block interleaver is conceptually described using a two-dimensional array of 38 rows and 46 columns (the multiplication result is 1748: the exact number of bits of subframe 2 plus subframe 3), as depicted in Figure D-1 [ARINC, 2006]. The LDPC encoded subframe 2 bits are written first (MSB first) into the interleaver from left to right starting at row 1. After row 1 is filled, row 2 is filled from left to right and this process continues until the 1748^{th} bit (LSB of LDPC encoded subframe 3) is written into the rightmost cell of the last (38^{th}) row. Once all 1748 bits are written into the array, the symbols are sequentially read out of the array, for broadcast to user, from top to bottom starting at column 1. After reading out the last (38^{th})

symbol in column 1, column 2 bits are read out from top to bottom and this process continues until the last symbol (38th) of the last column (46th) is read out.

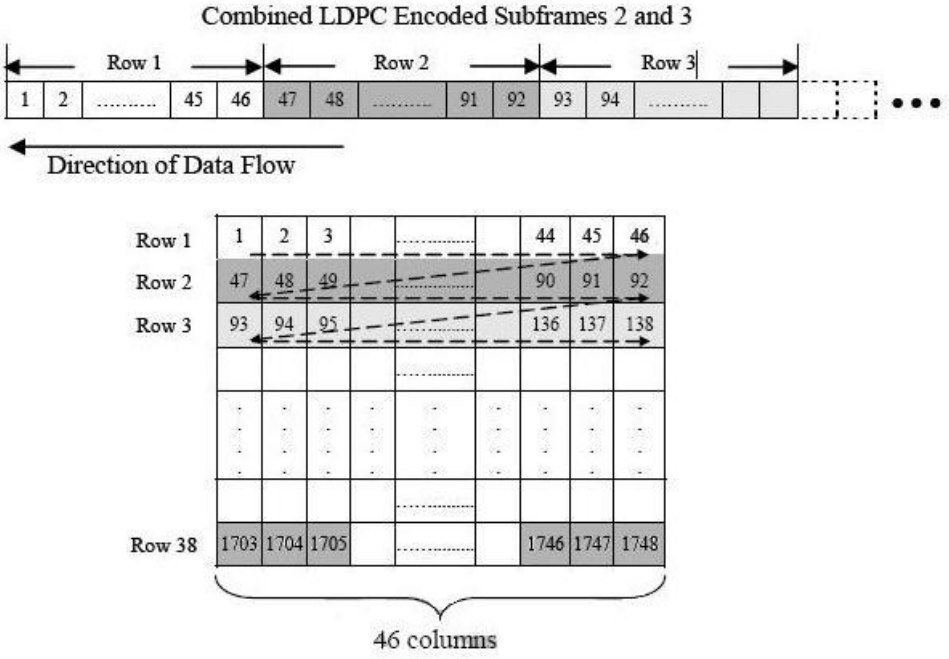

Figure D-1: Interleaver GPS L1C Navigation Message [ARINC, 2006]

Therefore, the desinterleaver process is to first fill the array from column 1 to column 46 (last one) beginning at the top and finishing at the bottom of the columns, and, afterwards, to read the array from row 1 to row 38 from left to right of the rows. Figure D-2 shows a scheme of the desinterleaver process [ARINC, 2006].

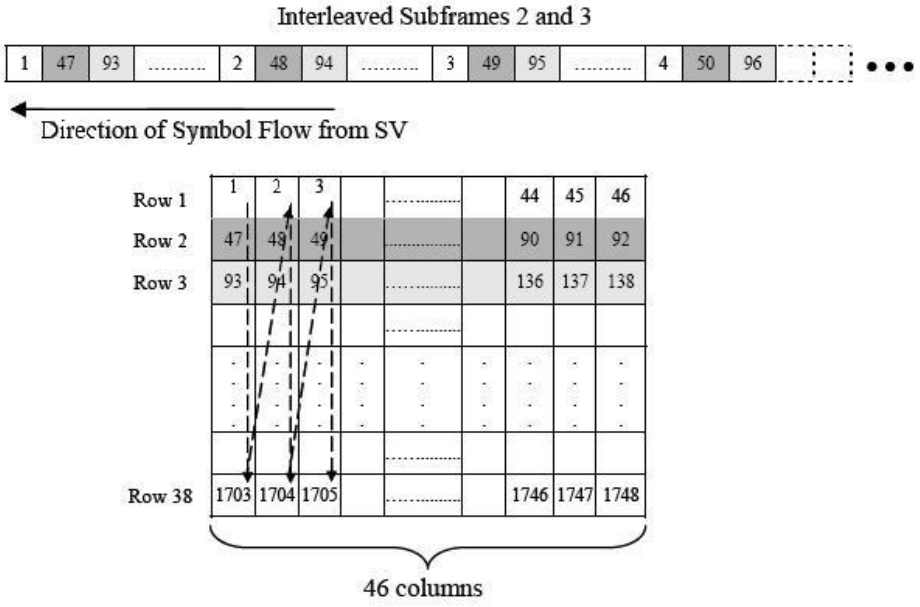

Figure D-2: Desinterleaver GPS L1C navigation message [ARINC, 2006]

D.5. GALILEO E1 signal

A possible GALILEO E1 OS signal modulator scheme is shown below.

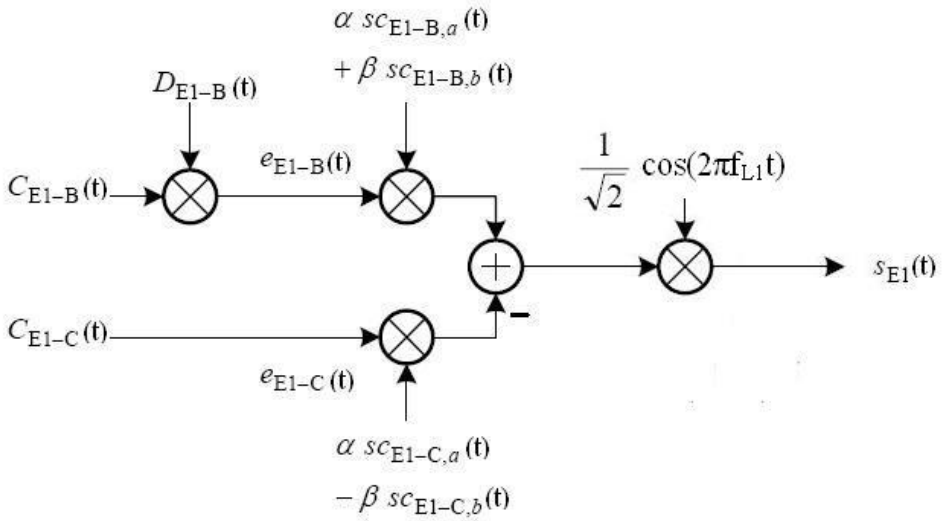

Figure D-3: GALILEO E1 Modulation Scheme

Where:

- $e_{E1\text{-}B}$ is the I/NAV navigation data waveform $D_{E1\text{-}B}$ multiplied by the ranging code waveform $C_{E1\text{-}B}$, then modulating the sub-carriers $sc_{E1\text{-}B,a}$ and $sc_{E1\text{-}B,b}$
- $e_{E1\text{-}C}$ (pilot component) is the ranging code waveform $C_{E1\text{-}C}$ including its secondary code, then modulating the sub-carriers $sc_{E1\text{-}C,a}$ and $sc_{E1\text{-}C,b}$

The mathematical models of $c_{E1\text{-}B}(t)$, $c_{E1\text{-}C}(t)$ and $d_{E1\text{-}B}(t)$ are:

$$c_{E1-B}(t) = \sum_{k=-\infty}^{+\infty} c_{E1-B}[k] \cdot rect_{Tc}[t - k \cdot Tc] \quad \text{(D-1)}$$

$$c_{E1-C}(t) = \sum_{k=-\infty}^{+\infty} c_{E1-C}[k] \cdot rect_{Tc}[t - k \cdot Tc] \quad \text{(D-2)}$$

$$d_{E1-B}(t) = \sum_{k=-\infty}^{+\infty} d_{E1-B}[k] \cdot rect_{Td}[t - k \cdot Td] \quad \text{(D-3)}$$

Where:

- T_c: Duration of a PRN code chip
- T_c: Duration of a data symbol
- $rect_T(t)$: Rectangular impulse of T width centered at t = 0 The sub-carrier rates are:

Component (Parameter Y)	Sub-carrier Type	Sub-carrier Rate (MHz)		Ranging Code Chip-Rate $R_{C,E1\text{-}Y}$ (Mcps)
		$R_{S,E1\text{-}Y,a}$	$R_{S,E1\text{-}Y,b}$	
B	CBOC , in-phase	1.023	6.138	1.023
C	CBOC, anti-phase	1.023	6.138	1.023

Table D-1 : GALIEO E1-B CBOC Chip and Sub-carriers Rate

The final mathematical model of $s_{E1}(t)$ is:

$$s_{E1}(t) = \frac{1}{\sqrt{2}} \cdot m(t) \cdot \cos\left(2\pi f_{L1} t\right) \tag{D-4}$$

Where:

$$\begin{aligned} m(t) = {} & e_{E1-B}(t) \cdot \left(\alpha \cdot sc_{E1-B,a}(t) + \beta \cdot sc_{E1-B,b}(t)\right) \\ & - e_{E1-C}(t) \cdot \left(\alpha \cdot sc_{E1-C,a}(t) - \beta \cdot sc_{E1-C,b}(t)\right) \end{aligned} \tag{D-5}$$

$$sc_X(t) = \mathrm{sgn}\left[\sin\left(2\pi R_{s,X}(t)\right)\right] \tag{D-6}$$

$$\alpha = \sqrt{\frac{10}{11}} \quad \beta = \sqrt{\frac{1}{11}} \tag{D-7}$$

The GALILEO E1 signal is contained within a band of 24.552 MHz centered on L1. The nominal carrier frequency (f_0) for L1 is 1575.42 MHz.

The PRN codes of the GALILEO E1 signal, $C_{E1\text{-}C}$ and $C_{E1\text{-}B}$, are different for each transmitting satellite as well as they are different from each other. This property allows the differentiation among satellites in addition to the separation between the part of the signal carrying the $C_{E1\text{-}C}$ code and the part of the signal carrying the $C_{E1\text{-}B}$ code. Therefore, due to the differentiation among satellites and the differentiation among GALILEO E1 codes, the GALILEO E1 signal can be divided into two channels: the pilot channel, part of the signal containing the $C_{E1\text{-}C}$ code, and the data channel, part of the signal containing the $C_{E1\text{-}B}$ code plus the navigation data message.

This separation into two channels allows a better performance in terms of acquisition and tracking process since the accumulation time on the pilot channel is no longer restricted by the duration of a data symbol [JULIEN, 2005]. The power distribution between the two channels is 50% for the pilot channel and 50% for the data channel of the total amount of power.

The PRN codes $C_{E1\text{-}C}$ and $C_{E1\text{-}B}$ are memory codes, or in other words, optimized pseudo-noise sequences which need to be stored in memory since they cannot be reproduced from a LFSR for example. The $C_{E1\text{-}B}$ is transmitted at a rate of 1.023 Mchip/s and has a length of 4092 chips meaning a duration of 4ms. The $C_{E1\text{-}C}$ is a primary memory code further modified by a secondary code. This secondary code is also a memory code, it is called $CS25_1$ and it has a length of 25 chips. This means that the primary code is repeated 25 times and its sign is controlled by the chips of the secondary code, see Figure D-4. Therefore, since the length of the primary code is 4092 chips transmitted at a rate of 1.023 Mchips/sec, the primary code has a duration of 4ms whereas the secondary one lasts 100ms. Figure D-4 illustrates the $C_{E1\text{-}C}$ code generation [ESA, 2008].

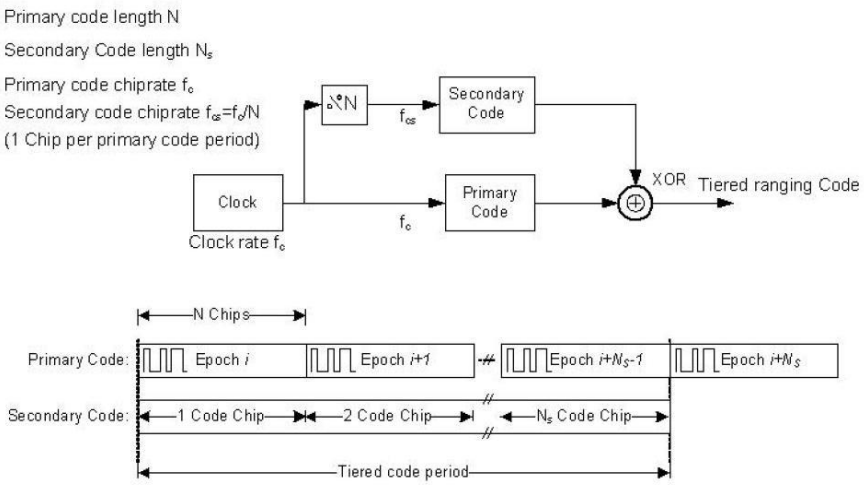

Figure D-4: Tiered Codes Generation [ESA, 2008]

D.5.1. Navigation message

In this annex, the different information units of the GALILEO E1 OS signal are described. The information units are the part of a page, a page, subframe and frame. Moreover, the interleaver used to interleave a part of a page is also described.

D.5.1.1. Part of a Page Structure

Each type of page is divided into two parts; the first part is denoted 'even' and the second one is denoted 'odd'. A part of a page, even or odd, carries 120 information bits which are encoded with a convolutional code (171, 133, r = ½) resulting into 240 coded bits. Moreover, 10 additional synchronization bits are inserted before the coded bits in order to allow the synchro-part-of-a-page process at reception. Figure D-5 shows the construction of a part of a page [ESA, 2008].

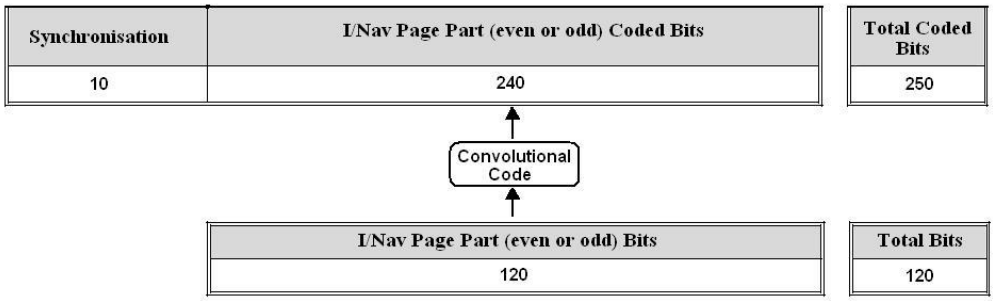

Figure D-5: Even or Odd Part of a Page Structure Galileo E1 OS [ESA, 2008]

The synchronization symbols are equal to 0101100000.

Since the total size of an even or odd part of a page is 250 symbols after applying the CBOC modulation and the GALILEO E1 signal symbol transmission rate is equal to 250 symb/s, the part of a page duration is 1 second.

D.5.1.2. Interleaver and desinterleaver

Each part of a page, even or odd, is processed with an interleaver block before being transmitted to the channel. The interleaver block is simply a matrix of 30 columns and 8 rows, which is equivalent to 240 slots. That is exactly the same number as the total amount of bits in a part of a page before inserting the synchronization symbols.

The interleaver first fills the array with the 240 coded bits starting at column 1 from its top to its bottom and continuing with column 2 until column 30. Afterwards, the bits are read from the array and transmitted to the channel starting at row 1 from its left to its right, and continuing with row 2 until row 8.

The process for the desinterleaver is the opposite, first the rows are filled and after the information is read from the columns.

D.5.1.3. Page Structure

Each page is composed of two parts of a page, the even part and the odd one. Moreover, each page is fully transmitted on the GALILEO E1 signal although their even and odd parts can be, at the same time, transmitted on the GALILEO E5b-I signal. Therefore, since the duration of a part of a page is 1 second, the duration of a page is always 2 seconds.

There are two different types of pages. These pages are called Nominal Pages and Alert Pages. The nominal pages are transmitted sequentially in time on the GALILEO E1 signal; their even and odd parts are transmitted one after the other, and they are not repeated at the same time on the GALILEO E5b-I signal. The alert pages are also transmitted sequentially in time on the GALILEO E1 signal, but during the epoch that this signal transmits the even part, the GALILEO E5b-I signal transmits the odd part; the opposite is also true.

The alert page structure is not available in the GALILEO ICD draft 1 [ESA, 2008]; therefore, the only page structure studied and displayed along this dissertation is the nominal one. This page carries the fundamental information necessary to obtain the user final position. Nevertheless, not all the information of a nominal page serves to carry the Open Service. In fact, only 128 out of the total 240 information bits carry the OS content. The other bits have different functions which are listed next.

- Part of the other services, SoL or CS (noted as reserved)
- Page and word identifiers
- SAR data: System information generated by the satellites destined to the ground segment.
- A parity check code: CRC-24Q which validates all the page information transmitted before it.
- Tail bits: Fixes the final state of the convolutional code applied on the page. A more detailed explanation is provided in annex E.3.2.3.

The structure of a nominal page is described below [ESA, 2008].

Page Part 1

Even/odd=0	Page Type	Data j (1/2)	Tail	Total (bits)
1	1	112	6	120

Page Part 2

Even/odd=1	Page Type	Data j (2/2)	Reserved 1	SAR	Spare	CRC_j	Reserved 2	Tail	Total (bits)
1	1	16	40	22	2	24	8	6	120

Figure D-6: Page Structure Galileo E1 OS [ESA, 2008]

Note that Figure D-6 shows the page structure of the information bits of a part of a page. Therefore, each part of a page has first to be encoded with the convolutional code, second the coded bits have to be interleaved and third the synchronisation symbols have to be added. Only then, the parts of a page are transmitted to the channel.

D.5.1.4. Subframe Structure

A subframe is composed of 15 pages. The information content type of a page depends on its position inside the subframe. This means that pages situated at the same position inside the subframe but transmitted inside different subframes have the same type of information content. Figure D-7 shows the GALILEO E1 OS signal subframe scheme [ESA, 2008].

T_0 (GST_0 sync.) (s)	E1-B Content	E1-B Page	E1B Sub frame ID
0	Spare Page (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N-1
1	Page 2 (1/2)	Even	N
2	Page 2 (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N
3	Page 4 (1/2)	Even	N
4	Page 4 (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N
5	Page 6 (1/2)	Even	N
6	Page 6 (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N
7	Page 7 or 9 (1/2)	Even	N
8	Page 7 or 9 (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N
9	Page 8 or 10 (1/2)	Even	N
10	Page 8 or 10 (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N
11	Reserved (1/2)	Even	N
12	Res. (2/2) | SAR | Spare | CRC | Res.	Odd	N
13	Reserved (1/2)	Even	N
14	Reserved (2/2) | SAR | Spare | CRC | Res.	Odd	N

T_0 (GST_0 sync.) (s)	E1-B Content	E1-B Page	E1B Sub frame ID
15	Reserved (1/2)	Even	N
16	Reserved (2/2) | SAR | Spare | CRC | Res.	Odd	N
17	Reserved (1/2)	Even	N
18	Reserved (2/2) | SAR | Spare | CRC | Res.	Odd	N
19	Reserved (1/2)	Even	N
20	Reserved (2/2) | SAR | Spare | CRC | Res.	Odd	N
21	Page 1 (1/2)	Even	N
22	Page 1 (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N
23	Page 3 (1/2)	Even	N
24	Page 3 (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N
25	Page 5 (1/2)	Even	N
26	Page 5 (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N
27	Spare Page (1/2)	Even	N
28	Spare Page (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N
29	Spare Page (1/2)	Even	N
30	Spare Page (2/2) | Res. | SAR | Spare | CRC | Res.	Odd	N

Figure D-7: Subframe Structure Galileo E1 OS [ESA, 2008]

Pages 1 to 4 contain the ephemeris and clock correction data. Page 5 contains the BGD, the signal health and data validity status, the ionospheric correction and the GST. The content of Page 6 is the GST-UTC conversion parameters, the signal health and the data validity status. Finally Pages 7 to 10 contain the almanacs.

The duration of a subframe is 30 seconds since it is carries 15 pages or 30 parts of a page.

D.5.1.5. Frame Structure

Pages 1 to 4 and, 5 and 6, have a content which is constant from one subframe to the next one until one new set of ephemeris and clock correction data is broadcasted. The time between

consecutive broadcasted ephemeris set is not specified, which means that this time can have any value.

Pages 7 to 10 carry the almanac information of all the satellites. One subframe is able to transport all the almanac data of a given satellite, and the satellite of which the almanac data is broadcasted by subframe *n* is different from the satellite of which the almanac data is broadcasted by subframe *n+1*. Hereby, a frame is defined by the group of subframes which broadcast all the available satellites almanac data. In GALILEO E1 OS navigation message a frame is composed of 24 subframes. A GALILEO E1 OS frame structure scheme is shown below [ESA, 2008].

T_0 [s]	Sub-frame ID	E1B
0	1	Page 7: Almanac SV 19 (1/2)
		Page 8: Almanac SV 19 (2/2) + almanac SV 20 (1/2)
30	2	Page 9: Almanac SV 20 (2/2) + almanac SV 21 (1/2)
		Page 10: Almanac SV 21 (2/2)
60	3	Page 7: Almanac SV 22 (1/2)
		Page 8: Almanac SV 22 (2/2) + almanac SV 23 (1/2)
90	4	Page 9: Almanac SV 23 (2/2) + almanac SV 24 (1/2)
		Page 10: Almanac SV 24 (2/2)
120	5	Page 7: Almanac SV 25 (1/2)
		Page 8: Almanac SV 25 (2/2) + almanac SV 26 (1/2)
150	6	Page 9: Almanac SV 26 (2/2) + almanac SV 27 (1/2)
		Page 10: Almanac SV 27 (2/2)
180	7	Page 7: Almanac SV 28 (1/2)
		Page 8: Almanac SV 28 (2/2) + almanac SV 29 (1/2)
210	8	Page 9: Almanac SV 29 (2/2) + almanac SV 30 (1/2)
		Page 10: Almanac SV 30 (2/2)
240	9	Page 7: Almanac SV 31 (1/2)
		Page 8: Almanac SV 31 (2/2) + almanac SV 32 (1/2)
270	10	Page 9: Almanac SV 32 (2/2) + almanac SV 33 (1/2)
		Page 10: Almanac SV 33 (2/2)
300	11	Page 7: Almanac SV 34 (1/2)
		Page 8: Almanac SV 34 (2/2) + almanac SV 35 (1/2)
330	12	Page 9: Almanac SV 35 (2/2) + almanac SV 36 (1/2)
		Page 10: Almanac SV 36 (2/2)

T_0 [s]	Sub-frame ID	E1B
360	13	Page 7: Almanac SV 1 (1/2)
		Page 8: Almanac SV 1 (2/2) + almanac SV 2 (1/2)
390	14	Page 9: Almanac SV 2 (2/2) + almanac SV 3 (1/2)
		Page 10: Almanac SV 3 (2/2)
420	15	Page 7: Almanac SV 4 (1/2)
		Page 8: Almanac SV 4 (2/2) + almanac SV 5 (1/2)
450	16	Page 9: Almanac SV 5 (2/2) + almanac SV 6 (1/2)
		Page 10: Almanac SV 6 (2/2)
480	17	Page 7: Almanac SV 7 (1/2)
		Page 8: Almanac SV 7 (2/2) + almanac SV 8 (1/2)
510	18	Page 9: Almanac SV 8 (2/2) + almanac SV 9 (1/2)
		Page 10: Almanac SV 9 (2/2)
540	19	Page 7: Almanac SV 10 (1/2)
		Page 8: Almanac SV 10 (2/2) + almanac SV 11 (1/2)
570	20	Page 9: Almanac SV 11 (2/2) + almanac SV 12 (1/2)
		Page 10: Almanac SV 12 (2/2)
600	21	Page 7: Almanac SV 13 (1/2)
		Page 8: Almanac SV 13 (2/2) + almanac SV 14 (1/2)
630	22	Page 9: Almanac SV 14 (2/2) + almanac SV 15 (1/2)
		Page 10: Almanac SV 15 (2/2)
660	23	Page 7: Almanac SV 16 (1/2)
		Page 8: Almanac SV 16 (2/2) + almanac SV 17 (1/2)
690	24	Page 9: Almanac SV 17 (2/2) + almanac SV 18 (1/2)
		Page 10: Almanac SV 18 (2/2)

Figure D-8: Galileo E1 OS Frame Structure [ESA, 2008]

In this scheme, it can be seen that only one couple of pages either, 7 and 8, or, 9 and 10, are transmitted for each subframe. This is due to the subframe structure which only reserves 2 pages for the almanac transmission.

The duration of a frame is 720 second that is to say 12 minutes.

D.6. New proposed GALILEO E1 navigation message structure

The new navigation message structure proposed by THALES ALENIA SPACE is presented in this section. The objective behind this navigation message proposal for a future GALILEO E1 signal is to improve the demodulation performance obtained by the current GALILEO E1 OS signal since this demodulation performance is proven to not be satisfying enough in 0. Moreover, the new proposed navigation message structure also seeks to increase the data symbol transmission rate in order to allow the providing of new services.

The new navigation message structure proposed by TAS-France recovers elements from the GPS L1C navigation message structure and also implements the CSK technique. This combination results into a new navigation message with a higher information transmission rate and a better ephemeris and almanac demodulation performance than the information transmission rate and demodulation performance obtained for the current defined GALILEO E1 navigation message.

More specifically, the GALILEO E1 original words and information structure are modified. The Viterbi pages of 250 symbols and their convolutional code no longer exist. They have been substituted by the GPS L1C subframe structures and its implemented channel codes. Nevertheless, in order to conserve the frame length of 30 seconds, the GALILEO E1 signal ephemerides and almanacs fields are located differently inside the frame from the location used for the GPS L1C fields.

Moreover, in addition to modifying the navigation message structure, this proposal also suggests changing the data channel PRN code structure in order to implement the CSK technique. Remember that the CSK technique is responsible of increasing the signal information transmission rate. The CSK is fully described in subsection 7.1. However, although this proposal changes the data channel PRN code structure, the power distribution between the data and pilot channels, the symbol transmission rate of the non-CSK symbols and the CBOC modulation of the pilot and data channels are not modified. More specifically, as it can be seen from Figure D-9, the total amount of 250 symbols is transmitted each second, which means that the original symbol transmission rate is conserved as it has been said previously.

A scheme of the new proposed navigation message structure called ERIS is shown below. More specifically, the frame of 30 seconds is presented.

Time (s)	Subframe (number of bits)						Total Number of bits
	1/3	2	4	5			
1	52	8	142	16	16	16	250
2	0	60	142	16	16	16	250
3	8	52	142	16	16	16	250
4	20	40	142	16	16	16	250
5	20	40	142	16	16	16	250
6	20	40	142	16	16	16	250
7	20	40	142	16	16	16	250
8	20	40	142	16	16	16	250
9	20	40	142	16	16	16	250
10	20	40	142	16	16	16	250
11	20	40	142	16	16	16	250
12	20	40	142	16	16	16	250
13	20	40	142	16	16	16	250
14	20	40	142	16	16	16	250
15	20	40	142	16	16	16	250
16	20	40	142	16	16	16	250
17	20	40	142	16	16	16	250
18	20	40	142	16	16	16	250
19	20	40	142	16	16	16	250
20	20	40	142	16	16	16	250
21	20	40	142	16	16	16	250
22	20	40	142	16	16	16	250
23	20	40	142	16	16	16	250
24	20	40	142	16	16	16	250
25	20	40	142	16	16	16	250
26	20	40	142	16	16	16	250
27	20	40	142	16	16	16	250
28	20	40	142	16	16	16	250
29	20	40	142	16	16	16	250
30	20	40	142	16	16	16	250
	548	1200					

Figure D-9: THALES ALENIA SPACE proposed ERIS frame of 30 seconds

The different subframes content is described next.

- Sub-frame 1 is transmitted during the 1st second. Carries the Time of Interval (TOI), a counter of 9 information bits which after BCH encoding results into 52 coded bits. Each coded bit is mapped with a BPSK symbol. This subframe is represented with yellow color.
- Sub-frame 2 is transmitted during the 30 seconds. Carries the Ephemeris and clock corrections error data parameters. This information is represented by 600 information bits which after LDPC encoding results into 1200 coded bits. Each coded bit is mapped with a BPSK symbol. This subframe is represented with green color.
- Sub-frame 3 is transmitted during the last 28 seconds. Carries different navigation parameters such as the almanac parameters, UTC parameters, ionospheric parameters, etc. This information is represented by 274 information bits which after LDPC encoding results into 548 coded bits. Each coded bit is mapped with a BPSK symbol. This subframe is represented with yellow-orange color.
- Sub-frame 4 is transmitted during 30 seconds. The information carried by this subframe is still to be determined. Nevertheless, it has to carry the data necessary to provide additional services such as Safety of Life, ERIS, SAR Return, ALIVE, etc. Moreover, the CSK signaling technique is being considered to be implemented in this subframe. In this case, the data channel PRN code of the subframe 4 has to be modified. This subframe is represented with violet color.
- Sub-frame 5 is transmitted during 30 seconds. This subframe carries the information necessary to provide each second the Integrity Status. This subframe is represented with blue color.

The navigation message structure proposed by THALES ALENIA SPACE suggests the introduction of the CSK technique inside the subframe 4. Subframe 4 is completely used to implement the CSK signaling technique except for some bits which determine the configuration of the content of subframe 5. Therefore, the number of available bits in subframe 4 which can be used to implement the CSK is 138. Some other parameters associated to the CSK signaling technique are not defined by this proposal, only some suggestions are made.

Annex E. Decoding methods

In this annex, the different decoding methods used or developed in this research work are presented.

E.1. Extended Hamming code (32, 26)

The extended Hamming code (32, 26) is the channel code implemented for the GPS L1 C/A signal. The encoding and decoding process of this channel code as well as the strategy of application of this code are explained next.

E.1.1. Encoding process

A Hamming code [PROAKISe, 2001] is a linear block code which can be systematic. A code is systematic if the information word is exactly found as a part of the code word, where the information word is the ensemble of information bits and the code word is the ensemble of coded bits. Besides, a block channel code normally transforms the information word to the code word by multiplying a vector defined from the information word by the generation matrix (G) of the channel code [PROAKISe, 2001].

The introduction of a parity bit is the process of adding to the n-bits code word an extra bit whose value is equal to the modulo-2 sum of the n bits of the code word. In other words, if the modulo-2 sum is equal to 0 the added bit is also equal to 0, but if the modulo-2 sum is 1 the bit is equal to 1. Another way of introducing the parity bit into the code word is to define a new generation matrix (G') for the extended channel code which takes into account the parity bit generation. In this latter case, the new generation matrix is the generation matrix of the extended code. In the GPS L1 C/A case, the original channel code is the (31, 26) Hamming code and the final extended code is the (32, 26) extended Hamming code.

Therefore, in order to obtain the GPS L1 C/A code word, the encoder should define an input vector of 26 information bits and multiply it by the generation matrix (G') in order to obtain the 32 bits code word. However, in the GPS L1 C/A navigation message, the size of the code word is equal to 30 bits where 24 are information bits and 6 are parity ones. This means that two information bits are discarded or not transmitted with the code word although they are necessary to execute the coding/decoding process. In fact, these 2 bits are the last 2 bits of the previous code word. In other words, the bit 29 and 30 (parity bits) of the code word *n* are used as the first 2 information bits of the information word *(n+1)*. Therefore, these 2 bits are not transmitted again with the code word *(n+1)* because their value is already known when the word *n* is received. The entire process is displayed on Figure E-1.

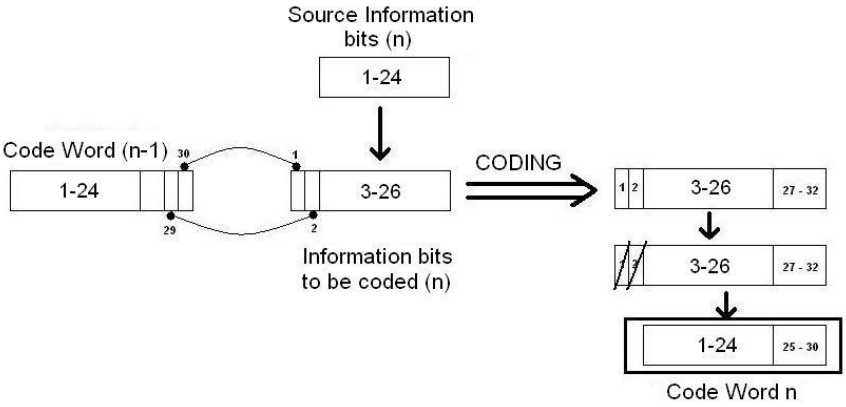

Figure E-1: Coding scheme of a word of the GPS L1 C/A navigation message

Finally, one important characteristic of the GPS L1 C/A code is that it is not systematic, since the bit 30 of the word *n* determines if the information bits values of the word *(n+1)* are inverted. Therefore, the information word (n+1) may not be found in the code word (n+1) but instead its complementary can be.

Another method to generate a GPS L1 C/A coded word is to use the algorithm presented in the document [ARINC, 2004].

E.1.2. Decoding process

The decoding process of a channel code consists in correcting the erroneous bits introduced by the transmission channel (annex C.4). Nevertheless, the extended Hamming code (32,26) was not originally designed to correct errors but rather to detect the erroneous words. Therefore, the GPS L1 C/A channel code strategy was to discard all the erroneous words and to only use the information of the a priori correct words.

The traditional linear block channel codes technique employed to detect if the received code word is free from errors consists in calculating a vector called syndrome. This vector is calculated from the multiplication of the received word by the parity matrix (H), where the parity matrix can be generated from the generation matrix [PROAKISe, 2001].

Once the syndrome has been obtained, the inspection of its value determines which pattern of errors is found on the word. The null syndrome, i.e. the all-zeros vector, indicates that the word does not have any error. Therefore, if the syndrome is the all-zeros vector, the receiver declares the word as error free; otherwise the word is discarded. Additionally, it is possible to correct some of the errors introduced by the transmission channel by using the syndrome vector. This method is fully described in the next reference [PROAKISe, 2001].

Nevertheless, the previously explained detection technique needs as input the complete received word, the word *n* plus the last two bits of the word *(n-1)*; which means that before calculating the syndrome, the original coded word has to be reconstructed. Figure E-2 illustrates this reconstruction.

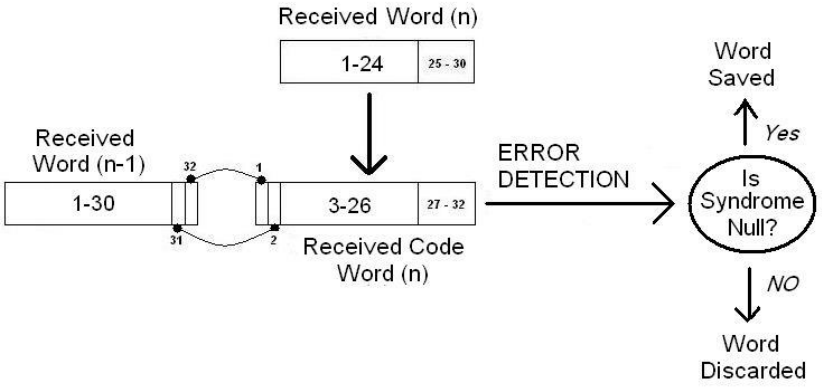

Figure E-2: Error detection scheme of a word of the GPS L1 C/A navigation message

Finally, another technique applied to detect the errors is described in this document [ARINC, 2004].

E.2. CRC-24Qualcom channel code

The CRC-24Q [ARINC, 2004] is the outer channel code implemented on the GPS L2C, GPS L5, GPS L1C and GALILEO E1 OS signal. The CRC-24Q is a cyclic channel code which generates 24 parity bits from any number of input information bits. The coding process is achieved through algebraic manipulation and the process is customized in order to make the code systematic.

The code word is equal to:

$$c(X) = m(X) \cdot X^{24} + r(X) \tag{E-1}$$

$$r(X) = \frac{m(X) \cdot X^{24}}{g(X)} \tag{E-2}$$

Where:

- c(X): Code Word
- m(X): Information word
- g(X): Generator polynomial
  - $g(X) = (1 + X) \cdot p(X)$
  - $p(X) = X^{23} + X^{17} + X^{13} + X^{12} + X^{11} + X^9 + X^8 + X^7 + X^5 + X^3 + 1$
- r(X): Remainder of the division between $m(X) \cdot X^{24}$ and q(X)

This remainder represents the cyclic code parity bits and the information word m(X) is expressed as:

$$m(X) = m_k + m_{k-1} \cdot X + m_{k-2} \cdot X^2 + \ldots + m_1 \cdot X^{k-1} \tag{E-3}$$

Where

- k: Number of input bits

Finally, the decoding process consists in dividing the received code word, a word provided by the inner channel code, by the generator polynomial g(X). In fact, this operation obtains a

syndrome and thus, if the syndrome is equal to the zero polynomial, the word is claimed as error free. And, since the channel code is systematic the information bits correspond to the first K-24 bits of the received code word, where K is the total number of information bits.

As has been said previously, the objective of the outer channel code is to detect and to discard the words provided by the inner channel code which still contain some error. Some properties concerning the CRC-24Q detection function are given below [ARINC, 2004]:

1. It detects all single bit errors per code word.
2. It detects all double bit error combinations in a codeword because the generator polynomial g(X) has a factor of at least three terms.
3. It detects any odd number of errors because g(X) contains a factor 1+X.
4. It detects any burst error for which the length of the burst is ≤ 24 bits.
5. It detects most large error bursts with length greater than the parity length r = 24 bits. The fraction of error bursts of length b > 24 that are undetected is:
    a. $2^{-24} = 5.96 \times 10^{-8}$, if b > 25 bits.
    b. $2^{-23} = 1.19 \times 10^{-7}$, if b = 25 bits.

E.3. Convolutional code

The convolutional code presented in this annex is the inner convolutional code implemented for GPS L2C signal, for GPS L5 signal and for GALILEO E1 OS signal. This channel code has a code rate, r, equal to ½ and two polynomial generators equal to G1 = 171o and G2 = 133o.

The encoding and decoding process of this convolutional code are presented next.

E.3.1. Convolutional channel code coding process

The coding process can be perfectly summarized by the following block scheme. See Figure E-3 [ARINC, 2004].

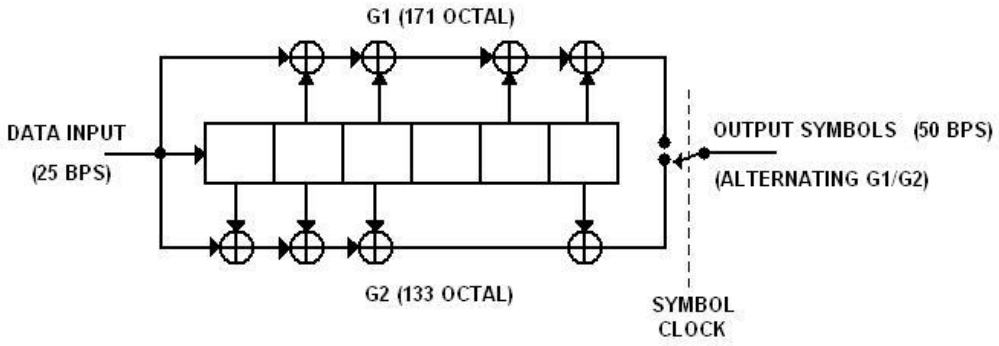

Figure E-3: Encoding block scheme of the convolutional code (171,133) [ARINC, 2004]

For each bit entering the linear shift register at its left side, the register calculates two new output bits. Each one of these 2 new output bits is calculated from a 2-modulo addition between the new entering bit and the previous L-1 input bits, where all the bits are weighted by a polynomial generator. Afterwards, the bit calculated from G1 is transmitted first to the

channel followed by the G2 bit. Finally, the oldest bit of the L-1 previous bits is eliminated and the entering bit is stored inside the encoder memory in order to prepare the system for the arrival of the next input bit.

In the previous paragraph, it has been said that the generation of the output bits depends on the polynomial generators (G1, G2), on the new input bit (bit *n*) and on the old ones (bits *n-1..n-L+1*). In fact, these old input bits are the memory of the code and they define the state of the encoder. Therefore, the output bits only depend on the input bit and on the state of the encoder. Moreover, the transition of one state to another is controlled by the input bit and thus some transitions are impossible. This is the one of the keys of the most common decoding method and has to be remembered.

E.3.2. Convolutional channel code decoding process

The inner channel code is a convolutional code, therefore the two main decoding methods are the Viterbi algorithm [VITERBI, 1967], [VITERBI and OMURA, 1979], and the BCJR (Bahl, Cocke, Jelinek and Raviv) algorithm [BAHL et al., 1974].

The Viterbi technique searches for the most probable sequence of symbols/bits being transmitted; or, in other words, this method minimizes the probability of error of the decoded sequence. The BCJR method searches the most probable transmitted symbol/bit; or, in other words, this technique minimizes the probability of error of each received symbol/bit. The most used decoding method is the Viterbi algorithm because, although the performance of both methods is almost the same, the Viterbi algorithm has smaller computational and storing costs. Therefore, the decoding technique described below is the Viterbi algorithm. Moreover, the BCJR technique does not allow the implementation of the technique presented in 0.

E.3.2.1. Viterbi Definitions

First of all, some definitions and concepts are given in order to better understand the Viterbi algorithm and in order to allow an easier comprehension of the algorithm. Remember that the detailed Viterbi algorithm can be found in [VITERBI, 1967], [VITERBI and OMURA, 1979].

The state, the state of the encoder or the state of the decoder is the set of the last L-1 information bits which have been encoded. The state altogether with the polynomial generators and the input information bit to be encoded determine the value of the code bit.

From the state notion, it can be said that each information bit sequence generates or travels a determined sequence of states. Therefore each sequence can be defined by the sequence of states that it travels. In this report, a sequence of states is called path, and thus each different sequence travels a different path.

Moreover, since the transition from one state to another one can only occur when a unique pattern of symbols is applied at the convolution encoder input, a unique path also determines a unique sequence of symbols. In this case, 1 symbol maps only 1 bit.

The trellis of a convolutional code is a time representation of all the possible paths that can be generated by the convolutional code. Obviously, each different path is generated by a different symbols sequence input. A simple trellis example is shown in Figure E-4. The trellis corresponds to a convolutional code (7, 5, r = ½) and it is only plotted until t = 4 because from there on the trellis is repeated; there is at least one path in each state.

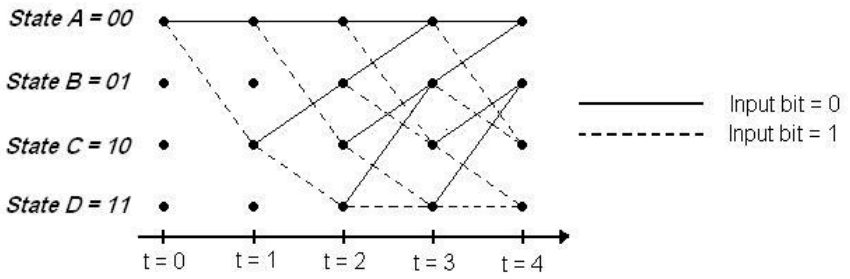

Figure E-4: Trellis of the convolutional code (7, 5, r = ½)

In Figure E-4, it can be seen that more than one sequence, in this case 2, arrive at the same state. In this dissertation, the action of two or more sequences arriving at the same state at the same instant of time is called as that these sequences merge into that state. Moreover, the state where the two or more sequences merge is called merging state.

Another important definition is the transition distance between two states (state X and state Y). This distance is calculated from the coded symbol values estimated by the receiver and the ideal coded symbol values expected between the states. A code symbol value estimated at time *n* by the receiver is the transmitted coded symbol value at time *n* plus the distortion and/or noise introduced by the transmission channel. An ideal coded symbol value expected between two states (state X and state Y) at time *n* is the coded symbol value generated by the encoder when its state at time *n-1* is the state X and the input information symbol/s makes the encoder to pass to the state Y at time *n*.

Depending on the type of decoding, soft or hard, this distance is calculated differently. In this PhD manuscript, the decoding is always soft; therefore the transition distance is evaluated as the Euclidian distance between the receiver estimated code symbol values and the ideal expected code symbol values:

$$Transition\ Distance\ of\ the\ k^{th}\ info\ symbol = \sum_{i=1}^{N} \left| r_{ki} - c^{XY}{}_{ki} \right|^2 \qquad \text{(E-4)}$$

Where:

- N: Number of coded symbols encoding an information symbol
- r_{ki}: i^{th} estimated coded symbol component by the receiver of the k^{th} estimated coded symbol (representing the k^{th} information symbol).
- $c^{XY}{}_{ki}$: Ideal i^{th} coded symbol from the k^{th} information symbol within the inspected sequence. The coded symbol values are generated by passing from state X to state Y.

The last definition is the sequence accumulated distance. The accumulated distance of a sequence is the sum of all the path transition distances; or in other words, the addition of all the individual transition distances of the transitions between the states forming the path associated to the sequence. Moreover, the accumulated distance of a sequence determines the probability for that sequence to be the transmitted sequence. In fact, the larger the accumulated distance of a sequence is, the smaller the probability for that sequence to be the transmitted sequence is. The same can be said for the inverse case.

E.3.2.2. Viterbi Algorithm

The main principle of the Viterbi algorithm is to choose the sequence with the smallest accumulated distance as the transmitted sequence because, as it has been said in the previous section, the sequence with the smallest accumulated distance is the most probable transmitted sequence. Therefore, the Viterbi algorithm should calculate the accumulated distance of all the possible sequences that can be transmitted. However, this calculation is not possible to achieve because the number of possible sequences grows exponentially with the sequences length. Therefore, the main principle of the Viterbi algorithm is to calculate the accumulated distances of all the sequences with possibilities of being the transmitted sequence. This means that is not necessary to wait until the end of the sequence reception to decide which the most probable sequence is. In fact, during the reception process, there are some sequences which can already be discarded since they do not have any possibility of being selected by the receiver. Moreover, note that the receiver cannot usually wait until the end of the sequence reception since the data input can be a constant stream of bits.

The reason for early sequence elimination is explained next. The accumulated distance of a sequence depends on the states travelled by the sequence. Therefore, when two or more sequences merge into a determined state, a first accumulated distance comparison is made. This comparison evaluates the path travelled until this moment for each individual sequence. The path of each sequence can be partially or completely different. Therefore, the accumulated distance of each sequence at this point shows which sequence is more probable of being transmitted until this point. Besides, this first comparison already determines which sequence among the two sequences will be the most probable transmitted sequence at any future instant. Or, in other words, the comparison determines that the sequence with the smallest accumulated distance at the merging state will always have the smallest accumulated distance among all the sequences merging at that instant at the merging state. In fact, from the merging state on, any possible path travelled by the largest accumulated distance sequence can also be followed by the smallest accumulated distance sequence. Therefore, since the additional accumulated distance calculated from the merging state depends only on the future path, a path which can be equal for both sequences, the sequence with the largest accumulated distance will always keep the largest accumulated distance in relation to the other sequence. Therefore, since the largest accumulated distance sequence will always be less probable than the smallest accumulated distance sequence, the former sequence can be eliminated without consequences.

To sum up, only one of the sequences merging in the state S at each time t survives whereas the others are eliminated. This path is called the survivor. An example is given below.

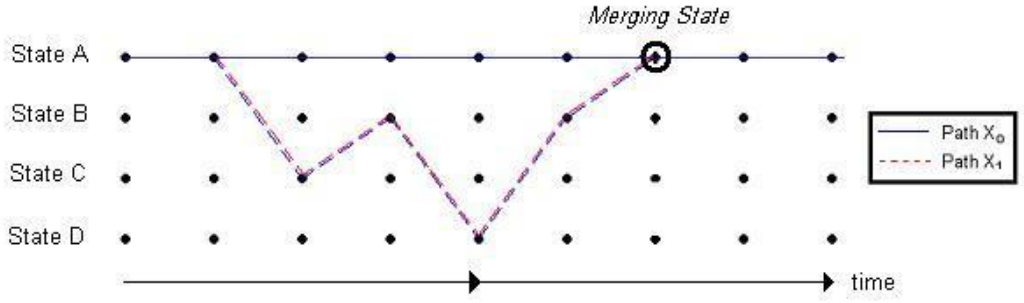

Figure E-5: Merging State and surviving path.

From Figure E-5, the X_0 path accumulated distance at the merging state was smaller than the X_1 path accumulated distance at the merging state. Therefore, the path X_0 is the survivor whereas the path X_1 is eliminated.

Once the justification of why not all the paths have to be conserved in order to find the more probable transmitted sequence has been explained, the Viterbi algorithm can be easily understood. First of all, a sequence of accumulated distance equal to 0 is stored at time t=0 for each possible state. Second, using the received estimated code symbols values and the ideal coded symbol values of each pair of states, the transition distances between t=0 and t=1 are calculated using equation (E-4). These transitions distances are added to the origin state survivors accumulated distance. In this case, the accumulated distance is 0. Once the sum has been done, each result is compared to the result of the other sequence/s merging in the same destination state. Finally, only the sequence with the smallest addition result is conserved whereas the others are eliminated. And the addition result is the new accumulated distance of the survivor. The process is repeated until the reception end when the sequence with the smallest accumulated distance is chosen as the transmitted one.

A scheme of the Viterbi algorithm at the iteration *n* is shown below.

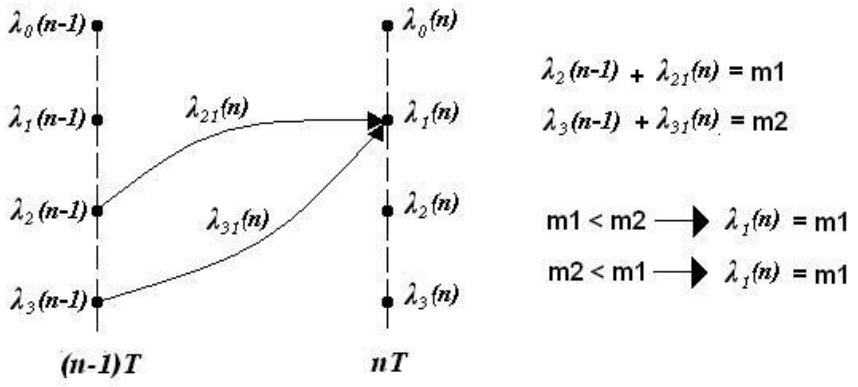

Figure E-6: Viterbi Algorithm - State Transition

Where:

- $\lambda_i(k)$ = Accumulated Distance for the survivor sequence at state *i* at the instant *kT*
- $\lambda_{ij}(k)$ = Transition Distance : Euclidian distance $d_E(X_{ij}, Y_K)$ between:
  - X_{ij}: Transmitted word between the i^{th} state and the j^{th} state (ideal coded symbols)
  - Y_k: Received Word at the instant *kT* (estimated code symbols)

This part of the algorithm is called ACS (Add Compare Select).

Finally, the choice of the most probable sequence is explained. Until now, the choice of the sequence with the smallest accumulated distance as the transmitted sequence has been avoided by assuming that at some instant the input of bits will stop. However, the input is usually a constant stream of bits and thus the decoder cannot wait until the transmission end, which does not exist, before providing the decoded bits. Therefore, the algorithm has to find another way of decoding the symbols while the transmission is still in process. In fact, a simple solution can be implemented since after several iterations, or state transitions, it is observed that each survivor has the beginning of its path equal to the other survivors

beginning of their path. This means that all the survivors have converged to the same path in the past. Therefore, the receiver can provide the information symbols corresponding to the common path part since it can be guaranteed that the most probable transmitted sequence would include these symbols. However, the path convergence does not always occur at the same past instant for each survivor. Besides, the time of path convergence is variable. Nevertheless, since it is known that the path of all the survivors converge with a high probability at time nT when the decoder is receiving the coded symbols of time $(n+\Delta)T$, the decoders can adopt the following strategy. At time nT, the receiver picks the path with the lowest actual accumulated distance, travels back the path of the chosen sequence ΔT seconds in time and selects as the transmitted information symbol the symbol marked by the selected path at the instant $(n-\Delta)T$. This effect is called Decoding Delay and the quantity Δ (= 3·L to 5·L, where L is the constraint length) is called Decoding Depth.

E.3.2.3. Insertion of tail bits in the information word

The insertion of tail bits in the information word has as effect to convert the convolutional code into a block code. This means that each block of bits is encoded independently with respect to another block of bits because the initial state and the final state of the encoder for each block of bits are known.

The GALILEO E1 OS navigation message implements the convolutional code explained in annex E.3 and introduces tail bits in order to convert the convolutional code into a block code. The tail bits are introduced at the end of the information bits of each part of a page which means that the input block of the code is the part of a page.

The number of tail bits is set to L-1, 6 in this case, in order to know the encoder register state after the tail bits encoding. As the tails bits are known from [ESA, 2008] (set to 0), the encoder always begins the encoding process of the next part of a page at the same state of the convolutional code. And this knowledge of the initial state transforms the convolutional code into a sort of block channel code. Moreover, since the initial and final states of the coded block are known, 0 state since the tail bits are equal to 0, the decoding performance is improved due to the decrease of the number of possible transmitted sequences. The only inconvenient of the transformation process of a convolutional code into a block code is the addition of L-1 additional bits; at least for the implementation used in GALILEO E1 OS.

An example of the effect of the insertion of the tail bits can be seen in Figure E-7. This figure represents the trellis of a convolutional code (G1, G2, r = ½) with L = 3. Therefore, the last two transmitted bits are the tails bits which are set to 0 forcing the last encoder state to the 0 state.

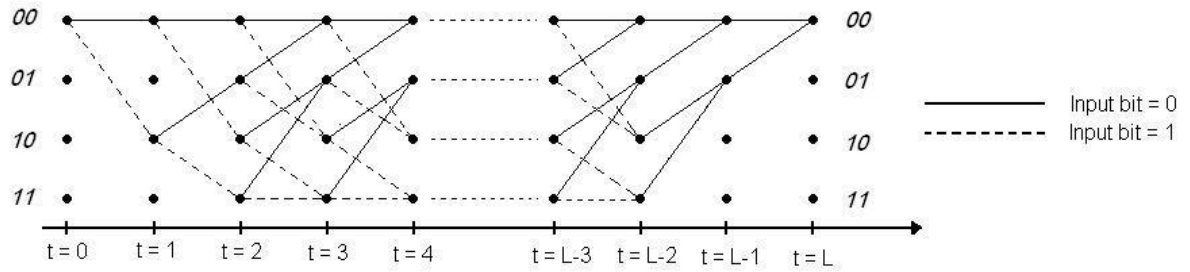

Figure E-7: Tail bits influence into the trellis of a convolutional code (G1, G2, r=1/2)

E.3.2.4. Viterbi algorithm with a priori bit probabilities

In order to better comprehend this section, we have presented the convolutional code encoding process and the Viterbi algorithm in annex E.3.1 and E.3.2.2.

The Viterbi algorithm objective is to minimize the error probability of the received sequence. Or, in other words, the fundamental expression consists of selecting the sequence, $s^{(m)}$, among all the possible existing sequences that have the highest probability of being the transmitted sequence when sequence r has been received.

$$\textit{Fundamental equation: } \max_{s^{(m)}} f\left(s^{(m)}\middle|r\right) \tag{E-5}$$

$$r_k = c_k + n_k \tag{E-6}$$

Where:

- m : Number of symbols of the source X alphabet
- $s^{(m)}$: Information sequence of K elements from source X alphabet
- $s_k^{(m)}$: k^{th} information symbol of the $s^{(m)}$ sequence
- r: Estimated coded sequence by the receiver of K elements
- r_k: k^{th} element or estimated coded symbol of the estimated coded sequence
- n_k: Additive Gaussian White Noise.
- c_k: k^{th} coded symbol of the coded sequence
- c_{ki}: i^{th} component of the k^{th} coded symbol
- $f(s^{(m)}|r)$: Probability of having transmitted the $s^{(m)}$ sequence when the r sequence has been received.

In order to better understand equation (E-5), one important remark concerning the information sequence, the information symbols, the coded sequence and the coded symbols for a convolutional code has to be made. An information symbol is formed by the bits fed to the convolutional code encoder at the same time, and each of these input bits is also called an information symbol component of this information symbol. A coded symbol is formed by the bits at the output of the convolutional encoder at the same time, and each one of these output bits is also called a coded symbol component of this coded symbol. Finally, an information sequence is the ensemble of all the transmitted information symbols and a coded sequence is the ensemble of all the transmitted coded symbols. Note that if the input is an infinite stream of bits the sequences have an infinite length.

For example, in the GPS L2C and GPS L5 signal case, the channel code rate is r = ½ , which means that the information symbol is formed by 1 information symbol component (1 bit) and that the coded symbol is formed by 2 coded symbol components (2 bits).

If we develop equation (E-5) by applying the Bayes theorem and since the denominator does not depend on $s^{(m)}$, the fundamental expression is equivalent to:

$$\text{Bayes Theorem: } f\left(s^{(m)}\middle|r\right) = \frac{f\left(r_1 \ldots r_K \middle| s^{(m)}\right) \cdot P\left(s^{(m)}\right)}{f\left(r_1 \ldots r_K\right)} \tag{E-7}$$

$$\max_{s^{(m)}} f\left(s^{(m)}\middle|r\right) = \max_{s^{(m)}} f\left(r_1 \ldots r_K \middle| s^{(m)}\right) \cdot P\left(s^{(m)}\right) \tag{E-8}$$

If we develop the right term of equation (E-8), we obtain:

$$f\left(r_1 \ldots r_K \middle| s^{(m)}\right) P\left(s^{(m)}\right) = f\left(r_K \middle| r_{K-1} \ldots r_1, s^{(m)}\right) \cdot f\left(r_{K-1} \middle| r_{K-2} \ldots r_1, s^{(m)}\right) \ldots f\left(r_1 \middle| s^{(m)}\right) \cdot P\left(s^{(m)}\right) \qquad \text{(E-9)}$$

Each individual term of the right part of equation (E-9) is conditioned by several factors; however, the majority of them can be removed by a close inspection of each component r_k.

First, r_k depends on the code symbol c_k and AWG noise n_k. Second, c_k is determined by the k^{th} information symbol, $s_k^{(m)}$, and the convolutional encoder state as has been explained in annex E.3.1. Third, the convolutional encoder state is determined by the L-1 previous information symbols, $s_{k-1}^{(m)} \ldots s_{k-L+1}^{(m)}$. Therefore, the other information symbols do not condition the r_k element. Fifth, since the n_k noises samples are independent among them and the c_k value is fully determined by the information symbols, the terms from r_{k-1} to r_1 influence on r_k can be removed from expression (E-9). Finally, equation (E-9) can thus be expressed as:

$$f\left(r_1 \ldots r_k \middle| s^{(m)}\right) P\left(s^{(m)}\right) = f\left(r_K \middle| s_K^{(m)} \ldots s_{K-L+1}^{(m)}\right) \cdot \ldots \cdot f\left(r_1 \middle| s_1^{(m)}\right) \cdot P\left(s_K^{(m)} \ldots s_1^{(m)}\right) \qquad \text{(E-10)}$$

Since symbols $s_k^{(m)}$ to $s_{k-L+1}^{(m)}$ determine the value c_k, the previous expression is equivalent to the following one:

$$f\left(r_1 \ldots r_k \middle| s^{(m)}\right) P\left(s^{(m)}\right) = f\left(r_K \middle| c_K\right) \cdot f\left(r_{K-1} \middle| c_{K-1}\right) \ldots f\left(r_1 \middle| c_1\right) \cdot P\left(s_K^{(m)} \ldots s_1^{(m)}\right) \qquad \text{(E-11)}$$

Therefore, it only remains to express the different probabilities terms, where each conditioned probability term is the determined by the probability density function of an AWGN:

$$f\left(r_k \middle| c_k\right) = \frac{1}{\sqrt{2\pi \cdot \sigma^2}} \cdot \exp\left(-\frac{1}{2 \cdot \sigma^2} \sum_i \left| r_{ki} - c_{ki} \right|^2 \right) \qquad \text{(E-12)}$$

Where:

- r_{ki} : i^{th} component of the k^{th} estimated coded symbol
- c_{ki}: i^{th} component of the k^{th} coded symbol
- σ^2 : AWGN variance

Therefore, equation (E-8) can be rewritten as:

$$\max_{s^{(m)}} f\left(s^{(m)} \middle| r\right) = \max_{s^{(m)}} \frac{1}{\left(2\pi \cdot \sigma\right)^{K/2}} \cdot \exp\left[\sum_{k=1}^{K} \left(-\frac{1}{2 \cdot \sigma^2} \sum_i \left| r_{ki} - c_{ki} \right|^2 \right) \right] \cdot P\left(s_1^{(m)}\right) \cdot \ldots \cdot P\left(s_K^{(m)}\right) \qquad \text{(E-13)}$$

Since $f(s^{(m)}|r)$ is always positive by definition, it is the same to search for its maximum or for the maximum of its logarithm. Therefore, the equivalent expression is:

$$\max_{s^{(m)}} f\left(s^{(m)} \middle| r\right) \Leftrightarrow \min_{s^{(m)}} \sum_{k=1}^{K} \left(\sum_i \left| r_{ki} - c_{ki} \right|^2 - 2\sigma^2 \log\left(P\left(s_k^{(m)}\right)\right) \right) \qquad \text{(E-14)}$$

Finally, a closer inspection of equation (E-14) provides the modification to apply to the original Viterbi algorithm. From equation (E-14), we can see that index k represents the different coded symbols forming the sequence. Therefore, the addition really represents the transition distances addition and the result represents the sequence accumulated distance. This means that we can identify from the individual addition term the modification which has been added to the original transition distance. This term is:

$$\textit{Additional transition distance term} = -2\sigma^2 \log\left(P\left(s_k^{(m)}\right)\right) \qquad \text{(E-15)}$$

Note that this additional term depends on the noise power and on the a priori bit probability at instant k. Therefore, the new transition distance is:

$$\textit{Modified Transition Distance} = \sum_i \left| r_{ki} - c_{ki} \right|^2 - 2\sigma^2 \log\left(P\left(s_k^{(m)}\right)\right) \qquad \text{(E-16)}$$

To sum up, the modified Viterbi algorithm is exactly the same as the original one but using equation (E-16) instead of equation (E-4).

E.4. Decoding method based on the combination of the inner convolutional code and the outer channel code

In this section, the decoding method based on the inner and outer channel codes combination which is used in 0 (see section 5.2.2) is further detailed. The inner channel code is the convolutional code presented in annex E.3 and the outer channel code can be the code presented in either the annex E.1 or the annex E.2.

The main idea of this method is to determine as the transmitted sequence, the sequence provided by the Viterbi decoding of the inner channel code which has the minimum accumulated distance and which verifies the outer channel code.

The main steps of this decoding method as well as its complete algorithm are detailed in this annex.

E.4.1. Search of the i^{th} candidate

The i^{th} candidate search is the identification of the sequence having the smallest accumulated distance among all the remaining sequences which are not yet candidates when (i-1) candidates have already been determined.

Before detailing the i^{th} candidate search, some definitions are presented. The particularities of the search of a candidate when the initial and/or the final state are known are also described.

E.4.1.1. Definitions of the method based on the inner and outer channel codes combination

In this subsection, some definitions required to correctly follow the explanation of the process of the i^{th} candidate search are presented. This subsection assumes that the reader is familiar with the candidate concept which is also an essential definition given in section 5.2.2.5 and with the Viterbi concepts given in annexes E.3.2.1 and E.3.2.2.

The *minimum alternative path* of the k^{th} candidate is defined as the path merging with the k^{th} candidate that has the minimal final accumulated distance among all the paths that, at any moment and at any state, merge once and only once with the k^{th} candidate. And once the minimum alternative path has merged with the k^{th} candidate, the minimum alternative path travels exactly the same path as the k^{th} candidate. One scheme representing the paths merging with a candidate is represented below.

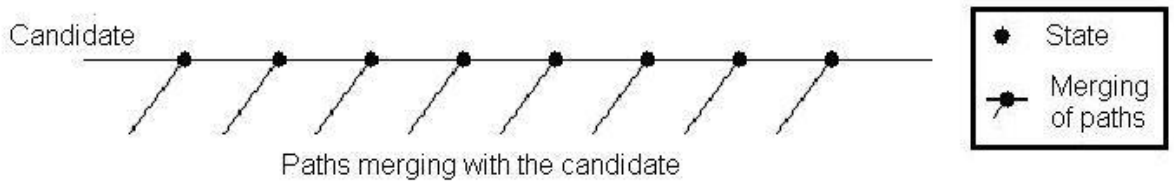

Figure E-8: Possible minimum alternative path - Sequences merging at any state with the k^{th} candidate

Another important definition is the *state difference*. The state difference of state X is the difference between the accumulated distances evaluated at state X of two sequences merging at state X.

The search of the minimum alternative path of the k^{th} candidate is equivalent to seek the state of the k^{th} candidate which has the minimal state difference among all the k^{th} candidate states. This state having the smallest state difference is called the *minimal state*. And once the minimal state has been found, we can directly determine the minimum alternative path of the k^{th} candidate. This path is defined in two parts. The first part is defined from t = 0 to the minimal state by the sequence which merges with the candidate at its minimal state. The second part is defined from the minimal state to the transmission end by the k^{th} candidate. Therefore, the minimum alternative path of the k^{th} candidate is completely defined in time by the two previous described parts.

The reason of this equivalence between the minimal state search and the minimum alternative path is presented next. From the definition of the minimum alternative path, we know that the minimum alternative path and the candidate follow the same path from their merging state. Therefore, the accumulated distance from the merging state until the transmission end is the same for both sequences. Consequently, the final accumulated distance of a sequence which merges with a candidate can be calculated by adding the final accumulated distance of the candidate to the accumulated distance of the sequence at the merging state with the candidate. In other words, the distance of the sequence is the final accumulated distance of the candidate plus the state difference of the merging state. Figure E-9 shows a graphical explanation.

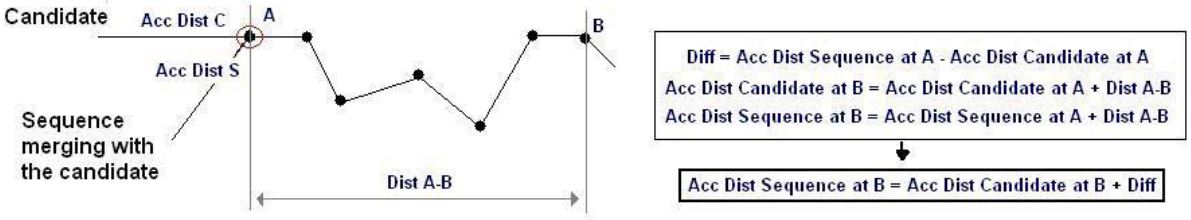

Figure E-9: Minimum alternative path selection justification

From Figure E-9 and from its previous explanation, it can be seen that the minimal state of a candidate is the merging state between the candidate and the minimum alternative path of the candidate.

Finally, we call *initial survivor* to each sequence surviving all the Viterbi decoding process. Therefore, the number of initial survivors is equal to the number of states. Moreover, even if the transmission has no end, at each instant of time the Viterbi has survivors although these survivors can be eliminated in the future.

E.4.1.2. Initial and final state knowledge effects

The main effect of the final and the initial state knowledge or their imposition is that all the possible decoded sequences begin at the same state A and finish at the same state B. Or, in other words, there is only one initial survivor. Therefore, it can be assured that the nearest sequence to another sequence in accumulated distance terms is a path that diverges from the original sequence at some instant t and merges again with it at some later instant t'. Consequently, one sequence which diverges twice or more times from the original sequence and thus merges twice or more with the original sequence can never be the nearest sequence to the original sequence in accumulated distance terms. The justification is presented next.

First, we assume a path X_1 diverging twice from the original path X_o. Second, we assume another path X_2 being equal to sequence X_1 until sequence X_1 merges for the first time with X_o at time t'. Afterwards, path X_2 follows the original path X_o. Therefore, we have assumed two paths, X_1 and X_2, having the same accumulated distance until t'. However, these paths have a different accumulated distance from t' until the transmission end. And it is path X1 which has a larger accumulated distance in relation to the path X_0 accumulated distance. The reason is that path X_2 follows exactly the same states as the original path X_o. Therefore, path X_2 is nearer to path X_o than path X_1. Consequently, since it is always possible to find a path X_2 for any path X_1, none path diverging twice or more times from the original sequence can be the nearest path to the original sequence. Figure E-10 shows a graphical example.

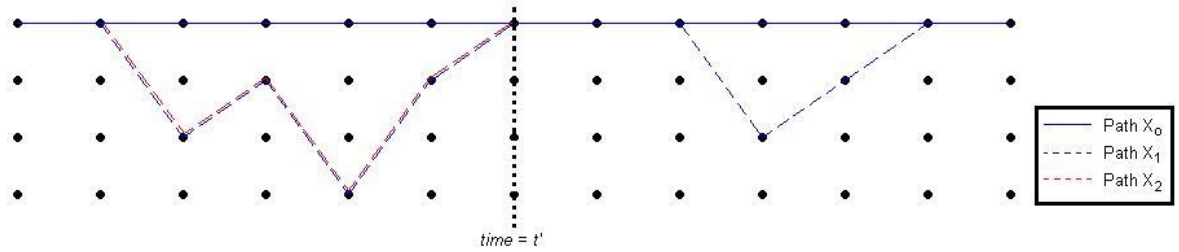

Figure E-10: Viterbi paths X_0, X_1 and X_2

The previous statement is only correct when all the paths begin and end at the same states. Therefore, if all the paths do not begin and do not end at the same state, nothing avoids that a random path is closer to the original path in terms of accumulated distance than the nearest path described previously. Figure E-11 shows a graphical example.

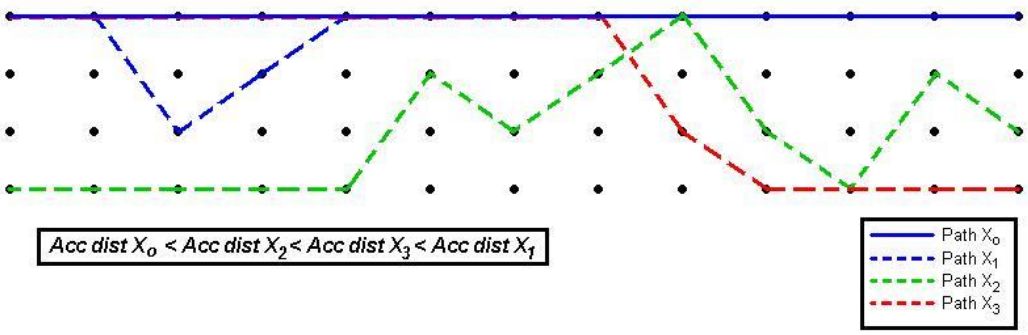

Figure E-11: Different possible Viterbi paths when the initial and the final state are not imposed

E.4.1.3. Search process of the i^{th} candidate

The search of the i^{th} candidate is accomplished by following the next 3 steps:

1. Searching among the initial survivors, which are yet not candidates, the one with the minimal accumulated distance.
2. Searching the minimum alternative path of each existent candidate.
3. Searching between the step 1 initial survivor and step 2 minimum alternative paths the sequence with the minimum accumulated distance. This sequence is the i^{th} candidate.

The justification is presented next. Nevertheless, in order to simplify the comprehension, we begin by explaining the case where the initial and final states are imposed, which means that there is only initial survivor of the Viterbi decoding.

The 1^{st} candidate is the only initial survivor of the Viterbi decoding since this sequence has the smallest accumulated distance. The 2^{nd} candidate is the minimum alternative path of the 1^{st} candidate, because as shown in section E.4.1.2, the nearest sequence to another one is a sequence which only diverges and merges once from the original sequence. And the minimum alternative path fulfills this requirement. The 3^{rd} candidate is searched from the 1^{st} and 2^{nd} candidates for two reasons. First, the 1^{st} candidate forbids the search of any sequence merging at the state where the 1^{st} and 2^{nd} candidate merged. Second, nothing avoids that the 2^{nd} candidate minimum alternative path has a smaller accumulated distance than the accumulated distance of the new minimum alternative path of the 1^{st} candidate. In fact, the 2^{nd} candidate travels some states which are different from the states of the 1^{st} candidate. Therefore, the state difference of the different states of 2^{nd} candidate from the 1^{st} candidate can be quite lower than any state difference of the states of the 1^{st} candidate. Figure E-12 shows a graphical explanation.

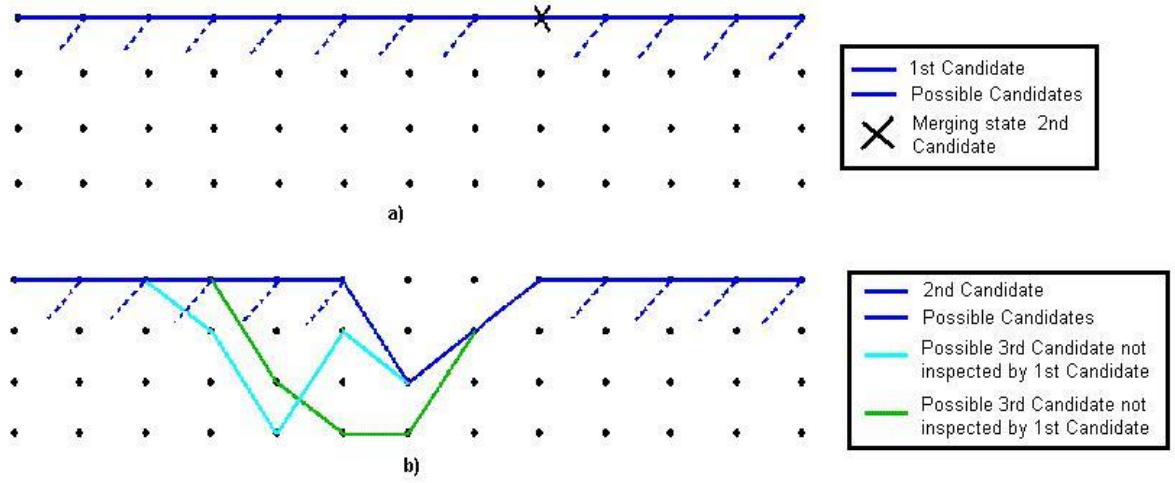

Figure E-12: a) Possible candidates from the 1^{st} Candidate. b) Possible candidates from the 2^{nd} Candidate.

The same justification can be applied for the search of the following candidates. When the initial and final states are imposed, the search of the k^{th} candidate is achieved following two steps. First, we search for the minimum alternative path of the candidates 1 to (k-1). Second, we choose among the minimum alternative path of the candidates, the minimum alternative path having the smallest accumulated distance as the k^{th} candidate. Note that each time that a candidate is selected, the merging state between the new candidate and the candidates from which this new candidate is created is eliminated from the search of the next candidates.

The explanation of the case where the initial state is unknown but the final state is known is presented next. In this case, the candidates can begin at any Viterbi state but they still have to end at the same known state. Therefore, there is still only one initial survivor which is selected as the 1st candidate. From this 1st candidate, the 2nd candidate is searched following exactly the same process as in the known initial and final states case. The reason is the following. The minimum alternative path of the 1st candidate continues to be the nearest sequence in accumulated distance terms to the 1st candidate. Besides, the initial state knowledge does not change the manner of searching a candidate minimum alternative path. In fact, the part of a sequence which is previous to its merging state with a candidate is irrelevant from the accumulated distance point of view since its influence is already observed with the state difference. And the initial state only has influence on this part of the path. Therefore, it does not matter whether before the merging state the minimum alternative path diverged from the candidate or whether it simply was a completely different sequence; the only important element for the accumulated distance calculation is the state difference. Consequently, the initial state knowledge does not modify the i^{th} candidate search; it only increases the number of possible transmitted sequences.

The case where the initial and final states are unknown is presented next. The main difference between the case where the final state is known and the case where the final state is unknown is the number of initial survivors. In the former case, there is only one initial survivor, whereas in the latter case, the number of initial survivors is equal to the number of states. This means that the choice of the 1st candidate is made by choosing among the initial survivors, the survivor having the smallest accumulated distance. The search process of the 2nd candidate is slightly modified from the process presented before. In this case, this process search cannot only consists in searching the minimum alternative path of the 1st candidate, since it cannot be guaranteed that this sequence is the closest one in accumulated distances terms to the 1st candidate. In fact, as said in section E.4.1.2, any of the remaining initial survivors can be nearer to the 1st candidate. Note that, an initial survivor can travel a 1st candidate path part or can follow a completely different path. Therefore, the search process needs to find the minimum alternative path of the 1st candidate (the nearest path to the 1st candidate if the final state was known) and needs to compare its accumulated distance to the accumulated distance of the other initial survivors. Then, the path having the smallest accumulated distance between the remaining initial survivor having the smallest accumulated distance and the minimum alternative path of the 1st candidate is the path which has the closer accumulated distance to accumulated distance of the 1st candidate. Therefore, this path is claimed as the 2nd candidate.

One important remark is that if an initial survivor is chosen as the 2nd candidate, the search of the future candidates employs this initial survivor as any other candidate. The reason is that we cannot guarantee that the minimum alternative path of the 2nd candidate has a larger accumulated distance than any minimum alternative path calculated from the 1st candidate or other candidates generated from it. The same is true for any other initial survivor chosen as candidate.

Finally, the justifications of the last two paragraphs can be applied for of any candidate from the 3rd candidate.

E.4.2. Generation of the i^{th} candidate

The generation process of the i^{th} candidate consists in forcing the i^{th} candidate survival during the Viterbi decoding process.

In this section, the generation process of the i^{th} candidate is presented. Nevertheless, before presenting the exact method, the objectives of the method and some definitions are given. Moreover, a detailed definition of the key element named restriction is also given.

E.4.2.1. Generation of the i^{th} candidate objectives

The main objective of the i^{th} candidate generation is to find the states travelled by the i^{th} candidate. From the i^{th} candidate travelled states, the information bits of the sequence can be extracted and the outer channel code can be applied in order to determine if this sequence is the transmitted one.

The generation of the i^{th} candidate has another fundamental objective. This objective is to segment the sequence space in order to simplify the search of the next candidates. We define the sequence space as the ensemble of possible coded sequences. This means that when the i^{th} candidate is generated, the sequence space associated to the merging state of the i^{th} candidate and the candidate from which the i^{th} candidate is generated is divided into two parts. One part is associated to the i^{th} candidate. The other part is associated to the candidate from which the i^{th} candidate is generated. Therefore, the sequences belonging to the sequence space associated to the i^{th} candidate can only be found by inspecting the i^{th} candidate. This means that the generation process of the i^{th} candidate has to introduce some mechanism in order to avoid the repetition of the same segmentation of the sequence space and in order to avoid the invasion of the sequence space of the i^{th} candidate by another candidate.

Nevertheless, note that some candidates can generate the same sequence since they can have the same minimum alternative path. And this fact can lead to the false impression that both candidates share the same segmented sequence space. More specifically, a sequence can be recovered from the n^{th} candidate and from the candidate from which the n^{th} candidate was generated; however, not all the state differences are the same for the sequence generated by the n^{th} candidate as for the sequence generate by the candidate from which the n^{th} candidate was generated. In fact, the segmentation of a sequence space refers to segment the sequence space of the sequences travelling a given state at a given time. This means that two or more candidates share a part of the general sequence space whereas they hold individually another part of the sequence space: no other candidate has this individual part of the segmented sequence space.

Apart from the two previous main objectives, there are two more requirements of the generation process of the i^{th} candidate which are also necessary in order to allow the correct functioning of this new proposed method.

First, in order to achieve the search process of the i^{th} candidate, the receiver had to employ the state differences of the states of the previous different candidates as specified in section E.4.1. However, the search process of the i^{th} candidate never explained how the state differences of the candidates were calculated. It simply assumed that their values were known beforehand. In fact, the i^{th} candidate state differences can only be calculated when the i^{th} candidate is forced to survive the Viterbi decoding process since it is the only moment when the accumulated distances of the paths merging with the i^{th} candidate can be compared to the accumulated distance of the i^{th} candidate. Therefore, it is during the generation of the i^{th} candidate when the state difference of the states of the i^{th} candidate must be stored.

Second, the generation of the i^{th} candidate is not easily achieved. In fact, if the previous candidates from i equal to 1 to i equal to i-1 are not eliminated during the generation of the i^{th} candidate, these candidates can cause two major complications. First, some of these previous

candidates can eliminate the i^{th} candidate because they merge at some state X with the i^{th} candidate and they have a smaller accumulated distance. Therefore, the sequence recovered after the Viterbi decoding process is not the i^{th} candidate. In other words, the method needs to correctly place itself inside the desired segmented sequence space in order to recover the desired i^{th} candidate. Second, some of these previous candidates can eliminate some sequence which could be a potential candidate because these previous candidates have a smaller accumulated distance. Therefore, in order to recover the correct i^{th} candidate and correctly store the differences of the states travelled by the i^{th} candidate, the previous candidates should be eliminated. In other words, the method has to avoid the elimination of any sequence belonging to the sequence space associated to the i^{th} candidate.

The elimination of previous candidates from the generation process of the i^{th} candidate generation can lead to the false conclusion that potential candidates merging at any state with these previous candidates are lost. This is not true because these potential candidates of the previous candidates are inspected during the search process of the minimum alternative path of the previous candidates. In other words, each possible candidate specifically belonging to a determined sequence space of a candidate is only inspected through the candidate.

E.4.2.2. Definitions of the generation of the i^{th} candidate

We call the candidate from which the i^{th} candidate is generated the *father* of the i^{th} candidate *x*. In other words, the i^{th} candidate is a minimum alternative path of its father. We also call a candidate generated from the i^{th} candidate the *descendant* of the i^{th} candidate. This means that an initial survivor does not have a father but can have zero, one or more than one descendants.

Additionally, the *ancestors* of the i^{th} candidate are the father of the i^{th} candidate, the father of the father of the i^{th} candidate, and so on until the initial survivor from which the candidates have been generated. Therefore, the father of the father of the i^{th} candidate is called the *grandfather* of the i^{th} candidate.

Finally, a *restriction* is an element associated to a couple state-time which chooses the survivor sequences between the two sequences merging at the state and at the time marked by the restriction. The restrictions are the elements used to segment a sequence space and to associate each resulting segment to a determined candidate. A more detailed definition is given in section E.4.2.5.

E.4.2.3. Generation process of the i^{th} candidate

The generation process of the i^{th} candidate is achieved by applying the Viterbi decoding process but forbidding the survival of the previous generated candidates. In other words, the receiver conducts a new Viterbi decoding process which removes the paths having a smaller accumulated distance than the i^{th} candidate and which does not affect at all the paths having a larger accumulated distance than the i^{th} candidate.

The method proposed consists in applying the Viterbi algorithm while:

1. Reproducing the exact conditions that allowed the generation of the father of the i^{th} candidate.
2. Adding some restrictions about other descendants of the father of the i^{th} candidate which were created between the generation of the father and the i^{th} candidate.

3. Forcing the receiver to choose the i^{th} candidate instead of its father at their merging state.

Note that the application of the previous points is in reality a segmentation of the sequence space. The justifications of the 3 points are presented next.

First, the exact generation of the father of the i^{th} candidate at the moment of its generation is necessary in order to guarantee that the i^{th} candidate is equal to the path anticipated by the search process of the minimum alternative path of the father. In fact, if the father is not rigorously reproduced, it is possible that previous candidates -which were already eliminated at the generation of the father- merge with the i^{th} candidate. Therefore, these previous candidates eliminate the it^{h} candidate due to their smaller accumulated distances which means that a previous candidate is reproduced instead of the i^{th} candidate.

Moreover, if the generation of the father of the i^{th} candidate at the moment of its generation is not exactly reproduced, some other sequences with larger accumulated distances than the i^{th} candidate can be erased by these previous candidates. Therefore, these other sequences are never examined since these previous candidates are eliminated later. This means that these other sequences will never be inspected even if they are the transmitted sequence. Besides, if the paths merging with the father are not the expected ones, the previous calculated state differences are false and the final accumulated distances are wrongly estimated.

In other words, the method needs to recover the original segmented sequence space of the father of the i^{th} candidate in order to find the i^{th} candidate. If the exact segmented space is not recovered, the i^{th} candidate and any other sequence can be modified or eliminated. Therefore, the further segmentation of the examined sequence space is not the correct one. And this leads to the loss of some sequence space segmentations and to the no inspection of some sequences.

To sum up, the exact generation of the father of the i^{th} candidate at the moment of its generation is required. Or, in other words, the exact regeneration of the conditions allowing the generation of the father of the i^{th} candidate is necessary in order to obtain the i^{th} candidate, in order to inspect any possible path, in order to obtain the correct state differences of the father and the i^{th} candidate, and in general, in order to recover the segmented sequence space of the father at the moment of its generation.

Additionally, note that in order to exactly generate the father at the moment of its generation, it is necessary to regenerate rigorously the grandfather at the moment of its generation too; more specifically, the moment where the grandfather state differences were calculated. Therefore, the generation of the i^{th} candidate is achieved by reproducing all the ancestors of the i^{th} candidate at the moment of their generation: the father, the grandfather, the father of the grandfather, etc, until an initial survivor is reached. Or, in other words, the desired segmented sequence space of the father is reproduced by starting in the general sequence space and by descending later to the specified segment.

Second, the restrictions imposed by other descendants of the father of the i^{th} candidate which were generated between the generation of the father and the generation of the i^{th} candidate, are required in order to avoid unnecessary candidate repetitions and in order to prevent the elimination of the i^{th} candidate by the other descendants of the father. First, when a new candidate is generated, the receiver has to mark on its father the state where the new candidate and the father merge. The marking consists in introducing a restriction in the determined couple state-time. In fact, if this state is not marked, the receiver will select again the merging state of the new candidate as the new minimal state during the search of the next candidate. Therefore, an unnecessary candidate repetition occurs and the process enters in a deadlock. Second, when the merging state of a previous descendant of the father is marked but it is not

taken into account in the generation of the i^{th} candidate, the i^{th} candidate is eliminated when it merges with the previous descendant at the previous marked merging state. The reason is that the previous descendant has a smaller accumulated distance than the i^{th} candidate. Therefore, the i^{th} candidate generation fails since the recovered sequence is not the desired one. Figure E-13 shows an example where if X_1 path is not restricted during generation of X_2 path from X_0 path, path X_2 is erased by X_1 path.

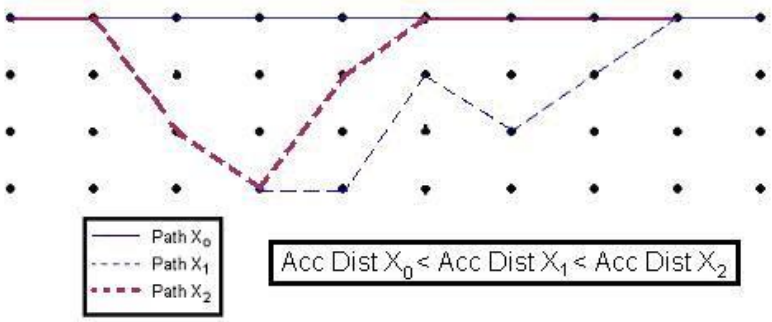

Figure E-13: i^{th} candidate elimination from previous i^{th} candidate father descendant

Moreover, note that this same type of restriction has to be also added to the generation of the grandfather in order to obtain the father at the moment of its generation since it cannot be guaranteed that the father is the first descendant of the grandfather. This means that the restrictions imposed by the generation of the previous descendants to the father of the grandfather of the i^{th} candidate have to be imposed to the father in order to allow its generation. These restrictions should be included into the first type of restrictions since they are necessary to reproduce the father, grandfather, etc.

Another form to interpret the second requirement is to think about the sequence space segmentation conducted with each descendant. In fact, each time that the father of the i^{th} candidate creates a descendant, the segmented sequence space associated to the father is divided into two parts. One part is associated to the father and the other to the descendant. Therefore, the segmented space associated to the father when the i^{th} candidate is generated has to be recovered from the father segmented space when the father had none descendants. And from this latter sequence space, the method needs to find the exact part associated to the father after the space sequence division among the previous descendants of the father of the i^{th} candidate.

Third and last, we force the Viterbi decoding process to make survive the i^{th} candidate instead of its father. This action is equivalent to divide the segmented space of the father into two and to associate one part to the i^{th} candidate and the other to its father.

E.4.2.4. Generation algorithm of the i^{th} candidate

The generation algorithm of the i^{th} candidate can be divided into two parts. The first part consists in executing the Viterbi algorithm with the restrictions commented in section E.4.2.3 in order to obtain the correct state differences of each travelled state. The second part consists in adding to the i^{th} candidate and to its father the new restrictions imposed by the i^{th} candidate in order to prepare the search and generation processes of the $(i+1)^{th}$ candidate. In other words, this second part defines the new segmented sequence spaces associated to i^{th} candidate and to its father. A more detailed explanation is given below.

The first part conducts the traditional Viterbi decoding method but with the particularity of making survive at the couple state-time marked by a restriction, the sequence indicated by this restriction. This means that all the restrictions conditioning the i^{th} candidate generation must be found before applying the Viterbi decoding method with restrictions. These restrictions are the following:

- Restrictions allowing the selection of the i^{th} candidate instead of the selection of any previous descendant of the father of the i^{th} candidate.
- Restrictions used to generate the father of the i^{th} candidate at the moment of its generation:
    - Restrictions allowing the selection of the father of the i^{th} candidate instead of the selection of any previous descendant -to the father- of the grandfather of the i^{th} candidate.
    - Restrictions used to generate the grandfather of the i^{th} candidate at the moment of its generation:
        - …
            - Restrictions allowing the selection of the father of father… of the i^{th} candidate instead of the selection of any previous descendant -to the father of the father…- of the father of the father… of the initial survivor of the i^{th} candidate.
            - All restrictions used to generate the initial survivor; which are none.
- Restriction selecting the i^{th} candidate instead of its father at the merging state.

Nevertheless, the previous scheme can be simplified by sorting out the previous restrictions into two groups, where each group of restrictions is individually stored inside a candidate. These groups are the restrictions defining the generation conditions of a candidate and the restrictions marking the descendants of a candidate.

First, the restrictions defining the generation conditions of a candidate are in reality the second group of restrictions of the father of the candidate at the moment of generation of the candidate. Therefore, the first group of restrictions of a candidate is only stored at the moment of its generation. Additionally, note that the restrictions defining the generation conditions of the father are the second group of restrictions of the grandfather of the candidate at moment of generation of the father. The same can be said for the restrictions defining the generation conditions of the grandfather and son on until arriving at the candidate who is an initial survivor. Therefore, if whenever a candidate is generated, the current second group of restrictions of its father is stored as the first group of restrictions of the candidate, the regeneration of any candidate is easily accomplished since all its ancestors have already stored the restrictions allowing their own regeneration.

In other words, the first group of restrictions defines the segmented sequence space to be divided between the i^{th} candidate and its father. And this segmented sequence space to be divided is defined from the segmented sequence space of the father at the moment of its generation.

Second, the restrictions marking the candidate descendants have two different origins. The first origin is that a descendant cannot be generated at a given state and at a given time because the state of the candidate is already marked as a merging state with an ancestor. Or, in other words, the receiver can only generated an ancestor from this state. Consequently, these restrictions are extracted from the second type of restrictions of the father of the candidate at the moment of generation of the candidate, but note that not all the restrictions are stored. In

fact, if the father has a restriction imposed at a certain time t at a state A and the candidate is not at this time t at the state A, this restriction does not have to be applied to the candidate. The reason is that the candidate at this time t is at another state B which merges with a still not inspected sequence; a sequence which can be a future descendant of the candidate. Therefore, this inclusion of this restriction in the candidate removes in reality this uninspected sequence instead of an ancestor of the candidate. Moreover, since it is possible that the father of the candidate does no travel all the state of the candidate although some ancestors can, the restrictions of the first origin have to be also searched among all the candidate ancestors first group of restrictions in order to avoid the unnecessary regeneration of any ancestor.

The restrictions of the second origin are the descendants generated by the candidate. In this case, the receiver seeks to avoid future unnecessary candidate repetitions. Therefore, the restrictions of the second origin are stored along the generation of the descendants of the candidate.

In other words, the second group of restrictions defines the current segmented sequence space associated the i^{th} candidate. This current segmented sequence space is defined from the segmented sequence space of the i^{th} candidate at the moment of generation of the i^{th} candidate.

Finally, the search of the restrictions for the generation of the i^{th} candidate is thus reduced to:

- Second group of restrictions of the father of the i^{th} candidate.
- First group of restrictions of the father of the i^{th} candidate.
- First group of restrictions of the grandfather of the i^{th} candidate.
- …
- Restriction selecting the i^{th} candidate instead of its father at the merging state.

Once the restrictions allowing the generation of the i^{th} candidate, the second part of the generation algorithm of the i^{th} candidate is explained. This part consists in storing the two previous defined groups of restrictions each time that a new candidate is generated. More specifically, the receiver has to store the following restrictions:

- **On the second group of restrictions of the father of the i^{th} candidate:** the restriction signaling the i^{th} candidate-father couple merging state-time. We associate to the father its new segmented sequence space after the generation of the i^{th} candidate.
- **On the first group of restrictions of the i^{th} candidate:** the current second group of restrictions of the father. We store into the i^{th} candidate the segmented sequence space of its father before dividing the sequence space and associating a part to the i^{th} candidate. These restrictions are never modified and none restriction can be added to them.
- **On the second group of restrictions of the i^{th} candidate:** the restriction signaling the i^{th} candidate-father couple merging state-time. We associate to the i^{th} candidate its first segmented sequence space.
- **On the second group of restrictions of the i^{th} candidate:** the current second group of restrictions of the father which has a couple state-time travelled by the i^{th} candidate. We associate to the i^{th} candidate its first segmented sequence space.
- **On the second group of restrictions of the i^{th} candidate:** The first group of restrictions of the father and of the ancestors of the i^{th} candidate which has a couple state-time travelled by the i^{th} candidate. We associate to the i^{th} candidate its first segmented sequence space.

E.4.2.4.1. Example

This example illustrates the algorithm of the previous section. First, we assume that a k^{th} candidate is generated from the 1^{st} candidate and that the restrictions of the 1^{st} candidate marking its previous descendants to the k^{th} candidate -which have been used to generate the k^{th} candidate- are stored into the k^{th} candidate. Second, we assume that another n^{th} candidate is generated from the k^{th} candidate where $n > k$. In this case, the restrictions of the 1^{st} candidate stored into the k^{th} candidate have been used to recreate the exact k^{th} candidate generation and the own restrictions of k^{th} candidate previous to the generation of the n^{th} candidate have been employed to mark the previous descendants of the k^{th} candidate. Then, once the n^{th} candidate has been generated, these restrictions of the descendants of the k^{th} candidate previous to the n^{th} candidate are stored inside the n^{th} candidate. Finally, we assume the generation of the m^{th} candidate from the n^{th} candidate where $m > n > k$. In this case, the employed restrictions have been first the restrictions of the 1^{st} candidate stored into the k^{th} candidate. These restrictions allow the exact regeneration of the k^{th} candidate. Second, the restrictions of the k^{th} candidate stored into n^{th} candidate are used in order to generate the n^{th} candidate. In fact, once the k^{th} candidate has been regenerated, these previous restrictions allow the exact regeneration of the n^{th} candidate from the k^{th} candidate. Third and last, the restrictions of the descendants of the n^{th} candidate previous to the m^{th} candidate are used in order to generate the m^{th} candidate.

E.4.2.5. Restrictions

A restriction is an element that determines which sequence between two sequences entering a specific couple state-time has to survive. Or, in other words, a restriction privileges one path among all the others paths entering a state. Therefore, the restrictions are used in order to allow the survival of a path with a larger accumulated distance than another path entering into a state.

The restriction is the element employed by the proposed method which is responsible for dividing the initial sequence space into smaller sequence space segments. Therefore, the structure of a restriction is defined in order to segment the sequence space. In fact, the segmentation of the sequence space is applied to each individual couple state-time. This means that the method seeks to separate the sequence which enters into state S at time t coming from state S', from the sequence which enters state S at time t coming from state S''. Therefore, a restriction consists of 3 elements: the time t when the restriction is applied, the state S where the restriction is applied, and the state S^{-1} from which the survival path enters the merging state.

E.4.3. Duration of the Viterbi decoding and bit recovery

A convolutional code can have as input an infinite stream of bits. This means that the receiver cannot wait until the transmission end, which does not exist, before beginning the decoding process.

The application of the traditional Viterbi algorithm can guarantee with a high probability the convergence of a part of all the survivors sequences at a determined transmission instant as said in annex E.3.2.2. More specifically, the Viterbi algorithm can guarantee with a high probability that the i^{th} symbol of all the survivors sequences is the same when the $(i+5 \cdot L)^{th}$ symbol is being received. Obviously, this probability increase for received symbols previous to the i^{th} symbol.

Therefore, since the proposed method is basically a Viterbi modification, the proposed method has to extend its Viterbi decoding until all the bits belonging to the same coded word have been determined with a high certainty. In other words, for the GPS L2C and GPS L5 navigation messages, the number of received symbols to process in order to ensure that the value of all the bits of a coded word have been provided with a high certainty by the Viterbi decoding process is M+5·L, where M depends on the decoded message. More specifically, M is equal to 300 bits for GPS L5 and GPS L2C mode CNAV data, and M is equal to 30 bits for GPS L2C mode NAV data with FEC.

Finally note that, due to the inclusion of the additional 5·L bits in the Viterbi decoding process, the decoding process of the n^{th} coded word requires the use of the bits of the next coded word. More specifically, for GPS L5 and GPS L2C mode CNAV data, some $(n+1)^{th}$ coded word bits are required whereas for GPS L2C mode NAV data with FEC all the $(n+1)^{th}$ coded word bits are used plus some $(n+2)^{th}$ coded word bits. Nevertheless, remember that only the M bits of the n^{th} coded word are actually decoded.

E.4.4. The Algorithm

The detailed algorithm of the inner and outer channel code combination method is presented next. The part of the algorithm concerning the verification of the outer code is not presented since this process is explained in annex E.1 and E.2. The maximum number of searched candidates before stopping the research and discarding the received word is set to K.

The algorithm is described below:

- Initialization:
  1. **Array of Distances:** 2^{L-1} double values to store the accumulated distances of the initial survivors.
  2. **Array of Survivors:** 2^{L-1} Booleans to mark the initial survivors which have already been used as candidates.
  3. **Matrix of Differences:** (M + $5 \cdot L$) columns, K rows. Double values to store the state differences of the candidates.
  4. **Matrix of Candidates:** (M + $5 \cdot L$) columns, K rows. Integer values to store the K travelled states of the candidates.
  5. **Matrix of Descendants:** (M + $5 \cdot L$) columns, K rows. Booleans to mark the descendants of the K candidates.
  6. **Matrix of Generated:** (M + $5 \cdot L$) columns, K rows. Booleans to mark the descendants of the father of the candidate at the moment of generation of the candidate.
  7. **Array of Accumulated Distances:** K doubles values to store the accumulated distances of the K candidates.
  8. **Array of Fathers:** K integer values to store the identifier of the father of the candidate.

- Searching and creating the 1^{st} Candidate:
  1. Complete Viterbi decoding process execution: (M + 5·L) received symbols.
  2. Store the accumulated distance of the initial survivors at the Distances array.

3. Select the initial survivor with the smallest accumulated distance as the 1st candidate.
4. Store the state differences of the 1st candidate into Differences matrix row 1.
5. Store the travelled states of the 1st candidate into Candidates matrix row 1.
6. Store the accumulated distance of the 1st candidate into Accumulated Distances array cell 1.
7. Store the value -1 into Fathers array cell 1; marking a no valid father.

- Searching the i^{th} Candidate

1. Loop: k = 1 to (i-1)
   1.1. Search for the minimum alternative path of the k^{th} candidate which is not yet a descendant. In other words, searching the minimal state of the k^{th} candidate.
      1.1.1. Initialize variable MIN to a big number e.g. 1000000000.
      1.1.2. Loop: j = 1 to (M + 5·L)
         1.1.2.1. Verify that the state travelled by the k^{th} candidate at time j has not already produced a descendant (inspect Descendants matrix row k column j).
            1.1.2.1.1. Verification fails: the system jumps to explore the next state.
            1.1.2.1.2. Verification succeeds: the system compares the state difference value stored into Differences matrix row k column j to variable MIN. If (MIN > state difference), the state and its state difference are stored, and MIN = state difference.
      1.1.3. The minimum alternative path of the k^{th} candidate is defined by:
         1.1.3.1. Accumulated distance: Accumulated Distances matrix cell k value plus MIN variable value.
         1.1.3.2. Merging state S: Candidates matrix row k column j.
         1.1.3.3. Previous merging state S^{-1}: Candidates matrix row k column (j-1).
         1.1.3.4. Candidate final state S_f: Candidates matrix row k column (M+5·L).
2. Search for the minimum alternative path with the smallest accumulated distance among all previously found minimum alternative paths in step 1.
3. Search for the initial survivor with the smallest accumulated distance among all the initial survivors still not being a candidate: the Survivors array cell marks if the initial survivor is already a candidate.
4. Choose the sequence with the smallest final accumulated distance between the step 2 and step 3 paths as the i^{th} candidate.

- Generate the i^{th} Candidate from the minimum alternative path of the k^{th} candidate:

Each time that a restriction is recovered, the system has to recover the time t, the merging state S and the state previous to the merging state S^{-1} from the corresponding sequence states stored inside the Candidates matrix.

1. Recover the restrictions used to generated the k^{th} candidate:

a. Active-Father is the father of the i^{th} candidate (the k^{th} candidate).

b. 1) If Active-Father is not the 1^{st} candidate, the system recovers all the restrictions of the father of Active-Father marked on the Active-Father Generated matrix row. The states S and S-1 are searched inside the father of Active-Father Candidates matrix row.

   2) If Active Father is the 1^{st} candidate, the restrictions search is over and the system goes to step 2.

c. Active-Father is the Active-Father father.

d. Going to step b.

2. Recover the restrictions of the k^{th} candidate marked into Descendants matrix row k. The states S' and S^{-1}' are searched inside the k^{th} candidate Candidates matrix row. Add only the restrictions that have not been recovered in step 1.
3. Add the restriction where the i^{th} candidate eliminates its father: time t, merging state S and previous state S^{-1}.
4. Store the father identifier id_f. This identifier marks the row where the father characteristics are stored into the different several matrices.
5. Store the father final state S_f.
6. Complete execution of the modified Viterbi decoding process: $(M + 5 \cdot L)$ received symbols. This execution differs from the normal one due to the restrictions imposed by the previous points:

   6.1. Restrictions used to generate the k^{th} candidate (Step 1 and 2 restrictions): The restrictions impose that the survivor path entering the merging state S at time t' is the sequence that comes from the previous state S^{-1}.

   6.2. i^{th} candidate restriction (Step 3 restriction): The restriction imposes that the survivor path entering the merging state S at time t is the sequence that does not come from the previous state S^{-1}.

7. Save into Candidates matrix row i the states of the path arriving at state S_f (i^{th} candidate).
8. Save into Differences matrix row i the state differences of the i^{th} candidate.
9. Marking into Descendants matrix row k column t that state S has been used to generate the i^{th} candidate. Row k stores the information of the descendants of the father of the i^{th} candidate.
10. Copy Descendants matrix row k, the restrictions of the descendants of the father of the i^{th} candidate, to Generated matrix row k.
11. Mark into Descendants matrix row i all the cells corresponding to a couple state-time travelled by the i^{th} candidate where a restriction has been applied. Note that the time specifies the cell inside the row i where the restriction has to be marked.
12. Storing into the Accumulated Distances array cell i the accumulated distance of the path arriving at the state S_f (i^{th} candidate).
13. Storing into Father array cell i the identifier of the father of the i^{th} candidate, id_f.

- Generate the i^{th} Candidate from an initial survivor:

1. Identify the final state of the initial survivor, S_f.
2. Complete execution of the Viterbi decoding process: (M + 5·L) received symbols.
3. Store the travelled states of the initial survivor into Candidates matrix row i.
4. Store the states differences of the states travelled by the initial survivor into Differences matrix row i.
5. Store into Accumulated Distances array cell i the accumulated distance of the initial survivor.
6. Store into Father array cell i the value -1 marking that the initial survivor, or i^{th} candidate, does not have a father.
7. Mark into Survivors array cell S_f that the initial survivor having as final state S_f has been used as candidate.

E.4.5. Variations of the decoding method

In this section, the 6 variations of the decoding method based on the inner and outer channel codes variation are detailed.

E.4.5.1. Viterbi method with a priori bit probabilities

The inner and outer channel code combination method implements the Viterbi algorithm as the inner channel code. Therefore, any method improving the Viterbi algorithm decoding should improve the proposed method performance. In this case, we propose to introduce the use of the a priori bit probabilities as explained in section 5.2.1.3.

Remember that the use of a priori bit probabilities helps the Viterbi decoding to obtain a lower BER than the BER obtained with the traditional technique. Therefore, in this case, the a priori bit probabilities increase the probability of finding the transmitted sequence among the inspected candidates. In other words, due to the use of a priori bit probabilities, after the Viterbi decoding process the receiver observes, in average, fewer sequences than before which have a smaller accumulated distance than the actual transmitted sequence. Therefore, the proposed method has a higher probability of finding the transmitted sequence in addition to finding the sequence more quickly.

This variation can only be included in the GPS L2C mode NAV data with FEC signal since it is the only signal which has the navigation data message encoder equal to the GPS L1 C/A navigation data message before being fed to the convolutional.

E.4.5.2. Sliding Window

The variation of the algorithm proposed in this section is relative to the Viterbi decoding length and the decoded bits. Therefore, before presenting the modification, the reason why the change of the Viterbi decoding length improves the method decoding performance is given.

The main algorithm proposes to apply the Viterbi decoding over a code word length, M symbols, plus 5·L additional symbols. Remember that a code word is a set of bits protected by the outer channel code. The reason of this Viterbi decoding length is to reach the convergence

of all the survivor sequences over the M symbols to be decoded before estimating the symbols values. However, we could search for another decoding length which optimizes the decoding performance in addition to guarantying the symbols convergence.

In fact, after examining the source of errors of the proposed method, we propose to search for the most probable sequence spanning N coded words which verifies the outer channel code of each one of the N code words. Note that the expansion of the candidate from 1 code word to N code words is possible since at the end of each code word the calculation of the accumulated distance of the sequence is not interrupted and the final state is not fixed. This new decoding length, (N·M+5·L) symbols, implies an increase of the computational cost of the process since the number of possible sequences grows exponentially with the number of decoded symbols (N·M). This means that this algorithm variation depends on the trade-off between the decoding performance gain and the extra computational cost.

The justification of why the decoding performance is improved is explained by analyzing the source of errors of the method. The source of errors of the method is the limitation of the outer channel code to detect any possible pattern of errors contained by a candidate, i.e. its detection capacity. Note that the Viterbi algorithm does not introduce single independent errors but bursts of errors. Consequently, if we consider a candidate as the transmitted sequence plus a pattern of errors, we can have wrong sequences verifying the outer channel code due to the previous commented limitation. This means, that the search for the transmitted sequence is stopped and the wrong code word is provided to the GPS receiver.

Two different patterns of errors can be distinguished depending on their position inside the wrong code word: at the middle of the code word or at the code word edges. First, the pattern of errors located at the code word edges implies that either the initial state of the code word, the final state of the code word or both states are different from the correct states of the code word. Moreover, this pattern of errors also affects the previous or posterior code word. Figure E-14 shows an example of a burst of errors located at two consecutive code words edges.

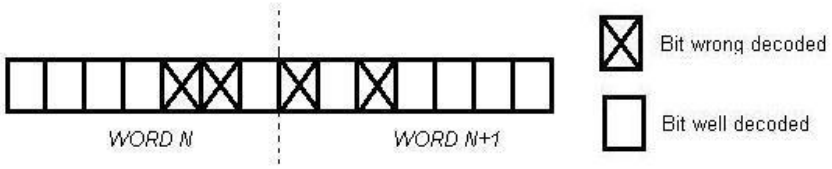

Figure E-14: Pattern of errors situated at the edge of 2 consecutive code words

From Figure E-14, it can be seen that if the proposed decoding method is applied at the same time for both codewords, the shared pattern of errors can be corrected. In fact, when the pattern of errors is situated at the code word edge, it is quite probable that the candidate which spans on 1 codeword and which verifies the outer channel code of this 1st code word fails the verification for the 2nd codeword when it is extended to the 2nd code word; the candidate is now the union of two code words. The justification of this previous statement is that the probability of two erroneous code words passing the outer channel code verification is much smaller than the probability of only one code word verifying the outer channel code.

Second, the patterns of errors located at the middle of the code word are the patterns of errors which do not have any influence on the initial and final code word states. Therefore, the initial and final states of the code are the same for the correct code word as for the erroneous code words. This means that the decoding of an extra word cannot help the correction of the pattern of errors. The justification is that since the final state of the 1st code word is the same for the correct coded word as for the erroneous code word, the candidate extension until the end of

the 2nd code word is the same for the correct code word as for the erroneous code word. Therefore, although the 1st code word is wrong, the 2nd word cannot contain any error.

One remark that can be made about the extension of the decoding process to for example 2 code words concerns the situation where the 1st code word is correct but the 2nd word is erroneous. In this case, the error on the 2nd code word can avoid the correct decoding of the 1st code word and thus the decoding performance gets worse. Nevertheless, it has been shown through simulations that the extension of the candidate to more than one word increases the decoding performance of the method. And this means that the decoding performance gain obtained through the increase of the probability of detection is higher than decrease of decoding performance due to the impossibility of recovering the 1st correct code word if the 2nd one cannot be corrected.

One last remark to make about the modification of the Viterbi decoding length is the number of bits that can be extracted from the single decoding of a candidate. In fact, each time that the decoding length is extended to N words and a candidate verifies the outer channel code of each one of the N code words, the algorithm variation should only recover the bits of the first N-1 code words. The reason is that the Nth code word can still have a pattern of errors located at its right edge. Therefore, in order to decode these bits of the Nth code word, the algorithm implements a sliding window containing several code words where the last code word is never decoded but is used for the search of the extended candidates. More specifically, N code words define a candidate but only the bits of the first (N-1) code words are decoded; after, the Nth code word plus the next (N-1) words define a new candidate and the process begins again.

The sliding window algorithm is given below:

- ith window:
  - Words (i-1)·(N-1)+1 to (i-1)·(N-1)+N define the new candidate
  - Search of the candidate defined by these N words which verifies the outer channel code of each one of the N code words.
  - Decoding the bits of the code words (i-1)·(N-1)+1 to (i-1)·(N-1)+(N-1)

Finally, this variation of the decoding method increases significantly the navigation message GPS L2C mode NAV data number of correct decoded words; however its implementation has been discarded for GPS L2C mode CNAV data and GPS L5 due to the low CRC-24Q probability of non-detection [ARINC, 2004].

E.4.5.3. Sequence initial state imposition

This algorithm variation consists in imposing the initial state of the candidates. In other words, each time that a candidate passes the outer channel code verification, its final state is used as the initial state of the next candidate-to-search. However, if the previous candidate is not found, the initial state of the next candidate is not imposed.

The imposition of the initial state of a candidate implies that the number of sequences which can be candidates is reduced because only the sequences having their initial state equal to the imposed state are considered. Therefore, several advantages and one drawback are introduced by this fact.

The first advantage is that the possibility of providing the outer channel code with an erroneous code word passing its verification is lower. This means an improvement of the BER performance. The second advantage is that the probability of finding the transmitted sequence

among the K inspected candidates is increased. This also means an improvement of the BER performance. The third and final advantage is that since there are less erroneous code words between the 1st candidate and the emitted sequence, the time of the decoding process is reduced.

The drawback of the algorithm variation is due to the impossibility of the channel outer code to detect all the erroneous code words. This means that when an erroneous code word passes the verification of the outer channel code, the imposed initial state to the next candidate-to-search is false. Therefore, the next transmitted sequence can never be found by the proposed decoding method since all the candidates to be inspected will have the same different initial state from the initial state of the transmitted sequence. This means an increase of the BER. Nevertheless, this drawback is neglected for GPS L2C mode CNAV data and GPS L5 due to the low CRC-24Q probability of missing a wrong word [ARINC, 2004].

This variation of the decoding method is implemented for all the signals, GPS L2C mode NAV data with FEC, GPS L2C mode CNAV data and GPS L5.

E.4.5.4. Algorithms variations speeding up the execution of the proposed method

The 3 algorithm variations presented in this section increase the execution speed of the combination of channel codes method. The first and most important consequence of applying any of the 3 variations is the impossibility of guaranteeing that the method is still able to find the correct transmitted sequence when assuming an infinite number of candidates and a perfect detection capacity of the outer channel code.

More specifically, the application of any speed improvement is equivalent to reduce the sequence space in which the transmitted sequence is searched. In other words, the receiver no longer search for the transmitted sequence among all the possible sequences, the receiver limits its search to a reduced quantity of sequences. These sequences discarded are considered too improbable to be the transmitted sequence. And this means that if the transmitted sequence is one of these beforehand discarded sequences, the receiver cannot find the transmitted sequence.

Fortunately, there is always more than one sequence which contains the bits of the transmitted codeword. And this means that not all the sequences containing the bits of the transmitted codeword are lost due to the application of the speed improvements. In fact, since the last $5 \cdot L$ bits used during the Viterbi decoding process are part of a candidate but only the previous M bits represent the bits of the transmitted code word, all the sequences having the first M bits equal to the bits of the transmitted code word are considered as valid sequences regardless of their last $5 \cdot L$ bits. Therefore, not only the transmitted sequence contains the bits of the transmitted code word. Remember that only the first M bits have to pass the outer channel code verification. Nevertheless, note that the actual transmitted sequence should be the first sequence to be inspected among all the sequences containing the bits of the transmitted code words since this sequence should have the smallest accumulated distance.

Moreover, due to the possible elimination of all the sequences containing the bits of the transmitted code word, the application of any speed improvement can imply the increase of the probability of providing an erroneous code word to the receiver. In fact, when the sequences containing the bits of the transmitted code word are eliminated, the method has to reach the maximum number of inspected candidates before determining that any sequence containing the bits of the transmitted code word cannot be recovered. And this means that there are more erroneous sequences being tested by the outer channel code verification which

leads to more erroneous code words than usual provided to the higher layers of the receiver. Nevertheless, if the transmitted sequence is not eliminated, the effect is the opposite because there are probably few erroneous sequences between the 1st candidate and the transmitted sequence. Moreover, in this case, a decrease of the number of erroneous sequences between the 1st candidate and the transmitted sequence lead to the increase of the probability of having the transmitted sequence among the K inspected candidates.

Finally, three different types of speed optimizations are proposed in this section: the elimination of some sequences having the same last 5·L symbols, the elimination of some initial survivors and the elimination of candidates with the same final accumulated distance. Additionally, their application on the different GPS messages is commented.

E.4.5.4.1. Sequences having the same last 5·L symbols elimination

This algorithm variation consists in eliminating the sequences which diverge from a candidate at some of its last 5·L states and which merge again with the candidate at the a later state. In other words, if a sequence diverging from i^{th} candidate at state S, where S belongs to the last 5·L states, merges again to the i^{th} candidate at a later state S', where S' belongs to the last 5·L states and S < S' in time, the sequences can be eliminated from the search of the minimum alternative path of the i^{th} candidate.

This implementation of this algorithm variation is applied during the generation of each candidate. More specifically, once the i^{th} candidate has been generated, the receiver first searches for the sequences fulfilling the previous characteristics. And second, the receiver marks as created descendants the states where the candidate merges with the previous determined sequences. Note that this implementation can only be made during the generation of each candidate because it is the only moment where the states of the sequences merging with the candidate are available.

The justification of the implementation of this algorithm variation is given next. The last 5·L bits of a candidate only are used to obtain the convergence of the first M bits of the sequence; the bits which have to pass the outer cannel code verification. In fact, the values of these last 5·L bits do not determine the bits of the transmitted code word. However, since these bits are part of the candidate, the proposed method can waste some time searching different candidates with the same first M bits but with different last 5L bits. More specifically, the proposed method can search minimum alternative paths which do not change the candidate previous M bits and thus the outcome of the outer channel code verification is not modified. This means that method is wasting time and resources.

Nevertheless, remember that the application of any speed algorithm variation implies that not all the original sequence space is inspected. Therefore, the proposed algorithm modification cannot inspect the transmitted sequence. An example is given next. Although an X_0 sequence merging at some of the last 5·L states of the candidate do not modify the first M bits, it cannot be guaranteed that a X_1 sequence merging with X_0 sequence does modify these bits. Therefore, X_1 sequence could pass the outer channel code verification if X_1 sequence was not eliminated due to this improvement. A graphical example is given below in Figure E-5:

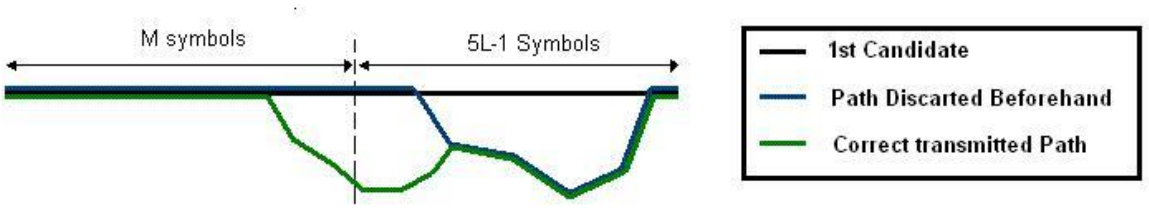

Figure E-15: Correct transmitted sequence loss due to the 5L paths improvement

Finally, we tested this speed improvement and we determined that it is highly improbable to lose the correct transmitted sequence due to this improvement.

E.4.5.4.2. Elimination of initial survivor

This algorithm variation consists in eliminating the initial survivors having the same first M states as another initial survivor already inspected as candidate. In other words, if an initial survivor diverges from another initial survivor, already inspected as a candidate, at state S belonging to the some of the last 5·L states, the initial survivor can be eliminated.

This implementation of this algorithm variation is made during the generation as candidate of each initial survivor. More specifically, the receiver executes the complete Viterbi decoding without any restriction and selects the desired initial survivor as a candidate. Then, the receiver searches for the remaining initial survivors, still not being candidates, which have the same M states as the initial survivor being generate as candidate. Finally, these selected initial survivors are marked as generated candidates in order to avoid their future generation.

Moreover, note that this modification can be implemented each time that a candidate is generated. In this case, since the Viterbi decoding has been executed with some restrictions, the initial survivors do not have to be the same initial survivors obtained when the Viterbi decoding was executed without restrictions. Nevertheless, the principle is the same and thus the initial survivors obtained by a Viterbi decoding with restrictions can also be eliminated if they have the same first M states as the generated candidate.

The justification of the implementation of this algorithm variation is the same as the speed improvement commented in annex E.4.5.4.1. Since some initial survivors do not change the first M bits of a candidate which has already failed the outer channel code verification, the receiver does not have to waste time and resources inspecting more sequences which do not change these first M bits.

Nevertheless, this speed improvement also removes some segments of the sequence space from the search of the transmitted sequences and thus, the correct transmitted sequence can again be lost. The same example as the example presented in section E.4.5.4.1 can be applied here. The only difference is that the sequence diverging from the candidate does not merge again with the candidate. Figure E-16 presents a graphical example.

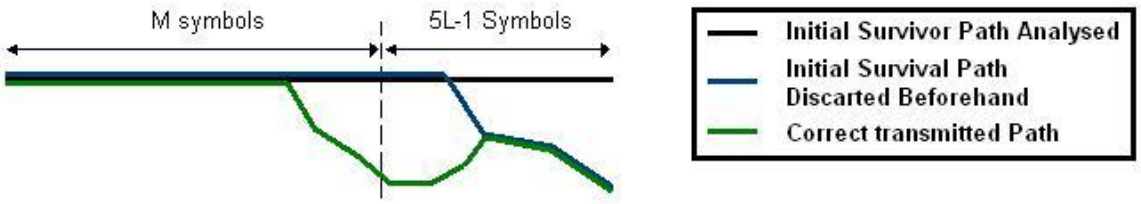

Figure E-16: Correct transmitted sequence loss due to the initial survivor paths elimination improvement

Finally, we tested this speed improvement and we determined that it is highly improbable to lose the correct transmitted sequence due to this improvement.

E.4.5.4.3. Elimination of sequences having the same accumulated distances

This algorithm variation consists in eliminating the sequences having the same accumulated distance as a candidate. In other words, if a sequence has exactly the same accumulated distance as a candidate, it means that the sequences and the candidate are in reality the same sequence: the candidate and the sequence having the same accumulated distance as the candidate are the same sequences but generated from different fathers. Therefore, the receiver eliminates the new sequence since this sequence has already been generated as candidate.

This implementation of this algorithm variation is made during the minimum alternative path search of the k^{th} candidate. More specifically, if the receiver saves the accumulated distance of the previous candidate, the receiver can compare this value with the accumulated distance of the minimum alternative path of the k^{th} candidate. Therefore, if the distances are equal, this means that both sequences are the same and thus, the receiver marks into the k^{th} candidate that a descendant has already been generated from the merging state of k^{th} candidate with the minimum alternative path.

Nevertheless, this application of this algorithm variation, as well as the two previous speed improvements, implies that some segment of the sequence space is not inspected during the sequence search. Therefore, the transmitted sequence can be lost. The reason is given next. Although the inspected sequences are equal to a candidate because they have the same accumulated distances, their generation conditions or, in other words, the restrictions of their ancestors are different. Consequently, some states differences of each sequence having the same accumulated distance can be different. This means that at each different state difference, each sequence having the same distance has a different sequence merging in this state. And each different sequence can be a different future candidate. Therefore, this technique does not explore the complete sequence space.

An example illustrating the previous explanation is shown in Figure E-17.

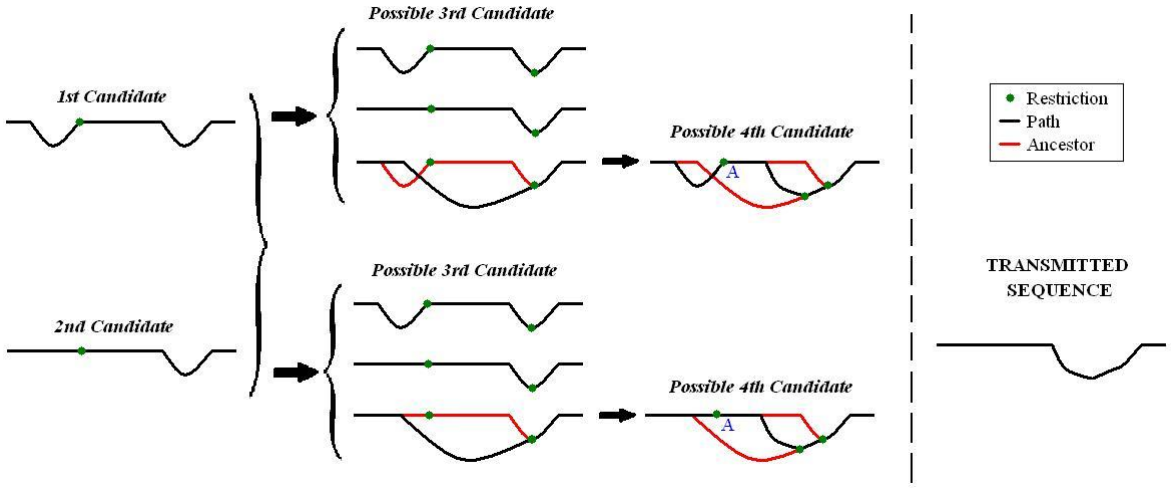

Figure E-17: Correct transmitted sequence loss due to the same distance paths elimination improvement

Figure E-17 shows the following scenario. First, two candidates are created, one from the other, and the restriction which divides their sequence space segment is marked in green. Second, the algorithm searches for and creates the 3rd candidate either from the 1st candidate or from the 2nd candidate. We can see that the states of the 3rd candidate do not depend on the identity of the father of the 3rd candidate. Therefore, if the proposed modification is applied, the new restriction is marked over the 1st candidate as well as over the 2nd candidate regardless of the identity of the father of the 3rd candidate. Third, two different 4th candidates can be generate depending on the father of the 3rd candidate. More specifically, we can observe that the 4th candidate generated by the 3rd candidate when the 2nd candidate was the father of the 3rd candidate can never be generated by the 3rd candidate when the 1st candidate was the father of the 3rd candidate. And the opposite is also true. In fact, the restriction imposed by the 1st candidate does not allow that the part of the 4th candidate until point A is flat. Consequently, since the sequences which can be generated from the 3rd candidate when its father is the 1st candidate cannot be flat before the A point, the transmitted sequence cannot ever be recovered.

Finally, we observed that the cases when this algorithm variation implies the no inspection of the transmitted sequence are really infrequent. Moreover, this algorithm variation is really significant in speed terms because in some cases the receiver tends to spend a lot of time generating the same candidate. Therefore, due to the favorable relationship between the speed improvement and the number times that the transmitted sequence is not inspected this improvement becomes very attractive.

E.4.5.4.4. Application of the speed improvements on the GPS signals

The effects of the application of the speed up improvements on the GPS navigation messages having a different outer channel code have to be differentiated.

E.4.5.4.4.1. Application of the speed improvements on the GPS signals having the CRC-24Q as outer channel code

The CRC-24Q channel code has a very low probability of non-detection [ARINC, 2004]. This means that although the application of the speed up improvements can make the receiver not to inspect the transmitted sequence, the receiver hardly accepts any erroneous code word as

the transmitted code word due to the high detection probability of the CRC-24Q. In fact, none of the conducted simulation has ever accepted an erroneous word and the total number of tested words is about 4 million words. In other words, for the GPS message implementing the CRC-24Q channel code, the only source of error is that the maximum number of candidates, K, is reached without finding a sequence containing the bits of the transmitted code word.

To sum up, since the proposed speed modifications increase a lot the execution speed of the method, help the introduction of the transmitted sequence among the K candidates and the number of times that a sequence space containing the transmitted sequence is discarded beforehand is very low, the GPS messages implementing the CRC-24Q outer channel code should always apply all the speed improvements.

E.4.5.4.4.2. Application of the speed improvements on the GPS signals having the extended Hamming as outer channel code

The main difference between the CRC-24Q case and the extended Hamming case is the detection capacity of the channel code. In this case, the detection capacity is much lower than the detection capacity of the CRC-24Q channel code. This means that the receiver will select as the transmitted sequence an erroneous sequence more often than in CRC-24Q case. In fact, for the CRC-24Q case, the receiver will usually inspect the K candidates without finding a sequence which passes the outer channel code verification whereas for the extended Hamming case, it cannot be guaranteed.

Another difference is the size between the words when the CRC-24Q code is applied and the words when the extended Hamming code is applied. For the CRC-24C code, the code word size is equal to 300 bits whereas for the extended hamming code this number is reduced to 30. Therefore, the number of sequences is drastically reduced, and the probability of discarding beforehand the transmitted sequence when the speed improvements are applied is increased for the extended hamming channel code with respect to the CRC-24Q channel code.

Nevertheless, despite these two significant drawbacks, the improvement based on the elimination of the sequences having the same accumulated distances is still worthy and even necessary. More specifically, there are a lot more cases where the receiver reaches the maximum number of candidates, K, without finding the transmitted sequence than cases where the receiver accepts an erroneous sequence because the transmitted sequence is discarded beforehand.

However, the application of the other two speed improvements is not so evident since too many sequences are discarded beforehand with respect to the total number of sequences. Therefore, we only apply the speed improvement based on the elimination of the paths having the same last 5·L bits elimination improvement since it provides a better increase of the execution speed of the algorithm than the other improvement.

To sum up, for the extended hamming case, we apply the improvements based on the elimination of the sequence having the same last 5·L bits and on the elimination of the sequences having the same accumulated distances, but not the improvement based on the eliminate of the initial survivors.

E.5. Low Density Parity Check (LDPC) channel code

Two different LDPC channel codes are implemented on the subframes 2 and 3 of the GPS L1C signal [ARINC, 2006].

A LDPC code is a linear channel code block which can be systematic by construction. The GPS L1C LDPC subframe 2 and 3 channel codes are systematic and different for each subframe. The LDPC channel codes provide a very high performance: the BER versus Eb/N_0 relationship is very close to the theoretical limit calculated by Shannon, and known as the Shannon Limit [PROAKISd, 2001]. This high performance is obtained when the channel codes are applied on very long messages but this is not the case for subframe 2 and 3 sizes. However, the generation matrix, G, of each channel code subframe has been optimized in order to achieve a high decoding performance. In fact, the factors establishing the channel code performance, BER versus E_b/N_0, are the G matrix of the LDPC code altogether with the chosen decoding process. Therefore, since subframe 2 and 3 sizes are different, the G matrices are different, the LDPC codes are different and thus their performance is also different.

The coding process is equal to any other linear block channel code: it is an algebraic multiplication between the generation matrix and the information word expressed as a vector. However, for the GPS L1C navigation message, another method is defined in [ARINC, 2006].

In this work, the typical LDPC decoding method developed by D.J.C. MacKay and R.M. Neal is used. The main principle is commented but the reader is directed to [MACKAY and NEAL, 1995] [MACKAY, 1999] for a detailed explanation.

The main principle of the decoding method is to calculate and propagate probabilities. There are two types of probabilities, the bits probabilities and the checks probabilities. They are calculated using the parity matrix, H (orthogonal matrix to the generation matrix). Remember that the checks of a parity matrix are the rows of this matrix, where the multiplication between a check and the received code word has to be equal to 0 in order to decide that the coded word does not contain any error. Therefore, a received code word has to verify all the checks before deciding that the word is error free.

The bits probabilities depend on the checks where the bits appear and the checks probabilities depend on the bits values which are used to calculate the check. Obviously, these two types of probabilities depend one to the other and thus and iterative algorithm of propagation has been established. Finally, the decoded information word is found when all the checks are verified (are equal to 0) or the maximum number of iterations is reached without deciding any word. In this last case, the last iterated word is delivered but indicating that it is not error free.

The complete algorithm can be found in [MACKAY and NEAL, 1995] [MACKAY, 1999].

Annex F. Probabilities of the binary prediction and influence of the bits forming the ephemeris to the described orbit

In this annex, all the probabilities analyses related to the binary prediction studied in Chapter 4 are presented in detail. Moreover, the influence of the bits forming the different Keplerian parameters on the orbit described by them is inspected.

F.1. Theoretical calculations of the required performance level of the binary prediction

In this section, the theoretical analyses conducted by this dissertation in order to determine the performance level of the binary prediction required to fulfill the conditions imposed by the Chapter 4 case of study are presented.

First the probability definitions are presented. Second, their mathematical expressions are calculated.

F.1.1. Probability definitions

In order to find the error probability of the first position error bit, this annex develops the equation already given in section 4.2.4.3.

$$P(0e\ aR\ TP) + P(1e\ aR\ TP) \geq 99\% \tag{F-1}$$

Where

- P(0e aR TP): Probability of having 0 user position errors after the RAIM application during TP time
- P(1e aR TP): Probability of having 1 user position error after the RAIM application during TP time
- TP: Time period defined by the case of study

Once the equation is reminded, this annex continues by dividing each individual probability into components, and continues to divide the resulting components into more elements until these elements can be calculated.

However, before beginning with the inspection of the terms, two important parameters have to be introduced. The first parameter is the number of satellites in sight and the second one is the number of independent positions that have to be calculated during the case of study period of time. Note that each independent user position is estimated with different satellite ephemeris data sets at different instants:

- Number of satellites: M
- Number of estimated positions: N = 12 (ephem/day)* Number of days inside the defined time period

Therefore, the previous expression (F-1) is equivalent to:

$$P(0e\ aR\ Npr) + P(1e\ aR\ Npr) \geq 99\% \qquad \text{(F-2)}$$

Where:

- P(Xe aR Npr): Probability of having X user position errors after the RAIM application when estimating N user positions

The first tem of equation (F-2) is equal to:

$$P(0e\ aR\ Npr) = P(0e\ bR\ Npr) + \sum_{k=1}^{N} P(ke\_1se\ bR\ Npr) \cdot P(Rd\_ke|ke) \qquad \text{(F-3)}$$

Where

- P(0e bR Npr): Probability of having 0 user position errors before the RAIM application when estimating N user positions
- P(ke_1se bR Npr): Probability of having k user position errors, each one caused by 1 individual satellite position error before applying the RAIM when estimating N user positions
- P(Rd_ke|ke): Probability of RAIM detecting k user position errors, where each user position error is caused by only 1 satellite position error. The total amount of existing user position errors is k.

And each term P(ke_1se bR Npr) can be separated into the sum of several other terms. Nevertheless, in order to simplify the study, before giving a general expression, each term is analyzed individually.

$$P(1e\_1se\ bR\ Npr) = \sum_{i=1}^{N} \sum_{n=1}^{M} P(Sat_n\_1e\ Sat_{\neq n}\_0e\ Npr)_i \qquad \text{(F-4)}$$

Where:

- $P(Sat_n\_1e\ Sat_{\neq n}\_0e\ Npr)_i$: Probability of having a position error of the satellite n during the i^{th} user position estimation. The position of the satellite n is correct for the other user position estimations (≠i). The positions of the remaining satellites (≠n) are always correct. The total number of user positions estimated are N.

$$P(2e\_1se\ bR\ Npr) = \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j \neq i}}^{N} \sum_{n=1}^{M} \sum_{m=1}^{M} P(Sat_{n,m}\_1e\ Sat_{\neq n,m}\_0e\ Npr)_{i,j} \qquad \text{(F-5)}$$

- $P(Sat_{n,m}\_1e\ Sat_{\neq n,m}\_0e\ Npr)_{i,j}$: Probability of having a position error of the satellite n during the i^{th} user position estimation and of having a position error of the satellite m during the j^{th} user position estimation (m can be equal to n). The position of the satellite n is correct for the other user position estimation (≠i) and the position of the satellite m is correct for the other user position estimation (≠j) (except when m = n). The remaining satellites positions (≠n,m) are always correct. The total number of user positions estimated are N.

Therefore, the general expression can be extrapolated from the previous (F-4) and (F-5) mathematical equations.

Once the first term of equation (F-2) has been detailed, the second term can be also defined as a sum of terms.

$$P(1e\ aR\ Npr) = \sum_{k=1}^{M} P(ke\_1se\ bR\ Npr) \cdot P(Rd\_(k-1)e|ke) + \\ + \sum_{k=1}^{M} P(1e\_kse\ bR\ Npr) + P(>2e\ bR\ Npr) \quad \text{(F-6)}$$

Where:

- P(ke_1se bR Npr): Probability of having k user position errors, each one caused by only one satellite error position before applying the RAIM when estimating N user positions.
- P(Rd_(k-1)e|ke): Probability of RAIM detecting and correcting k-1 user position errors, where each user position error is caused by only 1 satellite position error. The total amount of existing user position errors is k and the total number of user positions estimated are N.
- P(1e_kse bR Npr): Probability of having 1 user position error caused by k satellite position errors before applying the RAIM when estimating N user positions..
- P(>2e Npr): Probability of having 2 or more user position errors, where 1 user position error among the total number of user position errors is caused by at least two satellite position errors and the remaining user position errors are caused by only one satellite position error before the applying the RAIM. This probability takes into account the probability of the RAIM detecting and correcting all the user position errors caused by only one satellite position error. The total number of user positions estimated are N.

The last term P(>2e Npr) can be expressed as.

$$P(>2e\ Npr) = \sum_{n=1}^{M} P(2e\_1se\_nse\ bR\ Npr) \cdot P(Rd\_1e|1e) + \\ + \sum_{n=1}^{M} P(3e\_1se\_1se\_nse\ bR\ Npr) \cdot P(Rd\_2e|2e) + \ldots + \\ + \sum_{n=1}^{M} P(Ne\_1se \ldots 1se\_nse\ bR\ Npr) \cdot P(Rd\_(N-1)e|(N-1)e) \quad \text{(F-7)}$$

Where:

- P(2e_1se_nse bR Npr): Probability of having 2 user position errors before applying the RAIM. One user position error is caused by only one satellite position error, and the other error is caused by n satellite position errors.
- P(3e_1se_1se_nse bR Npr): Probability of having 3 user position errors before applying the RAIM. Two user position errors are caused by only one satellite position error, and the remaining user errors are caused by n satellite position errors.
- P(Ne_1se…1se_nse bR Npr): Probability of having N user position errors before applying the RAIM. N-1 user position errors are caused by only one satellite position error, and the remaining user position errors are caused by n satellite position errors.

Finally, note that the previous probabilities given in equations (F-6) and (F-7) can also be expressed as equations (F-4) and (F-5).

F.1.2. Probability calculations

Once the probabilities definitions involved in the case of study have been presented, the mathematical expression of each probability, or the addition of some of them, is presented.

Nevertheless, before giving the mathematical expressions of the probabilities given in section F.1.1, the RAIM detection probabilities are presented since they are more easily modeled. Taking into account the assumption saying that each satellite position prediction is independent from the prediction of the other satellite positions and the prediction of the same satellite position but in a different instant of time, the probabilities can be modeled as a binomial variable.

$$P\left(Rd\_ke \middle| ke\right) = P_{dR}{}^{k} \tag{F-8}$$

$$P\left(Rd\_(k-1)e \middle| ke\right) = k \cdot P_{dR}{}^{(k-1)} \cdot \left(1 - P_{dR}\right) \tag{F-9}$$

Where:

- P_{dR}: Probability of RAIM detecting the user position error.

Therefore, since the algorithm fault detection function is able to detect any failure equal or larger than 600m with a non-detection probability of 10^{-3} for a mono-frequency receiver in a urban environment [ESCHER, 2003], the detection probability P_{dR} is equal to:

$$P_{dR} = 1 - 10^{-3} \tag{F-10}$$

Now, in order to express erroneous user position probabilities, we use the independence among user position errors in time and in space, as said in section 4.2.4.2. Therefore, a user position estimation is an event independent of these other user position estimations. This means that the ensemble of the user position estimations during the specified time period can be modeled as a binomial variable.

$$P\left(ne\ Npr\right) = \binom{N}{n} \cdot P_{no\_error}{}^{N-n} \cdot \left(1 - P_{no\_error}\right)^{n} \tag{F-11}$$

Where:

- P(ne Npr): Probability of having n user position errors when estimating N user positions.
- P_{no_error}: Probability of correctly estimating a user position

A user position error can be produced by different causes, each of them defined in annex F.1.1. Therefore, we can modify the previous mathematical expression in order to adapt it to the previous defined probabilities.

$$P\left(ne\ X\ Npr\right) = \binom{N}{n} \cdot P_{no\_error}{}^{N-n} \cdot P_{eX}{}^{n} \tag{F-12}$$

Where:

- P(ne X Npr): Probability of having n user position errors caused by X, when estimating N user positions.
- P_{eX}: Probability of a user position estimation error caused by X.

Therefore, for all the cases, we need to determine the possibility of correctly estimating a user position and to determine the probability of wrongly estimating a user position due to a given cause in order to express all the probabilities defined in annex F.1.1.

First, the probability of a correct user position estimation is equal to the probability that all the satellite position predictions are correct.

$$P_{no\_error} = P_{sat\_no\_error}^{\ M} \tag{F-13}$$

Where:

- $P_{sat\_no\_error}$: Probability of correctly predicting the satellite position.
- M: Average number of satellites.

Besides, the probability of well predicting a satellite position is equal to the probability of well predicting all its Keplerian parameters. And the probability of well predicting a Keplerian parameter depends on the number of erroneous bits accepted by each Keplerian parameter.

$$P_{sat\_no\_error} = \left(1 - P_{error\ \sqrt{A}}\right) \cdot \left(1 - P_{error\ e}\right) \cdot \left(1 - P_{error\ \omega}\right) \cdot \ldots \cdot \left(1 - P_{error\ i}\right) \tag{F-14}$$

Customizing this expression for different bad Keplerian parameters predictions, we can obtain different results.

Finally, expression (F-12) is customized in order to describe all the probabilities defined in annex F.1.1. We begin with the simple probabilities and continue with the complicated ones using the simplifications and the results of the first probabilities.

First, the event of not having any error before applying the RAIM during all the time period is simply a binomial random variable where all cases are the same:

$$P(0e\ bR\ Npr) = P_{no\_error}^{\ N} = P_{sat\_no\_error}^{\ M \cdot N} \tag{F-15}$$

Second, we can regroup all the probabilities of having one user position error before the RAIM processing in one compact expression. Note that this expression is the sum of having one user position error caused by 1 satellite position error or more.

$$\begin{aligned} P(1e\ bR\ Npr) = & \sum_{n=2}^{M} P(1e\_nse\ bR\ Npr) + P(1e\_1se\ bR\ Npr) \cdot P(Rd\_1e | 1e) + \\ & + P(1e\_1se\ bR\ Npr) \cdot P(Rd\_0e | 1e) \end{aligned} \tag{F-16}$$

$$P(1e\ bR\ Npr) = \sum_{n=1}^{M} P(1e\_nse\ bR\ Npr) \tag{F-17}$$

Each individual term of equation (F-17) can be expressed as:

$$P(1e\_1se\ bR\ Npr) = \binom{N}{1} \cdot P_{no\_error}^{\ N-1} \cdot P_{error\_1\_sat} \tag{F-18}$$

$$P_{error\_n\_sat} = \binom{M}{n} \cdot P_{sat\_no\_error}^{\ M-n} \cdot \left(1 - P_{sat\_no\_error}\right)^{n} \tag{F-19}$$

$$P(1e\_1se\ bR\ Npr) = N \cdot M \cdot P_{sat\_no\_error}^{\ N \cdot M - 1} \cdot \left(1 - P_{sat\_no\_error}\right) \tag{F-20}$$

Therefore, the total probability of expression (F-17) is:

$$P(1e\ bR\ Npr) = N \cdot \sum_{n=1}^{M} \binom{M}{n} \cdot P_{sat\_no\_error}{}^{N \cdot M - n} \cdot \left(1 - P_{sat\_no\_error}\right)^{n} \tag{F-21}$$

Once the probability of obtaining one user position estimation error before the RAIM processing is given, this study can follow the same analysis to find the remaining probabilities. First, we give the probability of having either 0 or 1 user position error after the RAIM processing when k user positions errors caused each one by one satellite position error occurred before the RAIM processing. We call this probability $P_{2..Ne\_1sat}$.

$$P_{2...Ne\_1sat} = \sum_{k=2}^{N} P(ke\_1se\ bR\ Npr) \cdot \left[P(Rd\_ke|ke) + P(Rd\_(k-1)e|ke)\right] \tag{F-22}$$

We first simplify the probability expression by customizing the probability with k = N.

$$P_{Ne\_1sat} = \binom{N}{N} \cdot \binom{M}{1}^{N} \cdot P_{sat\_no\_error}{}^{N \cdot M - N} \cdot \left(1 - P_{sat\_no\_error}\right)^{N} \cdot \left[P_{dR}{}^{N} + N \cdot P_{dR}{}^{N-1} \cdot \left(1 - P_{dR}\right)\right] \tag{F-23}$$

$$P_{2..Ne\_1sat} = \sum_{i=1}^{N} \binom{N}{i} \cdot M^{i} \cdot P_{sat\_no\_error}{}^{N \cdot M - i} \cdot \left(1 - P_{sat\_no\_error}\right)^{i} \cdot \left[P_{dR}{}^{i} + i \cdot P_{dR}{}^{i-1} \cdot \left(1 - P_{dR}\right)\right] \tag{F-24}$$

The same development can be made but collecting the event where at least two user position errors occur, where one user position error is caused by more than one wrong satellite position, whereas the other user position errors are caused by only one wrong satellite position. This probability has been previously called P(>2e Npr). Nevertheless, in order to simplify the process, the probability of having 2 user position errors when 1 user position error is caused by n satellite errors is expressed first.

$$P_{2e\_nsat} = \binom{2}{1} \cdot \binom{N}{2} \cdot \binom{M}{1} \cdot \binom{M}{n} P_{sat\_no\_error}{}^{N \cdot M - 1 - n} \cdot \left(1 - P_{sat\_no\_error}\right)^{n+1} \cdot P_{dR} \tag{F-25}$$

The main difference between this expression and the previous probability equations is the first combinatorial number. This number is necessary because we have to distinguish the user position errors caused by at least two wrong satellite positions from the user position errors caused by only one wrong satellite position. Indeed, it is not the same case to have the first user position erroneous due to two satellite position errors and the second user position erroneous due to one satellite position error compared to the case where we have the first user position erroneous due to one satellite position error and the second user position erroneous due to two satellite position errors.

Therefore, if we regroup all the previous probability expressions for the cases where 2 user position errors occur:

$$P(2e\ Npr) = 2 \cdot \binom{N}{2} \cdot P_{dR} \cdot M \cdot \sum_{n=2}^{M} \binom{M}{n} P_{sat\_no\_error}{}^{N \cdot M - 1 - n} \cdot \left(1 - P_{sat\_no\_error}\right)^{n+1} \tag{F-26}$$

The expression for 3 user position errors is thus.

$$P(3e\ Npr) = 3 \cdot \binom{N}{3} \cdot P_{dR}{}^{2} \cdot M^{2} \cdot \sum_{n=2}^{M} \binom{M}{n} P_{sat\_no\_error}{}^{N \cdot M - 2 - n} \cdot \left(1 - P_{sat\_no\_error}\right)^{n+2} \tag{F-27}$$

Finally, the expression of the probabilities of all the cases with more than one user position error, where one user position error is caused by at least two satellite position errors whereas the other user position errors are caused by only one satellite position error is:

$$P(>2e\ Npr) = \sum_{i=2}^{N} i \cdot \binom{N}{i} \cdot P_{dR}{}^{i-1} \cdot M^{i-1} \cdot f(M,i) \quad \text{(F-28)}$$

$$f(M,i) = \sum_{n=2}^{M} \binom{M}{n} P_{sat\_no\_error}{}^{N\cdot M-n-i+1} \cdot \left(1 - P_{sat\_no\_error}\right)^{n+i-1} \quad \text{(F-29)}$$

Finally, with this last equation, all the probabilities defined in annex F.1.1 have been given. Therefore, the sum of expressions (F-15), (F-21), (F-24) and (F-28) are equivalent to the left part of equation (F-1).

$$P(0e\ bR|Npr) + P(1e\ bR|Npr) + P_{2e\ldots Ne\_1sat} + P(>2e|Npr) \geq 99\% \quad \text{(F-30)}$$

F.2. Relationship between the binary domain and the decimal domain

In this section, the theoretical calculations conducted to establish the relationship between a number expressed in binary format and the same number expressed in decimal format are presented. More specifically, these calculations search for the first and second percentages associated to a decimal number in order to obtain the prediction of a given number of bits. See section 4.2.5 for a definition of the first and second percentage.

First of all, it has to be reminded that if the bit in position n is not well predicted by a physical prediction method, see section 4.2.2 for a definition of a physical prediction method, the method cannot guarantee the correct prediction of any bit value less significant than the bit in position n.

The theoretical analysis conducted to find the relationship between the decimal and the binary precision can be divided into four main parts. The first part searches for the worst prediction able to correctly predict a given bit. The second part searches for the best prediction which does not correctly predict a given bit. The third part calculates from each prediction the decimal difference expressed in binary terms between the predictions and the real values. These differences are called distances and there are several values to predict having the worst and best prediction distances. Note that the distance of the worst predictions is the maximal acceptable decimal distance and that the distance of the best predictions is the minimal acceptable decimal distance. Finally, the fourth part calculates from the worst prediction distance the first percentage and from the best prediction distance the second percentage described in section 4.2.5.

In order to accomplish the first and second parts, a preliminary step has to be made. This preliminary step consists in sorting all the range of values of the field to predict into two groups, one group gathering the values having the n^{th} bit equal to 0, and the other group gathering the values having the n^{th} bit equal to 1. Each set is called bit-value-set. From these sets, the search of the worst and the best prediction is simplified.

In order to find the worst prediction able to correctly predict a given bit, we search for the two values belonging to the same bit-value-set which are separated by the largest distance. Therefore, since there is no other pair of values farther apart, each value represents the worst prediction when the other value is the value to be predicted.

In order to find the best prediction unable to correctly predict a given bit, we search for the two values belonging to different bit-value-sets which are separated by the smallest distance. Therefore, since there is no other closest pair of values, each value represents the best prediction when the other value is the value to be predicted.

An example of the worst and best predictions of the first bit of a binary value represented by 5 bits is presented in Table F-1. In this case, the bit-value-set of the first bit equal to 0 consists of the binary numbers between 00000 and 01111, and the bit-value-set of the first bit equal to 1 consists of the binary numbers between 10000 and 11111.

Value of the First bit	**0**	**1**
Real value to predict associated to the best prediction	01111	10000
Best prediction erroneously predicting the associated value	10000	01111
Real value to predict associated to the worst prediction	00000/ 01111	10000/ 11111
Worst prediction correctly predicting the associated value	01111 / 00000	11111 / 10000

Table F-1: First bit best and worst predictions for a 5 bit number

Note that since the worst prediction is searched between pairs of numbers belonging to the same bit-value-set, the worst prediction and the real value to be predicted are interchangeable.

The table presenting the binary values of the worst and best prediction distances is given below.

Value of the First bit value	**0**	**1**
Best prediction distance	00001	00001
Worst prediction distance	01111	01111

Table F-2: First bit best and worst prediction distances for a 5 bits number

One important remark about Table F-2 is that the best and worst distances are equal for both bit-value-sets. Therefore, the study can be reduced to the analysis of only one of the bit-value-sets.

At this point, we should use the worst and best prediction distances expressed in decimal format in order to calculate the percentages with respect to the maximal range value, i.e. the first and second percentages. However, the conversion process of these binary distances into the decimal format is complex. The accuracy in decimal format for a value converted from a binary format is determined by this LSB of the value. This means that the worst and best predictions distances have an accuracy which depends on this bit, and this limit of accuracy is harmful for our calculation since it limits the precision of our percentages to the LSB accuracy. Therefore, in order to remove this limit of accuracy and in order to have very accurate decimal distance values, we have to artificially extend the accuracy of the binary format of the distances. This extension is easily achieved if we consider a decimal number as a binary number with an infinite number of bits. This means that if we insert a sufficient number of extra bits of lower weight than the LSB of the original binary format of the binary value to the binary value, we are approaching the binary value to the ideal binary

representation of a decimal value. Finally, the values of these extra bits are smartly determined in order to adapt the new precision to our requirements. An example of this process is given next when 10 low weight bits are inserted.

The insertion of 10 low weight bits to the initial number of bits of the binary field in order to modify the worst and best prediction distances and the maximal range value is explained next. The values of the bits inserted to the maximal range value are equal to 1. The values of the bits inserted to the worst distance are 1 for the bit with more weight and 0 for the others. The reason is that the worst decimal maximal distance with the new accuracy is the sum of two terms. The first term is the original binary worst prediction distance expressed in decimal format. The second term is half the decimal value of the LSB of the original binary format of the distance, where the original binary format is the binary format before inserting the 10 extra bits. And this half decimal value of the original binary format LSB is represented by the 10 additional bits where the first bit is equal to 1 and the last 9 bits are equal to 0. Finally, the same analysis can be made for the best prediction distance with new accuracy. In this case, we have to subtract half the decimal value of the LSB of the original binary format from the distance. This means that first we have to change the last bit of the original binary format of the best prediction distance to 0, and second we have to add the 10 extra bits with the same values as the values of the bits of the worst prediction distance.

Finally, the formula of the prediction relative distance is given below.

$$\text{Distance precision (\%)} = 100 \cdot \frac{Dist}{Max\ Range\ Value} \tag{F-31}$$

Where:

- Dist: Prediction distance expressed in decimal format
- Max Range Value: Maximal range value expressed in decimal format

Therefore, applying equation (F-31) and expanding the value precision with 10 bits as explained in the previous paragraph, the results obtained for the prediction of the first bit are presented below.

Prediction accuracy	
Maximal Precision Distance	*Minimal Precision Distance*
48.4375 %	1.5625 %

Table F-3: First bit prediction accuracy for a 5 bit number

Prediction accuracy	
Maximal Precision Distance	*Minimal Precision Distance*
49.8 %	0.2 %

Table F-4: First bit prediction accuracy for an 8 bit number

Table F-3 shows that if we want the physical prediction method to be able to predict the first bit of a 5 bit number, the prediction has to be at a distance from the real value lower than ≈50% of the maximal range value. Nevertheless, this table shows that despite obtaining an almost perfect prediction, such as a distance between the prediction and the real value of 2% of the maximal range value, the prediction of the first bit cannot be guaranteed. The same

conclusions can be made from Table F-4. Moreover, from both tables, we can conclude that the number of bits has little influence on the final results.

Another example is given before presenting a general equation of the evolutions of the maximal and minimal precision distances. The example searches for the accuracy necessary to predict the second bit. In this case, there are four different bit-value-sets because we assume that if the second bit is to be predicted, the first bit has to be correctly predicted. Therefore, it is impossible for a physical prediction method to predict a decimal value expressed as 01XXX when the first bit is known to be 1. Consequently, the four bit-value-sets for a 5 bit number are 00XX, 01XX, 10XX, 11XX. And from these sets, the worst and best predictions and their associated distances are found. The distances are shown below in Table F-5.

First bit value	**0**		**1**	
Second bit value	**0**	**1**	**0**	**1**
Best prediction distance	00001	00001	00001	00001
Worst prediction distance	00111	00111	00111	00111

Table F-5: Second bit best and worst prediction distances for a 5 bit number

And the distance precision for a 5 bit number, which is valid for a value expressed by any number of bits, is shown below on Table F-6.

Prediction accuracy	
Maximal Precision Distance	*Minimal Precision Distance*
23.4375 %	1.5625 %

Table F-6: Second bit prediction accuracy for a 5 bit number

Therefore, the minimal precision distance (MnPD) necessary to guarantee a bit prediction remains constant for all the bits of the field to predict. Nevertheless, the maximal precision distance (MxPD) necessary to try to predict a bit depends on the bit position inside the field. The formula of the MxPD is given below.

$$MxPD \approx \frac{100}{2^n} - 2^n \cdot MnPD \ (\%) \qquad \text{(F-32)}$$

Where:

- n = position of the bit to be predicted

F.3. Probabilities a priori of the ephemeris data

In 0, we have presented a modification of the traditional decoding Viterbi algorithm. This modification employed the a priori probabilities of the bits belonging to the ephemeris data. In this section, the method used to calculate the a priori probabilities is presented. These a priori probabilities have been calculated from the history of the ephemeris data which encloses the ephemeris data set broadcasted between 2004 and 2007. The data is recovered from the webpage http://igscb.jpl.nasa.gov.

We have obtained the probability of the n^{th} bit of the i^{th} Keplerian parameter by dividing the number of i^{th} Keplerian parameters having the n^{th} bit equal to 1 by the total number of i^{th} Keplerian parameters. Moreover, during the bit probabilities calculation, we remarked that some Keplerian parameters such as the M_0, C_{uc} or C_{rc} could be divided into two cases. In fact, when these Keplerian parameter bits probabilities were directly calculated, the results were quite disappointing with probabilities about 50%. However, when we separated first the negative and positive Keplerian parameter values, the probabilities, or at least the probabilities of the more significant bits were more satisfying. These probabilities had values about 80%. Therefore, we have separated the Keplerian parameters encoded 2's complement into positive value probabilities and negative value probabilities.

Finally, each satellite has its own singular orbit, as has been shown in section 4.3.2 which means that each satellite has to have its own ephemeris bit probabilities. These probabilities can be recalculated along the time since more samples or broadcasted ephemeris will be available. Therefore, the receivers can store the a priori bit probabilities and receive their actualizations by external platforms such as mobile telephone stations.

One example table summarizing the a priori bit probabilities is shown at the end of this annex.

F.4. Influence of an erroneous bit inside a Keplerian parameter on the satellite position

The combination of two of the main assumptions of section 4.2.4.2 specifies that the binary prediction error of only bit of a Keplerian parameter bit results into an erroneous satellite position prediction. Moreover, a third assumption specified that an erroneous satellite position prediction leads to a user position prediction error. Therefore, in order to validate these assumptions, the influence of an erroneous bit inside a Keplerian parameter on the satellite position is inspected.

The method used for analyzing the influence of the Keplerian parameters bits on the satellite position consists in inspecting the satellite position error introduced by the variation of the Keplerian parameters bits. The method first searches for the error introduced by the variation of the bits of each individual Keplerian parameter and second, it searches for the error introduced by the variation of the bits of two simultaneous Keplerian parameters.

Moreover, since the weight of the bit certainly determines the influence of the bit on the satellite orbit position, the method searches the bit of lowest weight which changes the satellite position farther than a determined threshold distance. Therefore, this method begins the inspection by the LSB of the Keplerian parameter and finishes, if necessary, by the MSB. Moreover, since we use physical methods to predict the satellite position, the wrong prediction of a bit of a Keplerian parameter implies the wrong prediction of this bit and all the bits having less weight inside the Keplerian parameter. Therefore, the method considers a wrong bit prediction as a wrong prediction of this bit and its previous LSB.

More specifically, the method searches the satellite position error in the worst scenario, which is the situation where a Keplerian parameter bit prediction causes the largest change on the Keplerian parameter value. The worst scenario for a positive Keplerian parameter value when its n^{th} bit is to be tested is simulated by creating a new ephemeris data set where the Keplerian parameter n^{th} bit value and the bits having a lower weight are set to 0. This new ephemeris set is called the reference ephemeris data set. From the orbit defined by the reference ephemeris set, we can calculate the maximal error distance introduced by the wrong prediction of the n^{th} bit of this Keplerian parameter. This calculation is done by measuring the distance between

the orbit defined by the reference ephemeris data set and the orbits defined by the different ephemeris data sets which are equal to the reference ephemeris data set but with different values for the n^{th} bit and its bits of lower weight of the Keplerian parameter. A similar process is conducted for a negative Keplerian parameter value.

Table F-7 shows the bit of more weight of a Keplerian parameter which causes a satellite position error larger than a determined threshold. This table has been calculated using 400 different ephemeris sets and varying the value of only one Keplerian parameter.

		Keplerian Parameters Bit position														
		$\sqrt{A}$	**e**	**ω**	**IDOT**	**Ω'**	**Δn**	**M0**	**cuc**	**cus**	**crc**	**crs**	**cic**	**cis**	**Ω_0**	**i**
Dist. (m)	**0.1**	31	28	31	13	23	15	31	15	15	14	14	15	15	31	31
	1	28	25	28	11	21	13	28	12	12	11	11	12	12	28	28
Num. Bits		32	32	32	14	24	16	32	16	16	16	16	16	16	32	32

Table F-7: First keplerian parameter bit position allowed to be wrong predicted when one and only one keplerian parameter is incorrectly predicted

From Table F-7, it can be observed that the bits of some Keplerian parameters have a larger influence on the satellite position than the bits of other Keplerian parameters. For example, the orbit semi-major axis accepts only a variation of its last 4 bits before modifying the satellite position by more than 1m, whereas the eccentricity allows a variation of its last 7 bits.

Table F-8 shows the bit of more weight which causes a satellite position error larger than a determined threshold when the bits of two different Keplerian parameters are modified at the same time. This table has been calculated using 400 different ephemeris sets.

		Keplerian Parameters		
		$\sqrt{A}$	**E**	**ω**
Distance (m)	**1**	32	---	32
	1	32	32	---
	1	---	32	32
Number of bits		32	32	32

Table F-8: First Keplerian parameter bit position allowed to be wrong predicted when two Keplerian parameters are incorrectly predicted

From Table F-8, it can be observed that the variation of the LSB of two different Keplerian parameters at the same time causes a satellite position error larger than 1 meter. Therefore, two or more Keplerian parameters cannot be incorrectly predicted at the same time if the GPS receiver seeks to implement a prediction method yielding a satellite position error lower than 1 meter.

One possible combination of wrong predicted bits of all the Keplerian parameters, except the t_{oe}, which cause a satellite position error larger than 600 meters is presented below.

		Keplerian Parameters Bit position														
		$\sqrt{A}$	**e**	**ω**	**IDOT**	**Ω'**	**Δn**	**M0**	**cuc**	**cus**	**crc**	**crs**	**cic**	**cis**	**Ω_0**	**i**
Dist. (m)	**600**	26	26	26	8	18	9	25	9	9	9	9	9	9	25	25
Num. Bits		32	32	32	14	24	16	32	16	16	16	16	16	16	32	32

Table F-9: Possible combination of wrong predicted bits of all the Keplerian parameters which cause a satellite position error larger than 600 meters

Finally, we can conclude that the specifications of the case of study are only fulfilled when one or more bits are erroneously predicted for one Keplerian parameter. The erroneous prediction of bits belonging to different Keplerian parameters leads to a satellite position error larger than 1 meter and thus it is not accepted by our case of study. Moreover, it has been shown that in order to use the RAIM function, we cannot predict the LSBs of the Keplerian parameters since the satellite position error will be smaller than 600 meters and thus it will not be detected.

SAT 01	Probabilities of having bits equal to 1																
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
sqrtA	1	0	1	0	0	0	0	1	0	0	0	0	1	0,871	0,129	0,908	0,476
E	0	0	0	0	0	0	1	0,937	0,18	0,399	0,641	0,429	0,532	0,509	0,519	0,494	0,504
Om	1	0	1	1	0,613	0,487	0,579	0,461	0,412	0,614	0,522	0,476	0,49	0,501	0,509	0,501	0,503
Id_first	0.654																
Id_pos	---	0	0	0,167	0,376	0,446	0,461	0,486	0,485	0,492	0,5	0,506	0,497	0,501	---	---	---
Id_neg	---	1	1	0,984	0,736	0,594	0,55	0,526	0,499	0,515	0,49	0,496	0,488	0,496	---	---	---
Od	1	1	1	1	1	1	1	1	1	0	1	0,005	0,775	0,457	0,502	0,498	0,501
Dn	0	0	1	0,004	0,855	0,383	0,493	0,504	0,499	0,501	0,508	0,499	0,497	0,502	0,502	0,504	---
M_first	0,4995																
M_pos	---	0,499	0,501	0,499	0,498	0,501	0,497	0,498	0,503	0,496	0,499	0,499	0,503	0,508	0,506	0,499	0,496
M_neg	---	0,497	0,499	0,499	0,499	0,499	0,501	0,5	0,503	0,495	0,502	0,504	0,505	0,491	0,499	0,5	0,498
Cuc_first	0,4438																
Cuc_pos	---	0	0	0	0,28	0,407	0,461	0,472	0,486	0,488	0,492	0,491	0,501	0,51	0,499	0,493	---
Cuc_neg	---	1	1	0,971	0,595	0,539	0,53	0,518	0,518	0,502	0,507	0,496	0,512	0,503	0,503	0,497	---
Cus_first	0,9721																
Cus_pos	---	0	0,001	0,417	0,42	0,461	0,493	0,496	0,501	0,5	0,503	0,5	0,5	0,493	0,501	0,493	---
Cus_neg	---	1	1	1	1	0,9868	0,8785	0,6525	0,6073	0,5292	0,5678	0,5207	0,4981	0,5	0,4802	0,4463	---
Crc	0	0	0,5574	0,483	0,5284	0,503	0,4993	0,4965	0,4951	0,4965	0,498	0,4998	0,4996	0,4936	0,4981	0,5044	---
Crs_first	0,4439																
Crs_pos	---	0	0	0,0119	0,3441	0,4383	0,4726	0,4719	0,4909	0,4925	0,501	0,4891	0,5008	0,5047	0,5005	0,4999	---
Crs_neg	---	1	1	0,9094	0,5769	0,551	0,528	0,5102	0,516	0,5037	0,4955	0,5014	0,4938	0,5041	0,5007	0,4969	---
Cic_first	0,4869																
Cic_pos	---	0	0	0	0	0	0	0	0	0,2113	0,4593	0,4846	0,4883	0,498	0,488	0,4985	---
Cic_neg	---	1	1	1	1	1	1	1	1	0,8487	0,4845	0,5306	0,5127	0,5124	0,5072	0,5156	---
Cis_first	0,5268																
Cis_pos	---	0	0	0	0	0	0	0	0,001	0,1924	0,34	0,4313	0,4624	0,4878	0,4911	0,5053	---
Cis_neg	---	1	1	1	1	1	1	1	1,0006	0,78	0,6322	0,5533	0,5301	0,5105	0,5039	0,5045	---
O_first	0,493																
O_pos		0,4921	0,5066	0,5066	0,4908	0,4916	0,4983	0,5039	0,5805	0,498	0,5007	0,5004	0,5002	0,5007	0,5018	0,5047	0,5039
O_neg	---	0,4937	0,4943	0,5086	0,5226	0,5086	0,5024	0,492	0,4127	0,4884	0,5052	0,5018	0,5007	0,5004	0,5007	0,5013	0,5005
i	0	0	1	0	1	0	0	0	0	0,5328	0,3511	0,5336	0,6742	0,5133	0,5154	0,4863	0,4955

Annex G. Temporal methods of the binary prediction of the ephemeris data

In this annex, the temporal methods used to predict the bits of the ephemeris data are presented in detail.

G.1. Method allowing the use of the DIFF difference results

In this section, a method which is able to obtain the DIFF difference as result of the simplification process instead of the XOR difference is presented. Remember that the simplification process using the almanac data or the long term prediction ephemeris data has as result of the prediction the XOR difference, where the XOR difference determines the bit of most weight which cannot be perfectly predicted. Therefore, since the XOR difference is worse than the DIFF difference as verified in section 4.3.1.1.3 and 4.3.1.1.4, we are interested in applying the simplification process and in obtaining the DIFF difference as its result.

The proposed method consists in first making 3 predictions of the future ephemeris data set instead of only one, and second in choosing the correct prediction through the application of a determined criterion.

The explanation of the proposed is divided in 3 parts. First, the problems with the DIFF difference are identified and analyzed. Second the 3 predictions method used to solve the DIFF difference problem is presented. Third and last, the criterion used to select the correct prediction is explained.

G.1.1. DIFF difference problem

The DIFF difference is calculated by first subtracting the decimal reference value to the decimal broadcasted ephemeris data set. Second, we transform the absolute value of the resulting decimal difference into binary format. Therefore, the DIFF difference does not represent the bits of the reference value having the same value as the broadcasted Keplerian parameters bits. In fact, the DIFF difference represents the value missing from the reference value to reach the value of the broadcasted Keplerian parameter converted into binary format. From now on, we call DIFF difference value the binary value to be added or subtracted from the reference value in order to obtain the value of the broadcasted Keplerian parameter. Therefore, since the DIFF difference represents a binary format of the value to be added or subtracted, we cannot guarantee the prediction of any bit from the DIFF difference value. The reason is that even the binary addition of the LSB bit to a binary number can modify all the bits of the number as shown in section 4.3.1.1.2. However, if we are able to determine in advance the modification of the predicted bits by the addition or the subtraction of the DIFF difference value, we could make a prediction which will obtain as a result of the simplification process a XOR difference equal to the DIFF difference.

To sum up, in this section, we analyze the structure of the DIFF difference value in order to determine the effects of the binary addition and subtraction. Therefore, knowing these effects, we obtain as result of the simplification process the DIFF difference instead of the XOR difference.

The DIFF difference value can be divided into two parts. The first part of this difference is formed by the DIFF difference constant bits, or, in other words, from the MSB bit until the last bit with probability of error equal to 0. Remember that the probability of error of a bit, as defined in section 4.3.1.1.3, determines the percentage of times that the bit of the reference value has been the bit of most weight not equal to the bit of the broadcasted Keplerian parameter which is at the same position. Therefore, in the DIFF difference case, if the probability of error of a bit is equal to 0, this means that this bit has a constant value equal to 0 in the DIFF difference value.

The second part of this difference is formed by the non-constant or unknown bits, or, in other words, from the first bit with a probability of error not equal to 0 to the LSB bit. Therefore, since the constant bits of this difference are equal to 0, the decimal DIFF difference value is completely determined by these non-constant and unknown bits and thus it is not fixed.

The DIFF difference value can be defined as:

$$\text{DIFF difference value: } \overbrace{0\ldots0}^{Df1}\overbrace{x\ldots x}^{Df2} \tag{G-1}$$

Where:

- Df1: First part of the DIFF difference – Constant bits
- Df2: Second part of the DIFF difference – Not constant or unknown bits

From the structure shown in equation (G-1), we can draw some conclusions. First, the first part of the DIFF difference does not have any influence on the addition between DIFF difference value and the reference value. Second, the influence of the unknown bits of the second part of the DIFF difference on this addition is unpredictable. Therefore, since this influence is unpredictable, all the bits of the addition between the DIFF difference value and the reference value are also unpredictable. And this means that the DIFF difference value cannot be used as indication of which bits can be predicted by the simplification process. An example is given below.

Reference Value (RV)	DIFF Difference	Df1	Df2	RV + Df1 + Df2
101101	0000xx	**000**00	00000**xx**	xxxxxx

Table G-1: DIFF difference numerical example

Where:

- x: Represent the value of an unknown bit

In order to be able to exploit the DIFF difference value as the vector indicating the bits which can be predicted by the simplification process, we have to find the influence of Df2 on the MSB bits of the addition of the reference value with the DIFF difference value. Remember that the DIFF difference value allows the prediction of more bits than the XOR difference value.

In order to do so, we calculate all the possible Df2 contributions on the binary addition between the DIFF difference value and the reference value so that we can find a predictable pattern of the contribution on the MSB bits of the addition. The contributions of Df2 are divided into their contribution on the bits of the binary addition which are at the same position than the Df2 bits and their contribution on the bits of the binary addition which are at the same position than the Df1 bits. Table G-2 shows the contribution of different positive values

of the unknown bits. The vertical line separates the bits belonging to Df1 from the bits belonging to Df2.

Reference value	0..0|111	0..0|011	0..0|111	0..0|001
Positive DIFF difference (Df1 | Df2)	0..0|011	0..0|100	0..0|111	0..0|010
Df2 (Denoted as (2))	|011	|100	|111	|010
Reference value bits at the same position as Df2 (Denoted as (1))	|111	|011	|111	|001
(1)+(2) = Contribution of Df2 on the bits of the binary addition which are at the same position than the Df2 bits	|010	|111	|110	|011
(1)+(2) = Contribution of Df2 on the bits of the binary addition which are at the same position than the Df1 bits	0..01|	0..00|	0..01|	0..00|

Table G-2: Df2 contribution of a positive DIFF difference value on the binary addition between the DIFF difference and the reference value

It can be observed that whereas the contribution of Df2 bits on the bits of the binary addition situated at the same position as the Df2 bits is unpredictable, the Df2 contribution on the bits of the binary addition situated at the same position as the Df1 bits is easily predictable. In fact, there are only two types of contribution. The Df2 bits do not contribute at all on the bits situated at the same position as the Df1 bits, or the Df2 bits add 1 bit to the LSB bit of the bits situated at the same position as the Df1 bits.

Once a Df2 contribution of a positive DIFF difference value is analyzed, we continue with the Df2 contribution of a negative DIFF difference value. The contribution of a negative value is shown in complement-2 format. Table G-3 shows the examples.

Reference value	0..0|111	0..0|011	0..0|111	0..0|011
Negative DIFF difference (Df1 | Df2)	1..1|100	1..1|010	1..1|011	1..1|001
Df2 (Denoted as (2))	|100	|010	|011	|001
Reference value bits at the same position as Df2 (Denoted as (1))	|111	|011	|111	|011
(1)+(2) = Contribution of Df2 on the bits of the binary addition which are at the same position than the Df2 bits	|011	|101	|010	|100
(1)+(2) = Contribution of Df2 on the bits of the binary addition which are at the same position than the Df1 bits	0..0|	1..1|	0..0|	1..1|

Table G-3: Df2 contribution of a negative DIFF difference value on the binary addition between the DIFF difference and the reference value

The Df2 contribution of a negative DIFF difference value is also easily predicted. The contribution on the bits of the binary addition situated at the same position as the Df2 bits is unpredictable and there are two types of the Df2 contributions on the bits of the binary addition situated at the same position as the Df1 bits. The Df2 bits do not contribute at all on the bits situated at the same position as the Df1 bits, or the Df2 bits add 1 bit to all the bits situated at the same position as the Df1 bits.

Once the Df2 contributions of a negative and a positive DIFF difference value have been analyzed, we know all the possible variations of the bits of the binary addition between the reference value and the DIFF difference value which are situated at the same position as the Df1 bits. Therefore, we can implement a method allowing the exploitation of the DIFF difference value as the vector indicating the bits which can be predicted. This method consists in generating a new reference value such that when it is used in the simplification process yields a XOR difference value equal to the DIFF difference value obtained with the original reference value.

G.1.2. DIFF difference method

The method allowing the creation of the new reference value commented at the end of the previous section consists in 3 steps. The first step identifies the different parts of the DIFF difference value, Df1 and Df2. The second step creates 3 possible new reference values which cover all the possible Df2 contributions on the bits of the binary addition between reference value and the DIFF difference which are situated at the same position as the Df1 bits. Finally, the third step applies a criterion in order to determine among the 3 new reference values the one that correctly predicts the broadcasted ephemeris data set. Note that this prediction does not predict the bits situated at the same position than the Df2 bits.

In this section, only step 2 is presented since step 1 is easily accomplished and step 3 is presented in annex G.1.3. Step 2 describes the 3 new reference values and the justification of why these predictions cover all the possible Df2 contributions. The 3 new reference values are equal to the binary addition of the original reference value with the Df2 bits of either a positive or negative DIFF difference value. Each new reference value represents a different contribution of the Df2 bits on the binary addition. Remember that the bits situated at the same position as the Df2 bits are always unknown and thus, they are never predicted.

The 3 new reference values are given next.

1. Normal prediction: This reference value assumes that the Df2 contribution is 0 for either a positive or a negative DIFF difference value.
2. Addition prediction: This reference value assumes that the Df2 contribution is equal to the addition of the LSB of the Df1 part. This contribution can be brought by a positive DIFF difference value.

   Example: mod(010011xxx + 000001xxx, 2) = 010110xxx
3. Subtraction prediction: This prediction reference value assumes that the Df2 contribution is equal to the sum of a vector of ones on the bits of the binary addition which are situated at the same position as the Df1 bits. This contribution can be brought by a negative DIFF difference value.

   Example: mod(010011xxx + 111111xxx, 2) = 010010xxx

It can be observed that these 3 predictions cover all the possible contributions of the Df2 part of the DIFF difference value. Therefore, it can be guaranteed that one of them correctly predicts the bits of the broadcasted ephemeris data set which are situated at the same position as the Df1 bits.

G.1.3. Selecting criterion and algorithm

The selection consists in choosing as the correct new reference value, the new reference value formed by the bits which pass the channel code verification after the reorganization of the bits in code words. The channel code verification is defined in annex E.1.2. Besides, note that this verification has to use the bits of the new reference value which cannot be predicted, the bits situated at the same position as the Df2 bits. The values of these non-predictable bits are determined by the typical demodulation process.

In order to optimize the selection process, the receiver can test the new reference values using the following method. First, the receiver correlates the predicted bits with the values of these predicted bits but obtained by the typical demodulation process. Second and last, since there are 3 new reference values for each predicted Keplerian parameter, the code words of the message are constructed by first using the predicted Keplerian parameters providing the greatest correlation peaks. If the verification of the channel word fails for any of the constructed code words, the method uses the next predicted Keplerian parameter having the greatest correlation peak in order to construct a new code word. Therefore, the first group of Keplerian parameters trying to pass the channel code verification is the group formed by the new reference value of each Keplerian parameter obtaining the greater correlation peak.

G.2. Process of simplification using the almanac data

The process of simplification using the almanac data consists in first choosing the proper almanac data set to be compared to the ephemeris data set, because the almanac data sets are broadcasted about every day and the ephemeris data sets are broadcasted about each 2 hours. Once the selection has been done, the process continues by calculating the XOR and DIFF difference, and the bit error probabilities.

The association between almanac data and ephemeris data is not trivial, because there are more ephemeris data samples than almanac data samples, although the almanac data samples have a longer validity.

- 1 ALMA file every day (1 sample/day) – 6 days of validity
- 1 EPHEMERIS file every 2 hours (1 sample/2h) – 4h of validity

Therefore, we subtract the ephemeris data of day $(n+1)^{th}$ to the almanac data transmitted in day n^{th}, because the almanac data validity is 6 days. Consequently, the orbit represented by the n^{th} almanac data corresponds to the orbits represented by the $(n+1)^{th}$ ephemeris data. The resulting difference signal, y, can be modeled as:

$$y[x] = alma(\lfloor x/12 \rfloor) - ephemeris(x+12) \qquad \text{(G-2)}$$

Where

- y: XOR or DIFF difference between the n^{th} day almanac data set and the $(n+1)^{th}$ day ephemeris data sets.

- x: Index of 2-hour delta set
- alma(t): Almanac data of day t
- ephemeris(t): Ephemeris data of t set of 2 hours
- $\lfloor t \rfloor$: Lower integer part of number t

Nevertheless, whereas the ephemeris data are normally broadcasted at the same time every day, the transmission time of the almanac data varies with each day. Moreover, the orbit described by either the ephemeris data or by the almanacs data has an interval of validity such that at their middle point, the described orbit is closer to the real orbit. Therefore, instead of the calculating the difference signal as has been specified in (G-2), we can build another difference signal where each ephemeris data is subtracted to the almanac with interval center closest to the interval center of the ephemeris data. The optimal difference signal can be modeled as:

$$y'[x] = alma(f[x]) - ephemeris(x+12) \qquad \text{(G-3)}$$

Where:

- y'[x]: XOR or DIFF difference between the ephemeris data set n and the almanac data set m with interval center the closest to the ephemeris data set n interval center.
- f[x]: Function which searches for the almanac data set with interval center the closest to the ephemeris data set (x+12) interval center

This last difference signal should provide better results than the first defined difference signal since it is based on the optimal subtraction. However, the results with the optimal subtraction are not significantly better than the results with the first defined signal. Moreover, the last difference signal introduces a periodicity in the residual data which is more complex to predict than the periodicity introduced by the first difference signal. Therefore, we only analyze the first defined signal performance.

Once the association almanac data – ephemeris data pair has been made, the XOR and DIFF distance calculation can be done. However, the almanac data and the ephemeris data are represented with a different number of bits. Therefore, in order to have both parameters with the same binary representation and thus in order to allow the XOR and DIFF difference calculation, the almanac data has to be transformed. This transformation consists in converting the almanac data to decimal format and in quantifying the decimal format with the same number of bits as the ephemeris data.

G.3. Process of the simplification using long term prediction ephemeris data

In this section, the process of the simplification using long term prediction ephemeris data is presented. This process should only compare broadcasted ephemeris data to the ephemeris data provided by the part of the TAS program responsible for transforming the XYZ coordinates of a satellite orbit into a set of Keplerian parameters as specified in section 4.3.1.1.4. Nevertheless, in this section, the process of the simplification using the complete program is presented and later is customized for the transforming part of the program.

The process of the simplification using long term prediction ephemeris data is similar to the process of the simplification using the almanac data. First, we construct a difference signal, which consists in selecting the ephemeris data - long term predicted ephemeris data pairs having the closest interval of validity centers. Second, we convert the two values into the

same binary format. Third and last, we calculate the probability of error of each Keplerian parameter as specified in section 4.3.1.1.3. The first step of the process is further detailed below because it varies depending on the used long term orbital prediction program.

In this dissertation, we use a long term orbital prediction program provided by one of the thesis sponsors, THALES ALENIA SPACE - France (TAS-F). This program outputs one ephemeris data set every 4h whereas the GPS L1 C/A signal broadcasts one ephemeris data set every 2h. Therefore, we subtract the same long term predicted ephemeris data set to 2 consecutive broadcasted GPS L1 C/A ephemeris data sets. In other words, a TAS ephemeris data set having a validity period going from *t* to *t+4h*, is compared with the broadcasted ephemeris data sets having a validity period going from *t* to *t+4h* and from *t+2h* to *t+6h*. The reason of choosing these two specific broadcasted ephemeris data sets is that their theoretical time of application goes from *t* to *t+2h* for the first set and from *t+2h* to *t+4h* for the second set. And the combined time of application of the broadcasted ephemeris data sets is equal to the interval of validity of the TAS ephemeris set. A mathematical model for the difference signal is given below:

$$y[x] = TAS\_ephemeris(\lfloor x/2 \rfloor) - ephemeris(x) \tag{G-4}$$

Where

- y: XOR or DIFF difference between a TAS ephemeris data set and two consecutives broadcasted ephemeris data sets with interval centers closest to the TAS ephemeris data set interval center.
- x: Index of 2-hour delta set
- TAS_ephemeris(t): TAS ephemeris data of t set of 2 hours
- ephemeris(t): Ephemeris data of t set of 2 hours
- $\lfloor t \rfloor$: Integer part of number tdata sets

The method chosen to inspect the second source of error is the same method as the method used for the analysis of the simplification process using long term prediction ephemeris data. The only difference is the generation of the difference signal which no longer uses the TAS ephemeris data sets as one of its inputs. On the contrary, this method uses ephemeris data sets output by the XYZ-orbit-coordinates-to-Keplerian-parameters TAS conversion program when using as inputs to this program the orbits defined by the GPS L1 C/A broadcasted ephemeris data sets. More specifically, first, we obtain the orbit XYZ coordinates defined by the GPS L1 C/A broadcasted ephemeris data sets and second, from these XYZ orbit coordinates, we calculate by applying the TAS program part in charge of the XYZ-ephemeris transformation the new ephemeris data sets representing the broadcasted ephemeris data sets. Note that the TAS program continues to provide one ephemeris set each 4h which means that the difference signal is constructed as indicated in (G-4) but using the new defined input.

G.4. Justification of the results of the simplification using TAS long term prediction ephemeris data

The simplification process using the TAS program of long term prediction ephemeris data is not satisfying enough. In fact, it is surprising to observe such a significant difference between the GPS L1 C/A broadcasted ephemeris data sets and the ephemeris data sets extrapolated

from the XYZ orbit ordinates defined by the same GPS L1 C/A broadcasted ephemeris data sets.

The first plausible explanation is that the transforming program of TAS does not work properly. However, the distance between an orbit defined by the broadcasted ephemeris data set and an orbit defined by the transformed ephemeris data set proves the contrary. In fact, Figure G-1 shows a really small distance between the orbits, and thus confirms that the program works very well.

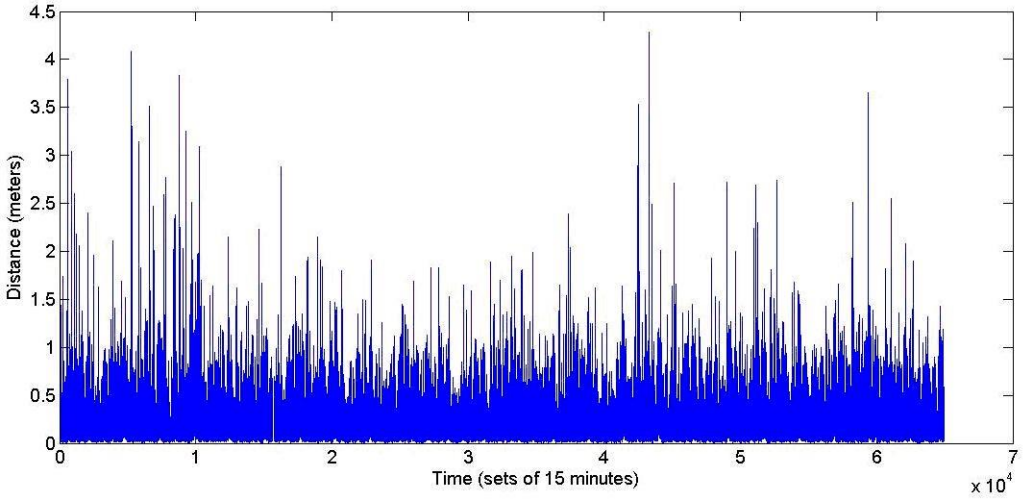

Figure G-1: Absolute distance between the broadcasted ephemeris orbit and the TAS transformed ephemeris orbit

There are two other possible explanations that justify this large binary divergence. The first one is that although the THALES program is able to obtain a really good accuracy in the decimal domain, the accuracy of the program is not enough to obtain a satisfactory binary performance as it has been explained in section 4.2.5. However, this explanation is inconsistent with the analysis conducted in annex F.4 about the influence on the satellite estimated position of the last bits of the Keplerian parameters. This analysis showed that the wrong prediction of the last bits of the Keplerian parameters changed the orbit defined by the Keplerian parameters by a distance far larger than the distances observed in Figure G-1. Therefore, the first explanation cannot be applied in this case.

The second explanation is that the ephemeris set found by the TAS converter software is not the same ephemeris data set as the broadcasted ephemeris data set. In fact, the objective of the XYZ-ephemeris converter part of the TAS long term orbital prediction program is to find an ephemeris data set which has a minimal distance to the orbit defined by the broadcasted ephemeris data set. Or, in other words, the program does not seek to optimize the binary distance between the converted Keplerian parameters and the broadcasted Keplerian parameters but rather the orbit distance. Therefore, whereas the distance between orbits is small, the Keplerian parameters binary difference is significant.

The justification of the second explanation can be found by inspecting the operations conducted by the TAS converter program and its inputs. First, the system is overestimated because the number of input points describing the orbit during the 4h period is larger than 16, the number of Keplerian parameters. Second, the function relating the ephemeris data to the XYZ orbit positions is not invertible; therefore the program is not ideal and an approximated model which relates the XYZ orbit position to the ephemeris data set has to be used. Third, the program implements a converging model based on iterations. Fourth, since the program is

not ideal and a unique solution cannot be determined, a new criterion has to be established in order to choose an acceptable solution. This criterion is that the distance between the broadcasted ephemeris data orbit and the converted ephemeris data orbit is smaller than a given threshold. Moreover, some other complementary criterions are applied such as the maximal number of iterations, the relative gain between iterations, etc. Finally, since there are several ephemeris data sets fulfilling the previous conditions, the program provides one of them. Therefore, the program has not calculated an ephemeris data set equal to the broadcasted data set, but rather an ephemeris data set fulfilling other conditions. Consequently, the second explanation is justified.

Another justification of this explanation is presented on the following tables. The first table shows some distances between broadcasted ephemeris data orbit positions and converted ephemeris data orbit positions. The second table shows the bit of most weight which is not equal between the broadcasted ephemeris data and the converted ephemeris data.

These distances shown in Table G-2 are larger for the left part of this table than for its right part. However, Table G-3 shows that the converted ephemeris data sets of its left part have more correctly predicted bits than the converted ephemeris data sets of its right part.

Ephemeris Set Id	**488**	**725**	**6369**	**6392**	**6583**	**124**	**130**	**878**	**1092**	**1432**
Orbit Distance (meters)	2,107	1,288	0,564	2,687	0,347	0,387	0,066	0,102	0,076	0,383
	0,986	0,061	0,072	1,392	0,150	0,209	0,112	0,344	0,272	0,261
	0,177	0,657	0,402	0,395	0,381	0,367	0,106	0,319	0,246	0,417
	0,437	0,828	0,421	0,312	0,344	0,242	0,066	0,113	0,088	0,244
	0,655	0,466	0,138	0,637	0,048	0,055	0,063	0,157	0,136	0,075
	0,577	0,408	0,433	0,630	0,494	0,264	0,073	0,354	0,283	0,329
	0,245	1,64	1,214	0,318	1,216	0,310	0,102	0,356	0,275	0,334
	0,328	3,14	2,146	0,261	2,064	0,0760	0,088	0,111	0,021	0,095

Table G-4: Distances between the broadcasted ephemeris data orbit and the TAS converted ephemeris data orbit

Ephemeris Set Id	488	725	6369	6392	6583	124	130	878	1092	1432
$\sqrt{A}$	24	24	24	29	25	21	21	21	21	21
E	20	22	21	24	24	19	19	19	18	22
ω	17	17	20	24	18	15	15	15	16	15
$IDOT$	6	9	8	11	13	6	8	6	7	11
$\dot{\Omega}$	19	18	21	15	21	15	17	15	17	16
Δn	11	10	12	7	12	9	9	9	9	8
$M0$	1	17	20	3	18	3	3	3	1	3
CUC	9	9	12	10	13	7	8	8	8	11
CUS	9	10	9	13	10	12	12	11	9	7
CRC	9	10	10	13	11	11	11	11	9	8
CRS	9	10	13	13	14	7	7	8	7	11
CIC	11	16	14	9	14	10	13	11	10	10
CIS	10	11	11	14	16	9	10	10	13	13
Ω_0	17	26	26	17	31	17	17	17	17	17
I	24	26	31	22	29	22	26	22	24	23

Table G-5: First different bit between the broadcasted ephemeris set and TAS converted ephemeris set

Therefore, the comparison of these two tables shows that the objective of the TAS converter program of searching the ephemeris data set which defines the closest orbit to the orbit defined by the broadcasted ephemeris data set does not take into consideration the binary distances between the ephemeris data set. And this means that this program is not ideal to make a binary prediction of the Keplerian parameters of the broadcasted ephemeris data sets.

G.5. Blackman-Tuckey method

In this section, first, the mathematical expressions of the Blackman-Tuckey method used to relate the spectral components of the observed signal with the amplitude and frequency of the sinusoids forming the observed signal on the time domain are presented. Second, the method used to estimate the phase of the sinusoids forming the observed signal is presented. Third and last, the complete process using the Blackman-Tuckey spectral estimation is given.

G.5.1. Time-frequency mathematical models of the sinusoids amplitude and frequency

In order to find the mathematical relationships between the amplitude and frequency of a sinusoid in the time domain and the characteristics of the spectral component peak of the same sinusoid in the frequency domain, we apply the Blackman-Tuckey estimation to a generic sinusoid. This sinusoid is considered as a random signal since, although the amplitude and frequency are determined, the phase is uniformly distributed over [0, 2π[.

First of all, the definition of a sinusoid signal to observe, x[n], with a random phase modeled by a uniform distribution over [0, 2π[is given below:

$$x[n] = A \cdot \sin(2\pi f_0 n + \theta_0) \quad \forall n | n \in Z \tag{G-5}$$

Where:

- A: sinusoid amplitude
- f_0: sinusoid frequency
- θ_0: random variable uniformly distributed over [0, 2π[

The determination of the power density spectrum of the observed signal using the Blackman-Tuckey estimator is achieved by first estimating the signal autocorrelation. The signal autocorrelation is obtained by using an autocorrelation estimator. In this dissertation, we use the 'biased' autocorrelation estimator, $R_{xx}[m]$, and its definition is given below [MARPLEa, 1987]:

$$R_{xx}[m] = \frac{1}{N} \sum_{n=0}^{N-1-|m|} x[n+|m|] \cdot x^*[n] \quad 0 \le |m| \le N \tag{G-6}$$

Where:

- N: Number of samples of the observed signal used in the autocorrelation estimation. Ideally chosen as $N \cdot f_0 = k$, $k \in Z$.

The estimated autocorrelation has a length of 2N-1 samples when the input signal has a length of N samples. This estimator can also be expressed as:

$$R_{xx}[m] = \frac{1}{N} \sum_{n=1}^{\infty} x[n+|m|] \cdot p_N[n+|m|] \cdot x^*[n] \cdot p_N{}^*[n] \quad 0 \le |m| \le N \tag{G-7}$$

Where:

- $p_N[n]$ = Rectangular window of length N starting at n=0 and ending at n=N-1

Once we have an estimation of the autocorrelation of the observed signal, we apply the Blackman-Tuckey method in order to obtain an estimation of the power density function of the observed signal. The Blackman-Tuckey method consists in applying a Bartlett window of half the estimated autocorrelation size to the 'biased' autocorrelation estimator. Therefore, the Blackman-Tuckey signal, BT[m], can be modeled as:

$$BT[m] = R_{xx}[m] \cdot v_{T(N/2)}[m] \quad 0 \le |m| \le N \tag{G-8}$$

Where:

- $v_{T(N)}[n]$ = Triangular window of length 2N-1 centered at 0. The window starts at n=-(N-1) and ends at n=N-1.

Applying the DFT operator on the Blackman-Tuckey signal, we obtain the power spectrum density function of the observed signal when estimated by the Blackman-Tuckey method. And from this power spectrum density function, we can relate the amplitude and frequency of the observed sinusoid with the characteristics of the estimated spectral components of the signal.

However, the mathematical expression of the power spectrum density function is not easy to interpret since the signal, and the window are mixed. Therefore, in order to simplify the

interpretation, we analyze the mean of the Blackman-Tuckey signal and we transform the mean of the Blackman-Tuckey signal in order to obtain an approximation of the power spectrum density function of the observed signal which is much easier to analyze. Remember that the observed signal is a random signal since its phase is uniformly distributed over [0, 2π[.

The mean of the Blackman-Tuckey signal, BTm[m], is [MARPLEa, 1987]:

$$BTm[m] = E\left[BT[m]\right] \quad 0 \le |m| \le N \tag{G-9}$$

$$BTm[m] = E\left[R_{xx}[m]\right] \cdot v_{T(N/2)}[m] \quad 0 \le |m| \le N \tag{G-10}$$

And since the observed signal is random and the windows are deterministic, the mean of the 'biased' autocorrelation estimator is:

$$E\left[R_{xx}[m]\right] = r_x[m] \cdot v_{T(N)}[m] \quad 0 \le |m| \le N \tag{G-11}$$

Therefore, the mean of the Blackman-Tuckey signal, BTm[m], is:

$$BTm[m] = r_x[m] \cdot v_{T(N)}[m] \cdot v_{T(N/2)}[m] \quad 0 \le |m| \le N \tag{G-12}$$

The power spectrum density function of the observed signal when estimated with the Blackman-Tuckey method can be approximated by:

$$\hat{S}_{BT}[k] = DFT\left[BT[m]\right] \approx \frac{1}{(2N)^2} S_x[k] \otimes DFT\left[v_{T(N)}[n]\right] \otimes DFT\left[v_{T(N/2)}[n + N/2]\right] \tag{G-13}$$

Where:

- DFT[f[m]]: Discrete Fourier transform of f[m]
- ⊗ : Circular convolution of signals of length equal to 2N samples

From expression (G-13), we can see that although the peaks of the ideal power density function of the observed sinusoid signal are lost due to the 2N circular convolution of the windows, the maximum value is still situated at $\pm f_0$.

Moreover, we can evaluate expression (G-13) at f_0, if $f_0 = (k_0/2N)$, in order to relate the peak value to the sinusoid amplitude. Therefore, if we approximate the power density function of the signal estimated by the Blackman-Tuckey method with another expression:

$$\hat{S}_{BT}[k] \approx \sum_{n=-N/2+1}^{N/2-1} \left(\frac{A^2}{2} \cos(2\pi f_0 n) \cdot v_{T(N)}[n] \cdot v_{T(N/2)}[n] \cdot e^{-j\frac{2\pi k}{2N} \cdot n} \right) \tag{G-14}$$

And if we only inspect the positive frequencies part, the expression is simplified to:

$$\hat{S}_{BT}{}^{+}[k] \approx \sum_{n=-N/2+1}^{N/2-1} \left(\frac{A^2}{4} \cdot v_{T(N)}[n] \cdot v_{T(N/2)}[n] \cdot e^{-j2\pi\left(\frac{k}{2N} - f_0\right) \cdot n} \right) \tag{G-15}$$

Finally, evaluating the expression at $(k_0/2N) = f_0$:

$$\hat{S}_{BT}{}^{+}[k_0] \approx \frac{A^2}{2} \cdot \sum_{n=0}^{N/2} \left(1 - \frac{n}{N}\right) \cdot \left(1 - \frac{2n}{N}\right) \tag{G-16}$$

$$\hat{S}_{BT}{}^{+}[k] \approx \frac{5 \cdot A^2 \cdot N}{48} \tag{G-17}$$

A table summarizing the relationship between the amplitude and the frequency of the observed sinusoid and the spectral components of the power density spectrum of the signal estimated by the Blackman-Tuckey method are presented below.

Sinusoid Frequency	Sinusoid Amplitude
f_0 = Ordinate X of the peak	$A = \sqrt{\frac{48 \cdot Peak\_Amp(f_0)}{5 \cdot N}}$

Table G-6: Time-frequency sinusoid amplitude and frequency mathematical relations

Nevertheless, note that we have assumed that $f_0 = (k_0/2N)$; however this is hardly the case. Therefore, neither the frequency nor the amplitude is perfectly estimated.

G.5.2. Method of the estimation of the phase

The method implemented to estimate a sinusoid phase is explained next. This method needs to have estimates of the amplitude and the frequency of the sinusoid in order to estimate its phase.

The base of this method consists in generating a new signal which is equal to the subtraction of a sinusoid to the observed signal. The subtracted sinusoid is the sinusoid which phase is being estimated. The sinusoid is generated several times, each time with a different phase. The method chooses the phase that minimizes the peak amplitude of the spectral component of the residual spectrum (difference between the observed signal and the estimated signal) situated at the same frequency as the frequency of the sinusoid which phase is being estimated.

G.5.3. Blackman-Tuckey process

The complete Blackman-Tuckey process implemented to identify all the spectral components is presented next. The process consists in 5 steps:

1- Identification of the spectral component having the maximum peak amplitude.
2- Estimate of the amplitude and frequency of the sinusoid from theoretical expressions.
3- Estimate of the sinusoid phase from section G.5.2 method.
4- Subtract the new identified sinusoid to the signal being analysed in order to create a new signal without this spectral component.
5- Back to step 1 until reaching the desired number of identified sinusoids.

G.6. PRONY model

In this annex, the mathematical expression of the PRONY model as well as the method used to determine the order of the PRONY model, or, in other words, the number of sinusoids composing of the signal, are presented.

G.6.1. PRONY model mathematical expressions

The PRONY method models a signal with N data samples with the following mathematical expression [MARPLEd, 1987].

$$\hat{x}[n] = \sum_{k=1}^{p} A_k \cdot e^{\left[(\alpha_k + j2\pi f_k)nT + j\theta_k\right]} \quad n = 0..N-1 \tag{G-18}$$

Where:

- $\hat{x}[n]$: Signal Estimation
- T: Sample interval
- A_k: Exponential amplitude
- α_k: Damping factor (1/sec)
- f_k: Sinusoid frequency (Hz)
- θ_k: Sinusoid initial phase (radians)
- p: Number of exponentials

Nevertheless, for this study, this expression is simplified because the signal is real and the damping factor is equal to 0 for all the exponentials.

$$\hat{x}[n] = \sum_{k=1}^{p/2} A_k \cdot \cos\left(2\pi f_k nT + \theta_k\right) \tag{G-19}$$

Therefore, we need to find the amplitude, frequency and phase of each existing sinusoid. In order to do so, we express the original signal without taking into account any assumptions as:

$$x[n] = \sum_{k=1}^{p} h_k \cdot z_k^n \quad n = 0..N-1 \tag{G-20}$$

$$h_k = A_k \cdot e^{j\theta_k} \tag{G-21}$$

$$z_k = e^{(\alpha_k + j2\pi f_k)T} \tag{G-22}$$

Consequently, we can determine the characteristics of the sinusoids by solving the following equation obtained from (G-20):

$$\begin{pmatrix} {z_1}^0 & {z_2}^0 & \cdots & {z_p}^0 \\ {z_1}^1 & {z_2}^1 & \cdots & {z_p}^1 \\ \vdots & \vdots & \ddots & \vdots \\ {z_1}^p & {z_2}^p & \cdots & {z_p}^p \end{pmatrix} \cdot \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_p \end{pmatrix} = \begin{pmatrix} x[0] \\ x[1] \\ \vdots \\ x[p-1] \end{pmatrix} \quad \Rightarrow \quad h = Z^{-1} \cdot x \tag{G-23}$$

Therefore, we first determine the z_k coefficients, and second, we use equation (G-23) to determine the h_k coefficients.

In order to determine the z_k coefficients, we define a homogeneous linear constant-coefficient difference equation with the z_k coefficients as roots.

$$\phi(z) = \prod_{k=1}^{p} (z - z_k) \tag{G-24}$$

$$\phi(z) = \sum_{m=0}^{p} a[m] z^{p-m} \quad a[0] = 1 \tag{G-25}$$

Where:

- z_k: Polynomial roots
- a[k]: Polynomial Coefficients

Therefore, we find the polynomial coefficients and we use them in order to find the z_k coefficients. The method selected in order to identify the a[k] coefficients is the modified covariance method [MARPLEd, 1987] because this method uses the forward and backward prediction. Therefore, the modified covariance method uses more points than the normal methods and thus uses more signal samples to find the a[k] coefficients. For a better explanation of the modified covariance method, the reader is directed to [MARPLEd, 1987]. Therefore, the determination of the a[k] coefficients by the modified covariance method is achieved by solving the following equation:

$$a = -R^{-1} r \tag{G-26}$$

$$a = (a[1] \quad a[2] \quad \dots \quad a[p])^t \tag{G-27}$$

$$R(i,j) = \sum_{n=p+1}^{N} \left[x[n-i]x[n-j] + x^*[n-p+i]x[n-p+j] \right] \quad i,j = 1..p \tag{G-28}$$

$$r(i) = \sum_{n=p+1}^{N} \left[x[n-i]x[n] + x^*[n-p+i]x[n-p] \right] \quad i = 1..p \tag{G-29}$$

Where:

- x[n]: analyzed signal at instant n

Therefore, using expression (G-26), we can find the a[k] coefficients defining a polynomial whose roots are the z_k coefficients. This means, that from the a[k] coefficients, we can determine the z_k coefficients, and using equation (G-23), we can find the h_k coefficients from the z_k coefficients.

Finally, once all the h_k and z_k coefficients have been determined, the parameters defining the exponentials components are identified using the following expressions [MARPLEd, 1987].

$$A_i = |h_i| \tag{G-30}$$

$$\theta_i = \arctan\left[\mathrm{Im}(h_i) / \mathrm{Re}(h_i) \right] \quad radians \tag{G-31}$$

$$\alpha_i = \ln|z_i| / T \quad 1/\sec \tag{G-32}$$

$$f_i = \frac{\arctan\left[\mathrm{Im}(z_i) / \mathrm{Re}(z_i) \right]}{2\pi T} \quad Hz \tag{G-33}$$

Remember that we have previously said that our signal is such that the damping factor of all the exponentials is equal to 0; therefore, equation (G-32) is not used.

Moreover, since the analysed signal is real, the number of exponentials has to be an even number. Therefore, from equations (G-30), (G-31) and (G-33), we must find twice the same amplitude, one positive frequency and one negative frequency with the same absolute value and one positive phase and one negative phase with the same absolute value.

G.6.2. PRONY model order selection

In the previous section, it has been shown how to determine the sinusoids characteristics from the analyzed signal. However, the previous section assumed that the number of exponentials is known. However, the number of exponentials is usually unknown, and it is one of the model parameters to be determined. Therefore, the first action to undertake before beginning with the estimation of the a[k] coefficients is to determine the number of sinusoids forming the analyzed signal. Note that twice the number of sinusoids is equal to the number of exponentials, also called the model order. Therefore, we need first to determine the model order.

There are different methods which can be used in order to find the number of exponentials forming a signal. And these methods can be sorted out into two main groups, the methods which base their criterion on the signal statistics properties [MARPLEc, 1987] and the methods which uses the signal singular values (SVD or singular value decomposition) [MARPLEd, 1987]. Each group of methods has its advantages and drawbacks such as the percentage of times that the number of exponentials is exactly found, upper or under estimations, or their processing time.

In this case, the group of methods best suiting our needs is the second type of methods. The reason is that the first type of methods provides a good performance when the number of exponentials generating the signal is low, between 3 and 5, but these methods provide an underestimation when the number of exponentials is larger. Moreover, the methods based on signal statistics properties, in addition of being less robust to the influence of the signal noise, obtain an abnormal estimation when the difference signal resulting from the application of the simplification process using almanac data is inspected. However, the method based on the signal singular values is quite slower than the methods based on signal statistics properties.

The main idea of the method based on the signal singular values is to differentiate the singular values associated to the signal exponentials from the singular values generated by the signal additive noise. Therefore, since the singular values of the exponentials are some orders of magnitude larger than the singular values of the noise, a threshold can separate the two types of singular values. And thus, this separation allows the identification of the total number of exponentials forming the signal.

Therefore, this method consists of two steps. First, we calculate the singular values of the matrix of the signal associated to the forward prediction, X^f_q, the singular values of the matrix of the signal associated to the backward prediction, X^b_q, or the singular values of the union of both matrixes. The mathematical expression of the matrixes is given below:

$$X_p^f = \begin{pmatrix} x[q] & x[q-1] & \cdots & x[1] \\ x[q+1] & x[q] & \cdots & x[2] \\ \vdots & \vdots & \ddots & \vdots \\ x[N-1] & x[N-2] & \cdots & x[N-q] \end{pmatrix} \tag{G-34}$$

$$X_q^b = \begin{pmatrix} x[2] & x[3] & \cdots & x[q+1] \\ x[3] & x[4] & \cdots & x[q+2] \\ \vdots & \vdots & \ddots & \vdots \\ x[N-q+1] & x[N-q+2] & \cdots & x[N] \end{pmatrix} \quad \text{(G-35)}$$

Second, we determine the number of exponentials forming the signal as the number of singular values larger than a fixed threshold. Note that q is the maximum model order possible when using the matrixes X^f_q and X^b_q. Therefore, if the method application results into a number of exponentials p equal to the maximum considered model order minus one, p = q-1, the method has to be applied again with a bigger value q. And this process is repeated until p is lower than q-1. Nevertheless, for the analyzed signal, if p is too big, the application of the method results into the dummy solution, p = q-1.

G.7. Artificial Neural Network method

In this section, the neural network structure is given and the learning phase of a neural network is described.

G.7.1. Neural Network structure

The fundamental and smaller element of a neural network is the neuron. Each neural network is formed by several neurons distributed in different layers, where each layer is hidden from the user. For example, Figure 4-3 presents a neural network with only one hidden layer. Nevertheless, we implemented only 1 layer neural networks.

A general neuron structure is given below [BHADESHIA, 1999] [YEGNANARAYANAa, 2006].

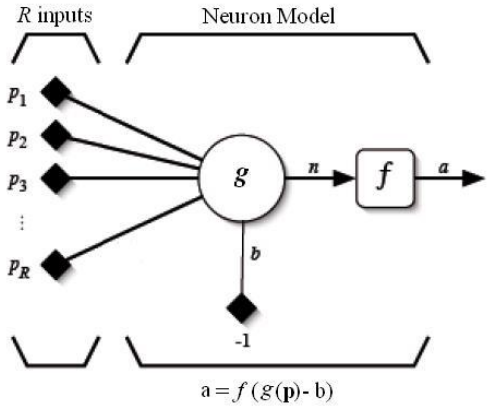

Figure G-2: Generic neuron model

From Figure G-2, it can be seen that a neuron is formed by a *g* function, also called combination function, a neuron bias, also called activation threshold and represented by the *b* variable, and a *f* function, also called the activation function. The function of these elements is presented next.

The combination function is responsible for manipulating the neuron inputs in order to create a suitable input to the activation function. Moreover, the choice of the combination function defines the type of implemented neural network. There are two main types of neural networks, the Multi-layer Perceptron (MLP) and the Radial Basis Function (RBF). The MLP

neural networks combination function implements a linear combination of the neuron inputs [YEGNANARAYANAc, 2006].

$$g_j(P) = \sum_{i=1}^{R} w_{ij} \cdot p_i \quad \text{(G-36)}$$

Where:

- $g_j(x)$: j^{th} neuron combination function
- P: vector of neuron inputs
- p_i: i^{th} neuron input
- w_{ij}: i^{th} neuron input weight of the j^{th} neuron allowing the correct output prediction

Note that the neuron input is in our case the known samples of the difference signal to be predicted. The last known samples of the difference signal before the first unknown sample to be predicted.

The RBF neural networks combination function calculates the distance or norm between the neuron inputs, interpreted as a vector, and a given vector [CHEN et al, 1991].

$$g_j(P) = \left\| P - P_j \right\| \quad \text{(G-37)}$$

Where:

- P_j: j^{th} neuron vector of delays allowing the correct output prediction

The activation function is responsible for introducing a non-linearity inside the neuron model. There are several activation function choices such as the sigmoid function, the Heaviside function and some trigonometric functions [YEGNANARAYANAb, 2006]. We chose to implement the Arctangent function.

The activation threshold is used to determine if the combination function output has reached a fixed value or threshold. Therefore, the activation threshold determines the input sign of the activation function [YEGNANARAYANAb, 2006].

Until now, we described the generic structure of a neuron; however, any neural network is formed by several neurons. Therefore, the neuron outputs are either linearly or non-linearly combined in order to provide one output or several outputs of the neural network which suit the needs of the problem. In our case, we implemented a linear combination of the neuron outputs in order to obtain a single final sample. A scheme of a generic neural network of 1 layer is given below [BHADESHIA, 1999].

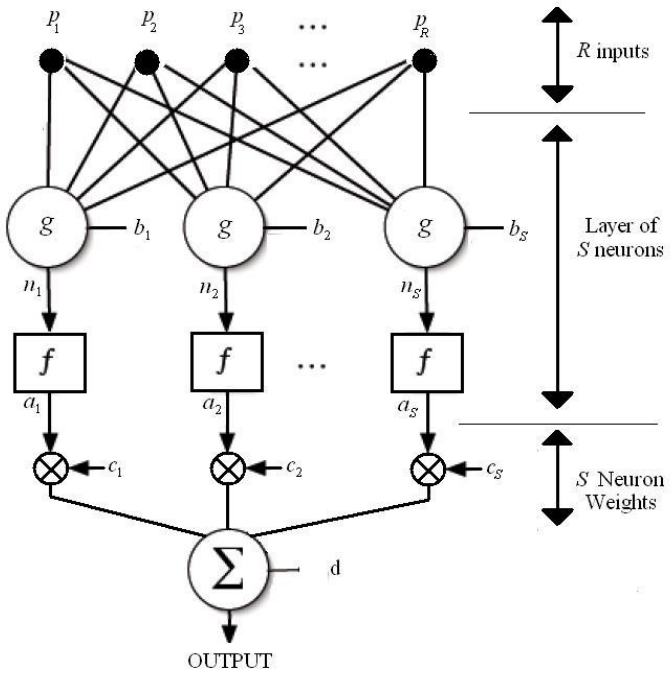

Figure G-3: Neural network structure

Therefore, the mathematical expression of the neural network output at instant n is [YEGNANARAYANAc, 2006]:

$$\hat{x}[n] = d + \sum_{j=1}^{S} c_j \cdot f\left[g_j(P) + b_j\right] \tag{G-38}$$

Where:

- $\hat{x}$ [n]: neural network output at instant n
- d: external constant
- c_j: neuron weights

Customizing expression (G-38) first for a MLP neural network and second for a RBF neural network, we obtain the following equations [YEGNANARAYANAc, 2006] [CHEN et al, 1991].

$$\hat{x}[n] = d + \sum_{j=1}^{S} c_j \cdot f\left(b_j + \sum_{i=1}^{R} w_{ij} \cdot x[n-i]\right) \tag{G-39}$$

$$\hat{x}[n] = d + \sum_{j=1}^{S} c_j \cdot f\left(\left\|X - X_j\right\|\right) \tag{G-40}$$

Where:

- x[n-i]: signal input at instant n-i
- X: vector of signal inputs from x[n-1] to x[n-R]
- X_j: constant j^{th} neuron vector of delays

G.7.2. Neural Network learning phase

All the values of the parameters defined in the previous section, such as the neuron weights, the activation thresholds, the input weights of each neuron and the external constant, have to be correctly determined in order to allow the desired signal prediction. Therefore, the first step to take in order to predict a signal by using a neural network is to launch the neural network learning phase which is responsible for determining the values of the neural network parameters.

More specifically, this phase consists in varying the values of the parameters in order to find the values which minimize the power of the error between the value of the predicted sample and the value of the sample which is being predicted. The power of the error is denoted as E. Since we are applying the learning phase and the value of the sample which is being predicted has to be known, the values of the neural network output and the neuron inputs are the following. The sample of the neural network output is a known sample of the difference signal and the neuron inputs are the known delayed samples of the signal which should be used to predict the output sample if it was unknown. The variation of the value of the constant parameter is achieved as specified below [YEGNANARAYANAb, 2006].

$$parameter[n+1] = parameter[n] + \mu \frac{\partial E}{\partial parameter} \tag{G-41}$$

$$E = \frac{1}{2}\sum_{i=1}^{L}\left(e_i[n]\right)^2 \tag{G-42}$$

$$e_i[n] = x_i[n] - \hat{x}_i[n] \tag{G-43}$$

$$\frac{\partial E}{\partial parameter} = \frac{1}{\partial parameter}\left(\frac{1}{2}\sum_{i=1}^{L}\left(x_i[n] - d - \sum_{j=1}^{S} c_j \cdot f\left[g_j(P) + b_j\right]\right)^2\right) \tag{G-44}$$

Where:

- $e_i(n)$: Prediction error between the i^{th} known signal sample $x_i[n]$ and the i^{th} predicted sample $\hat{x}_i[n]$ at instant n. $\hat{x}_i[n]$ is modeled as indicated in equation (G-38).
- L: number of samples to predict/estimate. In our case, L is equal to 1.
- μ: learning rate.

The learning rate must not be too large because the neural network would not converge. And it must not be too small because the converging process can be too slow.

Finally, the specific steps of the learning process are explained below.

1. Random selection of a group of consecutives signal samples.
2. The oldest signal samples are used as the neural network inputs, or neuron inputs, to calculate a signal sample prediction of the latest signal sample of the group.
3. The signal sample prediction is subtracted to the latest signal sample of the group in order to calculate the prediction/estimation error.
4. The variations of the parameter values, neuron weights, the activation thresholds, the input weights of each neuron and the external constant, are calculated from the prediction error and the inputs.
5. Back to step 1 until a certain number of iterations is reached.

Annex H. Transmission channels and simulators schemes

In this annex, the simulator schemes used in this dissertation in order to simulate the signal transmission through an AWGN channel and through a mobile channel are presented. Moreover, the mathematical expression of the autocorrelation function of a mobile channel is given and the mathematical model of a frequency-selective channel is described.

H.1. Autocorrelation function of a mobile channel

In order to determine the autocorrelation function of a channel, first the received signal and the channel impulse response has to be modeled.

The received signal is modeled as the addition of the LOS signal and the multipath component. The multipath component or the echoes can be modeled as a succession of transmitted signals arriving at the receiver antenna at different delays since they are the refractions, reflections, etc, of the transmitted signal arriving at the receiver antenna by another path different from the LOS path. Moreover, due to the variability of the surroundings, the movement of the transmitting satellite and the receiver, the delay, the phase and the attenuation of the echoes vary at each instant. Therefore, the received band-pass signal at the receiver antenna output can be expressed as [PROAKISg, 2001]:

$$v(t) = \sum_n \alpha_n(t) s[t - \tau_n(t)] \tag{H-1}$$

With:

- v(t): received band-pass signal
- s(t): transmitted band-pass signal
- $\alpha_n(t)$: Complex attenuation factor for the n^{th} path
- $\tau_n(t)$: Delay of the n^{th} path

The transmitted signal can be also expressed using its equivalent baseband complex envelope expression, $s_l(t)$:

$$s(t) = \mathrm{Re}[s_l(t) \cdot \exp[j2\pi f_c t]] \tag{H-2}$$

With:

- f_c: Carrier frequency of the signal

Thus, the equivalent baseband complex envelope received signal, $v_l(t)$, can be expressed as:

$$v_l(t) = \sum_n \alpha_n(t) \cdot \exp[-j2\pi f_c \tau_n(t)] \cdot s_l[t - \tau_n(t)] \tag{H-3}$$

Finally, the equivalent baseband complex envelope channel impulse response, $c(\tau, t)$, can be described as:

$$c(\tau; t) = \sum_n \alpha_n(t) \exp[-j2\pi f_c \tau_n(t)] \delta[\tau - \tau_n(t)] \tag{H-4}$$

From equation (H-4), parameter t is used to indicate the instant of time at which the attenuation factor, $\alpha_n(t)$, and the propagation delay, $\tau_n(t)$, of the n^{th} echo are evaluated. Parameter τ is used to determine the echo which has a propagation delay equal to τ.

Then, from this channel impulse response expression, the channel autocorrelation can be defined [PROAKISg, 2001]. The $c(\tau, t)$ is assumed wide-sense-stationary.

$$\phi_c(\tau_1, \tau_2; \Delta t) = \frac{1}{2} E\left[c^*(\tau_1; t) c(\tau_2; t + \Delta t)\right] \quad \text{(H-5)}$$

Where:

- τ_1, τ_2: Echoes propagation delays. Determine the propagation delay of the echoes which are compared.
- Δt: Channel variation time. Since the propagation delay, the phase shift and the attenuation factor of an echo vary with time, the channel impulse response also varies with time. Therefore the Δt parameter is used to indicate the amount of time between the two channel impulse responses which are compared.

Therefore, assuming that the attenuation and phase shift of the channel associated with the echo having a propagation path delay equal to τ_1 is uncorrelated with the attenuation and phase shift of the echo having a path delay equal to τ_2, the Fourier transform of the autocorrelation function results in [PROAKISg, 2001]:

$$\phi_C(\Delta f; \Delta t) = \frac{1}{2} E\left[C^*(f_1; t) C(f_2; t + \Delta t)\right] \quad \text{(H-6)}$$

Where:

- Δf: Frequency difference between two frequencies of the mobile channel.
    - $\Delta f = f_2 - f_1$

This Fourier transform is called *spaced-frequency, spaced time correlation function of the channel* [PROAKISg, 2001]. Therefore, imposing Δt equal to 0, the channel autocorrelation as a function of the frequency is obtained from this expression and thus the channel coherence bandwidth can also be obtained. The channel coherence bandwidth, $(\Delta f)_c$, is the width of the region which is not null of the *spaced-frequency, spaced time correlation function of the channel* when Δt is equal to 0 and its value only depends on the frequency difference, Δf. Then, since this autocorrelation specifies how the channel is correlated with itself for two different frequency values, we can assume that two sinusoids with frequency separation larger than $(\Delta f)_c$ are affected differently by the channel.

Moreover, from expression (H-6) and assuming Δf equal to 0, the channel autocorrelation function variation along the time can be obtained. If the channel does not vary along the time, the channel is considered time-invariant and the autocorrelation function should be a constant value equal to 1. However, since the channel varies with time, the autocorrelation function has a limited time span, and the length of time region where the autocorrelation function is not zero is the called channel coherence time, $(\Delta t)_c$.

In other words, the delay, attenuation and phase of each received echo vary for each instant of time. More precisely, although the echoes are completely independent among them, they are not uncorrelated with themselves over time. This means that the values of delay, phase and attenuation of the n^{th} echo at time t_1 are related with the same values at time t_2. Therefore, the $(\Delta t)_c$ represents the duration of time in which the channel remains about constant. This

variation of the channel is termed *fading* [PROAKISg, 2001]. Therefore, the longer the signal symbol duration is, the more the symbol amplitude and phase vary.

H.2. Frequency-selective and not slowly fading mathematical model

In the event that the transmission channel is frequency-selective and not slowly fading at the receiver correlator output, the final mathematical model can also be represented by the Perez-Fontan model. In this case, the only difference with the previous presented frequency non-selective channel model is the presence of the other tap delays –or main echoes- of Figure 3-7 and their complex coefficients $c_n(t)$. Therefore, if the distribution of the $c_n(t)$ coefficients is known, the frequency-selective channel may be easily reproduced

As previously said in section 3.2.2.2.2, the complex coefficient $c_n(t)$ models the contribution to the received signal amplitude and phase of a main echo and of all the received echoes between half the time to the previous main echo and half the time to the next main echo. Therefore, each complex coefficient $c_n(t)$ is independent from any other coefficient $c_n(t)$ but they all follow the same statistical distribution. Moreover, the complex coefficient $c_1(t)$, which models the contribution of the LOS signal and of the echoes received until half the time of the first main echo, is also different from any complex coefficients $c_n(t)$ but also follows a different statistical distribution. Nevertheless, the contribution made to a main echo by all the echoes received between half the time to the previous main echo and half the time to the next main echo is not statistically different from the contribution made to the LOS signal by the echoes received until half the time to the first main echo. In fact, the only difference is the statistic of the main echo with respect to the LOS signal. Therefore, since the complex coefficient $c_1(t)$ was modeled as a Loo distribution, the addition of a Log-normal variable representing the LOS signal and a Rayleigh variable representing the echoes, the complex coefficients $c_n(t)$ are directly modeled by a Rayleigh variable since they only represent the addition of echoes.

One last factor to take into account during the modeling of the complex coefficients $c_n(t)$ is the power carried by the LOS signal and each main echo. In fact, since the echoes travel more distance than the LOS signal plus the extra attenuation introduced by the obstacles, the average power of any echo is smaller than the average power of the LOS signal. And the same statement is verified for echoes received later than other echoes. Therefore, in order to fit this decrease of the average power to our $c_n(t)$ coefficients, we have to multiply each $c_n(t)$ by a factor which decreases with each growing n. These factors are extracted from the average power delay profiles database. Therefore, the complex coefficients $c_n(t)$ can be finally modeled as:

$$c_n(t) = w \cdot e^{j\varphi} \cdot P_m((n-1) \cdot T_c) \qquad \text{(H-7)}$$

Where:

- w : Rayleigh variable with an average power equal to sqrt(MP) (see section 3.2.2.3.4)
- φ: Uniform variable on $[0, 2\pi[$
- $P_m(\tau)$: Linear average power delay profile evaluated at τ

An example of an average power delay profile is given below:

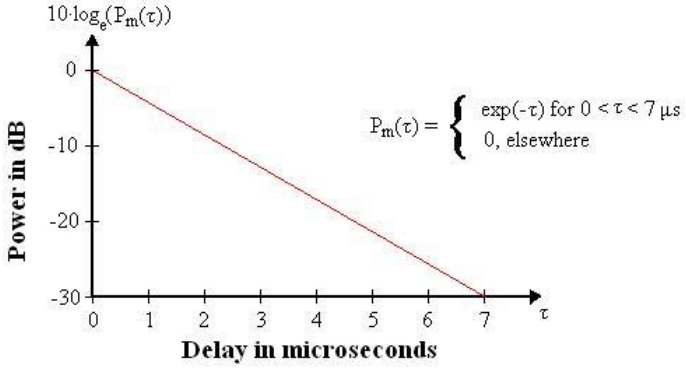

Figure H-1: Typical average power delay profile for suburban and urban areas

Where:

- $P_m(\tau)$: Linear average power delay profile

Finally, since the transmission channel is frequency-selective and not slowly fading at the correlator output, the correlation between the received signal PRN code and the generated local replica is not ideal and thus the main echoes are not eliminated. This means that if we want to implement a simulator for this kind of channel, the correlation between the received signal PRN code and the generated local replica has to be completely recreated.

To sum up, the mathematical model of a frequency-selective channel which is not slowly fading is the LOS signal plus L main echoes separated from each other by the signal symbol duration, T, -in this case T is equal to T_c- where $L = [T_m/T_c] + 1$. The LOS signal is multiplied by the complex coefficient $c_1(t)$ which follows a Loo distribution (defined in section 3.2.2.3.4) and each main echo is multiplied by a complex coefficient $c_n(t)$ which follows a Rayleigh distribution. The distribution parameters depend on the Markov state at which the model is at each instant. Moreover, each complex coefficient of the main echoes is multiplied by a factor simulating the channel average power delay profile. Finally, the correlation between the generated local replica and the complete receive signal –LOS signal plus main echoes- has to be completely calculated.

H.3. Simulator schemes

The simulator schemes of the transmission of a signal through an AWGN channel or a mobile channel are presented next.

H.3.1. AWGN channel simulator

The AWGN channel simulator depends on the tracking process. Therefore, two cases are detailed in the following subsections.

H.3.1.1. Ideal carrier phase tracking process

An ideal carrier phase tracking is assumed when the C/N_0 at the antenna output is so high that it induces a negligible PLL carrier phase estimation error. And this means that no useful

signal power is lost: the useful signal power at the receiver antenna output is the same as the useful signal power at the demodulator block input.

$$C_{RF/IF\ filter\ output} = C_{Input\ Demodulator} \quad \text{(H-8)}$$

Therefore, since all the useful power can be used to demodulate the signal, the C/N_0 and E_b/N_0 figures of merit can be linked as it is done in annex B.3. Remember that the demodulation/decoding performance expressed as a function of the E_b/N_0 provides different information that the demodulation performance expressed as a function of the C/N_0 as it is also said in annex C.5.

Finally, in an AWGN channel, in order to simulate the received signal at the correlator output when an ideal carrier phase tracking process is assumed, we have to generate the model presented in equation (3-14) but neglecting the cosine term. More specifically, we generate the data symbols with a power equal to 1 plus a Gaussian process representing the AWG noise whose generation is described in annex H.3.1.3. Note that for the reception of a GNSS signal when the subcarrier has been removed, the data symbol is a BPSK symbol which means that is either represented by a +1 or -1 value. Moreover, the simulated signal is a baseband signal since it not necessary to use a more complex model.

H.3.1.2. Non ideal carrier phase tracking process

A non-ideal carrier phase tracking process happens when the C/N_0 at the antenna output is not large enough to achieve a negligible PLL carrier phase estimation error but still high enough to allow the PLL lock. And this means that not all the useful signal power can be used to demodulate the signal.

There are two ways of obtaining the signal demodulation performance when the carrier phase tracking process is not ideal. The first one consists in averaging the demodulation performance values of an ideal carrier phase tracking process weighted by the probability of the carrier phase estimation errors. The second one consists in introducing the phase estimation errors into the simulation. Both methods are explained next. Nevertheless, the first method can only be applied to signals implementing an interleaver. In fact, the first method assumes that the errors of the received bits are independent, but the PLL introduces correlated errors in time. Therefore, only the signals which break this time correlation between errors through the application of an interleaver can be modeled with the first method. A more detailed justification is given in the method explanation (see next section).

H.3.1.2.1. Ideal demodulation performance values modification method

The effect of a non-ideal tracking process is the loss of part of the useful signal power. This loss can be modeled as a multiplying factor L to the data symbol power and it changes for each estimated carrier phase value. In this section, it is assumed that the PLL generates a different constant local carrier phase for each received data symbol.

More specifically, the parameter L represents the instantaneous useful signal power loss due to the non-ideal carrier phase tracking process and can be expressed as:

$$L = \cos^2(\varepsilon_\theta) \quad \text{(H-9)}$$

Where:

- ε_θ: Carrier phase estimation error.

- L: Instantaneous Power Loss

The carrier phase estimation error at the correlator output is due to 4 sources of error as has been said in subsection 3.1.3.2 and shown in equations (3-5) and (3-6). The Allan deviation noise and the oscillator vibrations are considered negligible in front of the two other sources in this dissertation and thus, the carrier phase estimation error has been modeled as a Gaussian variable with an average determined by the signal dynamic stress error and with a variance determined by the thermal noise. Assuming that the higher dynamics of the signal are represented by a constant jerk and that the chosen PLL discriminator is the Q discriminator, the carrier phase estimation error is modeled as Gaussian variable with the following characteristics.

$$\varepsilon_\theta \rightarrow N\left(m_{\varepsilon_\theta} = 2\pi \frac{T_I^3}{K_3} \cdot \frac{m \cdot g}{\lambda},\ \sigma^2_{\varepsilon_\theta} = \frac{B_L}{C/N_0} \right) \tag{H-10}$$

Where:

- B_L: PLL filter Bandwidth
- T_I: Coherent integration Time
- m: Constant number of jerks
- g: gravity acceleration
- λ: Wavelength of the carrier signal frequency
- T_I: Integration Time
- K_3: coefficient given by [STEPHENS and THOMAS, 1995], in their description of discrete-update PLL.

Moreover, the carrier phase estimation error samples are not independent among them since they represent the evolution in time of a white Gaussian noise filtered by the PLL close loop function equivalent filter.

From equation (H-9), it is easy to calculate the average loss of power by simply averaging the instantaneous power loss (L) at the correlator output weighted by their probability density function. And from the subtraction of the average loss of power divided by the symbol duration to the nominal E_b/N_0 value, a new average E_b/N_0 value can be calculated. Nevertheless, the new average BER cannot be calculated from the new average E_b/N_0.

The reason is that the E_b/N_0 averaging process is not equivalent to average the BER since there is not always a linear relationship between them. Therefore, the new BER has to be found through the average of the instantaneous BER and not through the average of the instantaneous E_b/N_0 values.

An instantaneous BER value is simply the BER value which would be obtained by a E_b/N_0 with the same value as the instantaneous E_b/N_0 value if the tracking process was considered ideal. Therefore, the ideal tracking process relationship between the BER and the E_b/N_0 is also used to link the instantaneous BER and E_b/N_0 values. Nevertheless, note that this statement is only true when the carrier phase tracking errors at the correlator output are independent. The reason is that the correlation among the carrier phase tracking errors at the correlator output correlates the useful power of the received data symbol at the correlator output. Therefore, the data symbol errors are no longer independent as they were when the relationship between the BER and the E_b/N_0 assuming an ideal carrier phase tracking was calculated. Therefore, in

order to apply this method, the signal needs to implement an interleaver. An interleaver is an element which redistributes the transmitted data symbols into a new order at the transmitter and thus allows breaking the correlation between consecutive samples. The original distribution is recovered at the reception by the desinterleaver. A better definition is given in section 6.2.4.

Finally, an instantaneous E_b/N_0 value is calculated by multiplying the nominal E_b/N_0 value by the instantaneous loss (L). And the probability of having an instantaneous E_b/N_0 at a given moment depends on the distribution of L. And the distribution of L depends on the distribution of ε_θ.

Equation (H-11) expresses the BER obtained for a given nominal E_b/N_0 when the tracking process is not assumed ideal.

$$BER_{real} = \int_{-\infty}^{+\infty} BER\left[\left.\frac{E_b}{N_0}\right|_{ideal} \cdot \cos^2(\varepsilon_\theta)\right] \cdot p(\varepsilon_\theta) \cdot d\varepsilon_\theta \qquad \text{(H-11)}$$

Where:

- BER[x]: Ideal carrier phase tracking BER obtained at an E_b/N_0 value equal to x.
- p(x): Probability density function of x.

The BER as a function of the E_b/N_0 when assuming ideal carrier phase tracking can be recovered from the already published theoretical curves of the signal or channel code, or it can be obtained through simulations as it has been explained in section H.3.1.1.

Introducing the carrier phase estimation error probability, equation (H-11) results into:

$$BER_{real} = \frac{1}{2\pi\sqrt{\sigma_{\varepsilon_\theta}^2}} \int_{-\infty}^{+\infty} \exp\left[\left(\varepsilon_\theta - m_{\varepsilon_\theta}\right)^2 / \left(2\sigma_{\varepsilon_\theta}^2\right)\right] \cdot BER\left[\left.\frac{E_b}{N_0}\right|_{ideal} \cdot \cos^2(\varepsilon_\theta)\right] \cdot d\varepsilon_\theta \qquad \text{(H-12)}$$

And from this equation, the BER as a function of the E_b/N_0 when the carrier phase tracking process is affected by thermal noise and a constant jerk can be calculated. Moreover, note that using the ideal conversion between C/N_0 and $E_b/N_{0\ ideal}$, the result can be expressed as a function of the C/N_0.

Previously, it has been commented that the proposed method can only be applied when an interleaver is implemented on the signal since it allows breaking the correlation between consecutive data symbols power introduced by the carrier phase estimation errors at the correlator output. Nevertheless, not all the interleavers can break this correlation. In fact, the redistribution of the data symbols made by the interleaver has to separate two samples of the original distribution of at least a minimum determined time. This time is specified by the PLL filter impulsive response. And this time is the time between the first filter impulse response sample and the sample of which the amplitude is attenuated by a factor of (1/e) with respect to the maximal filter impulse response amplitude. Note that this time is specified by the typical criterion of correlation loss of filtered samples. Moreover, this time is about the inverse filter bandwidth that is in this case of 10Hz. Therefore, the interleaver has to separate two consecutives samples of at least 0.1s.

Therefore, for GPS L1C signal transmitted at a rate of 100 symb/s and with an interleaver of 38 rows (data written) by 46 columns (data read), the time between the reception of two consecutives symbols is 0.38 seconds, which largely fulfills the criterion of correlation loss. For GALILEO E1 OS, the signal is transmitted at a rate of 250 symb/s with an interleaver of 8

rows (data read) and 30 columns (data written), which means 0.12 seconds between the reception of two consecutives symbols. However, the reception time between two bits separated by 9 bits in the original data symbol distribution is only 32ms, which means that they are correlated since this time is smaller than 0.1s. To sum up, the carrier phase estimation errors at the correlator output are independent among them for GPS L1C signal but they are not entirely uncorrelated for GALILEO E1 OS signal.

H.3.1.2.2. Noise error sources implementation

The second method consists in implementing the different error sources affecting the signal in the simulator and in demodulating the resulting signal in order to obtain the BER, the WER and the EER.

There are two ways of implementing these error sources. The first technique is to implement the error sources after the PLL processing, which implies the carrier phase estimation error implementation defined in subsection 3.1.3.2. The second technique is to implement the original error sources before their PLL processing and thus a PLL has also to be implemented.

H.3.1.2.2.1. Noise error sources implementation after the PLL processing

The simulator has to implement the phase estimation error sources after the PLL processing. Therefore, the simulator has to generate and directly demodulate the signal described in equation (3-15), which represents the I channel of the signal at the correlator output, ri[k]. It is modeled as:

$$r_i[k] = \frac{A}{2} \cdot d[k] \cdot \cos\left(\varepsilon_\theta[k]\right) + n[k] \qquad \text{(H-13)}$$

Where:

- $d_m[k]$: Navigation data at epoch k
- $\varepsilon_\theta[k]$: Phase estimation error
- $\varepsilon_\theta[k]$: $\theta[k] - \hat{\theta}[k]$
- n[k]: Filtered Gaussian noise

Consequently, the only difference with the simulator of an AWGN channel with ideal tracking process is the phase estimation error $\varepsilon_\theta[k]$ generation. And the carrier phase estimation error $\varepsilon_\theta[k]$ is equal to the addition of two factors: a constant factor representing the dynamic stress error bias and a variable factor representing the thermal noise.

For practical purposes the generation of the carrier phase estimation error due to thermal noise at the PLL output is generated from a white Gaussian noise filtered by the closed loop PLL transfer function. The white Gaussian noise variance is defined by (3-8); therefore, the closed PLL transfer function gain is set to 1 in order to conserve the final noise power.

The simulation scheme is the following:

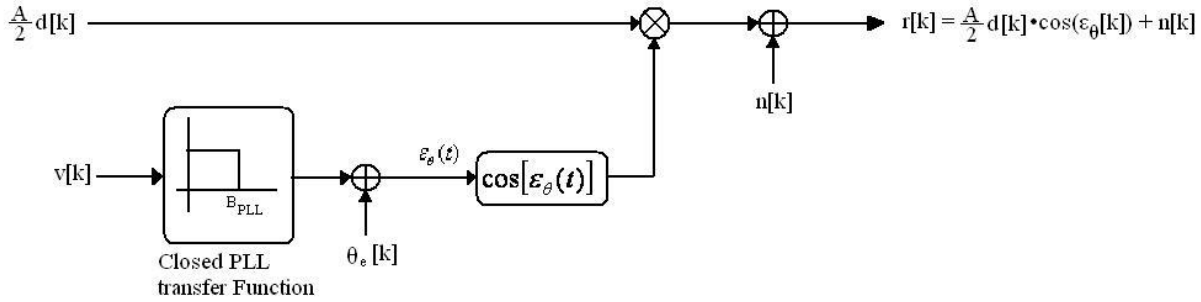

Figure H-2: AWGN channel with a non-ideal tracking process

Where:

- A: useful signal amplitude
- d[k]: Navigation data at epoch k
- n[k]: discrete white Gaussian noise – $N(0, \sigma^2)$
- v[k]: discrete white Gaussian noise – $N(0, \sigma_\varepsilon^2)$
- $\theta_e[k]$: dynamic stress error bias

Note that n[k] and v[k] are generated as specified in the begging of annex H.3.1.3 with $A \cdot d_m[k]/2 = 1$.

The coefficients of the filter representing the closed PLL transfer function are defined in [STEPHENS and THOMAS, 1995]. Nevertheless, these coefficients make that the closed PLL transfer function has a gain different from 1; therefore, in order to keep the desired input noise power, the output noise has to be multiplied by a corrector factor.

H.3.1.2.2.2. Noise error sources implementation before the PLL processing

In this case, the simulator has to implement the phase error sources over the signal before applying the PLL. Therefore, the main complication in this method is the PLL implementation since the phase error sources are much easily represented. In fact, the influence of the thermal noise over the tracking process is simply generated by filtered white Gaussian noise addition to the transmitted signal, the same addition used to generate the noise over the demodulation performance (see section H.3.1.3).

Another important decision to make is to decide at which rate the simulator is going to generate the input samples. This rate can go from the signal sampling rate, which depends on the signal bandwidth, to any rate which results from the division of the data symbol rate by an integer number. In any case, each generated input sample has a different carrier phase value. However, the PLL can generate new carrier phase estimations at the same rate that the generated input samples or it can generate new carrier phase estimations at a slower rate. For this last option, the PLL carrier phase estimation is used to remove the carrier phase of more than one input sample.

Additionally, this simulator has to generate the correlation between the received signal PRN code and the generated local replica. The correlation result is well known when the code delay is determined and when the carrier phase estimation error is constant during the length of the

received PRN code. But, in this case, since the carrier phase estimation is not constant, the contribution of the correlation has to be calculated differently. This contribution can be determined by first calculating the partial correlations of the parts of the received signal PRN code which has the same carrier phase, by second multiplying the partial correlations by their carrier phase estimation error, and by, third and last, adding all the weighted partial correlations composing the total correlation between the received PRN code and the generated local replica.

The scheme of the simulator is given below.

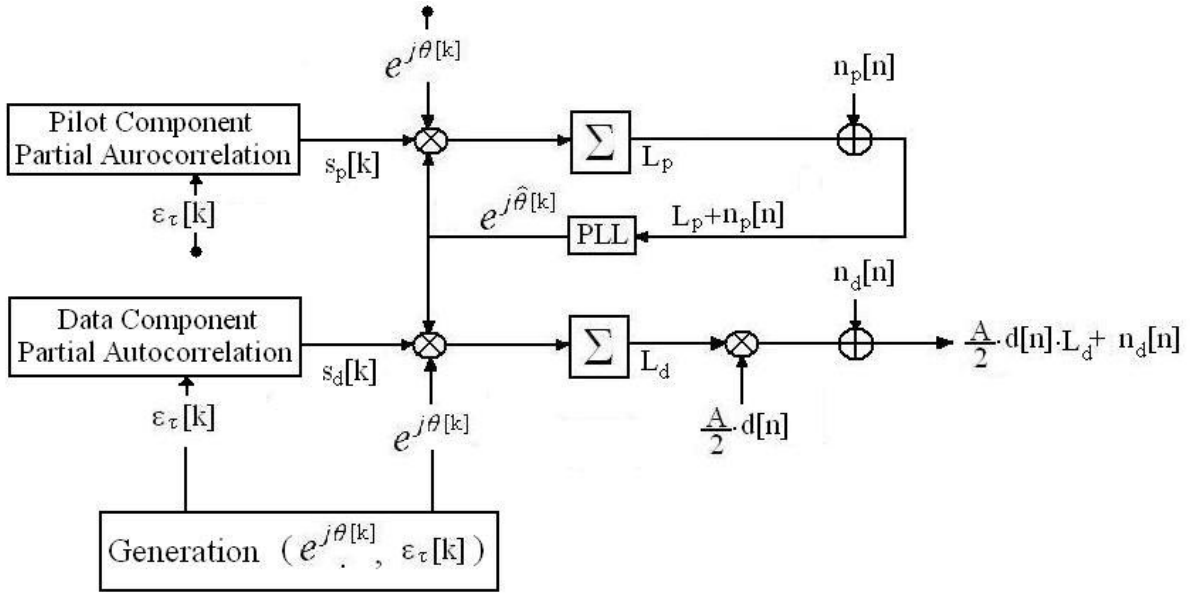

Figure H-3: PLL simulator scheme

Where:

- $\theta[k]$: Incoming signal carrier phase
- $\hat{\theta}[k]$: PLL signal carrier phase estimation
- $\varepsilon_\tau[k]$: Code phase delay at epoch k.
- d[n]: Navigation data at epoch n
- $n_p[n]$: Pilot channel noise at epoch n.
- $n_d[n]$: Data channel noise at epoch n.
- k: epoch at chip sampling rate
- n: epoch integration output rate
- L_x: Complex number at the output of the integration of channel x

$$L_x = \frac{1}{M}\sum_{k=0}^{M-1} s_x[k]\cdot e^{j\left(\theta[k]-\hat{\theta}[k]\right)} \qquad \text{(H-14)}$$

Where:

- $s_x[k]$: Partial correlation between the received signal PRN code chips and the generated local replica with a delay between them equal to $\varepsilon_\tau[k]$ at epoch k of the channel x.
- M: Total number of partial correlations composing the complete simulated PRN code

If the number of chips between two generated input samples is an integer number, a partial correlation can be modeled as:

$$s_x[k] = \frac{1}{R}\sum_{r=0}^{R-1} p_x[r] \tag{H-15}$$

Where:

- $p_x[r]$: Integration result of the channel x from the multiplication of the received code chip *r* by the generated replica code chip *r* having a delay between them equal to $\varepsilon_\tau[k]$
- R: Number of chips of the received PRN code used to make the partial autocorrelation

Assuming that the RF/IF block allows a received signal with an infinite bandwidth, the term $p_x[r]$ can be modeled as:

$$p_x[r] = \begin{cases} 1 & c_r^x = c_{r-1}^x \\ 1 - \dfrac{2 \cdot \varepsilon_\tau[r]}{T_c} & c_r^x = -c_{r-1}^x \end{cases} \tag{H-16}$$

Where:

- T_c: Code chip duration
- c_r^x: Code chip *r* amplitude of channel x $\in$ {-1, +1}

This model can be adapted to a real RF/IF block by filtering equation (H-16) by the RF/IF implemented filter. This is not done in this dissertation except for acquisition performance analysis in 0.

In this dissertation, the code delay estimation error $\varepsilon_\tau[k]$ is assumed to be 0; therefore, $s_x[k]$ is always equal to 1, and the scheme of the simulator can be simplified as shown below.

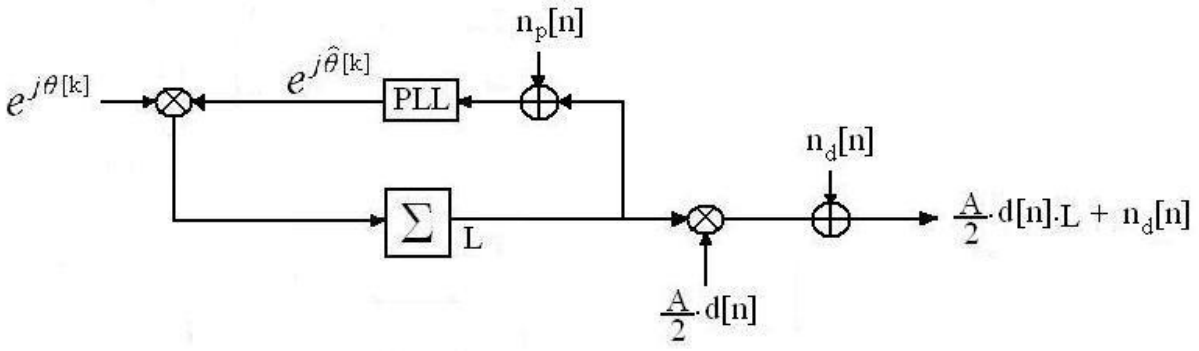

Figure H-4: Simplified PLL simulator scheme for $\varepsilon_\tau[k] = 0$

Where:

$$L = \frac{1}{M}\sum_{k=0}^{M-1} e^{j\left(\theta[k]-\hat{\theta}[k]\right)} \quad \text{(H-17)}$$

One important observation made from Figure H-4 is that the complete correlation between the received signal PRN code waveform and the generated local replica does not have to be simulated.

Another remark can be made about the pilot and data channel noises. In fact, both noises originate from the same thermal noise at the antenna output. This antenna output noise is linearly combined with the pilot PRN code to generate the pilot channel noise and is linearly combined with the data PRN code to generate the data channel noise. Therefore, a more realistic simulation should conduct these linear combinations in order to obtain the channel noises. Additionally, the noises are generated as indicated in annex H.3.1.3, where the Gaussian standard deviation is calculated using equation (H-23). In this equation, the simulator specifies the symbol transmission rate as the rate at which the noise samples are generated.

Moreover, note that the previous presented schemes have assumed that the pilot and data channel are in-phase as for GALILEO E1 OS. Nevertheless, this scheme is still valid for a time-multiplexed pilot and data channels, but some smaller changes should be introduced to adapt the schemes to an in phase-quadrature construction.

Finally, the signal dynamics that affect the signal phase are generated from equation (A-21). Nevertheless, since the PLL implemented in our simulations is of the order 3, and the jerk is considered constant and equal to 1, the model of the incoming signal can be reduced to:

$$\theta(t) = t^3 \quad \text{(H-18)}$$

H.3.1.3. Generation of the additive white or filtered Gaussian noise

In subsection 3.1.2, the main characteristics of the AWGN channel mathematical model have been defined. However, the relations between all the theoretical formulas and real implementation for simulation purposes have not yet been described.

In the next paragraphs, the generation of a white Gaussian noise or a filtered white Gaussian noise is shown. This generation can be used to create the noise of any signal, where the noise fulfils the previous characteristics. This means that the following proposed method can be used to represent either a white Gaussian noise or a filtered white Gaussian noise at the antenna output, at the correlator input, at the correlator output, etc. The only difference is the value of the parameters needed to generate the noise.

To simulate the AWGN channel mathematical model, we have to be able to generate an additive white Gaussian noise or an additive filtered white Gaussian noise. The power of this noise depends on the C/N_0 figure of merit; therefore, the choice made in this Ph.D. is to set the useful signal power to 1, to model the AWG noise as a random Gaussian process and to set the power of this Gaussian process in relation to the desired C/N_0 when the useful signal power is equal to 1. This means that the increase/decrease of the received signal power and the increase/decrease of the noise source power, or in other words the C/N_0 variation, are achieved by modifying the standard deviation value of the random Gaussian process which models the noise.

A Gaussian noise can be modelled as

$$n[k] = \sqrt{\sigma} \cdot g[k] \tag{H-19}$$

$$P_n = \sigma^2 \tag{H-20}$$

Where:

- n[k]: Gaussian variable: N(0, σ^2)
- g[k]: Gaussian variable: N(0, 1)
- P_n : Noise power

Using equations (3-1), (3-3) and (H-20), the random Gaussian variable standard deviation is linked to the C/N_0 value as it is shown below.

$$\sigma = \sqrt{\frac{B}{(C/N_0)}} \tag{H-21}$$

Where:

- B : Channel bandwidth

Finally, assuming that the channel bandwidth is equivalent to the symbol transmission rate (R_S) as it is the case when a matched filter is used at the reception [PROAKISc, 2001], equation (H-21) is equivalent to:

$$R_s = B \tag{H-22}$$

$$\sigma = \sqrt{\frac{R_s}{(C/N_0)}} \tag{H-23}$$

The symbol transmission rate is the parameter which allows the generation of the desired noise. For example, if we want a simulator generating one sample for each code chip at the correlator input, the chosen R_S will be the code chip rate. If we want a simulator generating two samples for each code chip at the correlator input, the chosen R_S will be twice the code chip rate. And if we want a simulator generating one sample for each data symbol at the correlator output, the chosen R_S will be the data symbol transmission rate.

Moreover, the conducted simulations can implement either a baseband signal or an equivalent baseband signal to a pass-band signal. In both cases, all the generated noises are based on equation (H-23). More specifically, the only difference between a baseband signal and an equivalent baseband signal to a pass-band signal is that for the latter option the generated noise is complex whereas for the first option is real. This means that for the latter option, two noises, one for the real component and the other for the imaginary component, are generated. Note that the sum of the noises power has to be equal to $Rs \cdot N_0$ as it has been specified in equation (3-1); therefore the standard deviation of the noise of the real component and the standard deviation of the noise of the imaginary component are calculated using equation (H-23) but with a dividing 2 factor inside the square root. For the base-band signal option, the imaginary component is never used during the demodulation process but the noise of the real component is generated as it was done in the equivalent baseband signal to a pass-band signal case:

$$\sigma = \sqrt{\frac{R_s}{2 \cdot (C/N_0)}} \tag{H-24}$$

Finally, in order to obtain a filtered Gaussian noise, as for example to model the influence of the thermal noise over the PLL phase estimation error, the generation is done following the same process as for an AWG noise but implementing a filter with the desired cut-off frequency: the input of the filter is the AWG noise and the output of the filter is the desired filtered Gaussian noise.

H.3.2. Mobile channel simulator

In this subsection, the simulator implemented to represent the transmission of a GNSS signal through a mobile channel is described. Using the simplified version of the second simulator of an AWGN channel (see Figure H-4), the mobile channel simulator scheme is presented below.

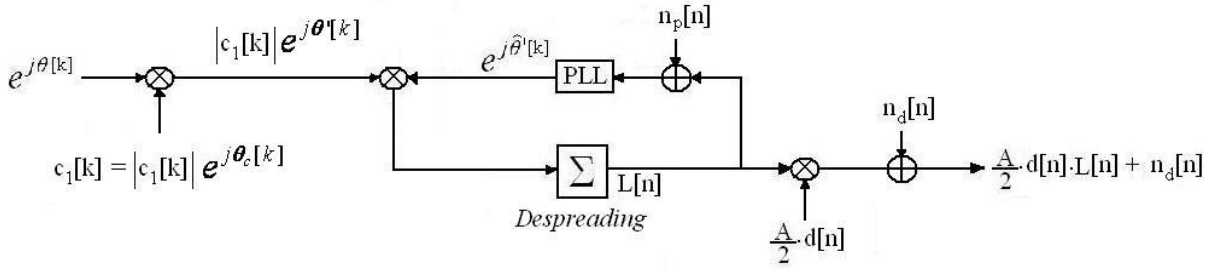

Figure H-5: Mobile channel simulator scheme

Where:

- $\theta'[k]$: Incoming signal carrier phase
- $\hat{\theta}'[k]$: PLL signal carrier phase estimation
- $c_1[k]$: Loo variable at epoch k
- $n_p[n]$: Pilot channel noise at epoch n.
- $n_d[n]$: Data channel noise at epoch n.
- d[n]: Navigation data at epoch n
- k: epoch at chip sampling rate
- n: epoch integration output rate
- L[n]: Correlator output at epoch n

$$L[n] = \frac{1}{M} \sum_{k=0}^{M-1} e^{j(\theta[k] - \hat{\theta}[k])} \tag{H-25}$$

The generation of the $c_1[k]$ coefficient and the generation of the noise is presented next.

H.3.2.1. Generation of the $c_1[k]$ coefficient

The $c_1[k]$ coefficient follows a Loo distribution and thus its generation is accomplished by reproducing this distribution type. Therefore, since a Loo variable is the addition of two

terms, first a variable with a Rayleigh distributed module and a phase uniformly distributed over [0,2π[, and second a Log-normal distributed variable, the generation of a Loo variable is achieved by generating these two variables.

$$c_1[k] = z[k] + w[k] \quad \text{(H-26)}$$

Where:

- z[k]: Log-normal distributed variable
- w[k]: Variable with a module Rayleigh distributed and a phase uniformly distributed over [0,2π[

The generation of a Rayleigh distribution variable, the generation of a log-normal distribution variable and the final $c_1[k]$ generation scheme are presented in the following subsections.

H.3.2.1.1. Generation of the variable with a Rayleigh distributed module and a uniform distributed phase

A variable having a Rayleigh distributed module and uniform over [0,2π[distributed phase can be modeled using two independent Gaussian variables $N(0, \sigma^2)$: one variable representing the real component, and the other variable representing the imaginary component.

$$w[k] = n_1[k] + jn_2[k] \quad \text{(H-27)}$$

Where:

- $n_1[k]$: Normal distribution variable $N(0, \sigma^2)$
- $n_2[k]$: Normal distribution variable $N(0, \sigma^2)$

The generation of the Gaussian variables is made by following the method explained in annex H.3.1.3. This method consists in generating a Gaussian variable with a standard deviation equal to 1, and in multiplying this variable by a coefficient calculated from the received useful signal C/N_0 in order to modify the Gaussian variable variance. However, in this case, the generation of the multiplying coefficient depends on the MP definition. This parameter represents the average multipath power with respect to the LOS signal having a power equal to 0dB. And since the LOS signal amplitude before the multiplication of the $c_1[k]$ coefficient has been set to 1, the simulator is representing the case defined in the definition. Therefore, the standard deviation of the final Gaussian variables, or in other words, the multiplying coefficient, is equal to σ = sqrt(MP/2) in order to impose a power of the multipath component equal to MP. The MP parameter, define in section 3.2.2.3.4, is used in its decimal format.

Finally, one last characteristic remains to be simulated. The samples generated from the previous Gaussian variables are independent among them whereas they should be correlated in time since these samples represent the multipath component evolution along the time. In fact, they should follow an evolution with a changing rate bounded by the channel coherence time. Therefore, in order to simulate this evolution, the independent Gaussian variables are filtered by a low-pass filter with a cut-off frequency equal to the Doppler spread. An adequate filter representing the satellite-mobile user communications is the Butterworth filter [BURZIGOTTI et al., 2008].

$$\left|H_{Butt}(f)\right|^2 = \frac{B}{1 + \left(f / f_{c-off}\right)^{2k}} \quad \text{(H-28)}$$

With:

- B: Constant setting the filter gain to 1
- $f_{c\text{-}off}$: cut-off frequency
- k: filter order

H.3.2.1.2. Log-normal distribution variable generation

The generation of a log-normal distribution variable is based on the generation of a Gaussian distribution variable. In fact, a log-normal variable has the same distribution as a Gaussian variable when a logarithm is applied to the log-normal variable. Therefore, in order to generate a log-normal variable, we are going to generate a Gaussian variable and from it, we are going to generate the log-normal variable applying an exponential to the Gaussian variable.

Therefore, this simulator is going to first generate a Gaussian variable with mean and standard deviation equal to the log-normal α and Ψ parameters. Remember that these parameters define the log-normal variable mean and standard deviation of the desired mobile channel. Second, the simulator is going to divide the normal variable by 20. And third and last, the simulator is going to power by 10 the division result. See equation (H-29).

$$z[k] = 10^{\frac{n[k]}{20}} \tag{H-29}$$

With:

- n[k]: Gaussian variable $N(\alpha, \Psi^2)$
  - α: log-normal mean (dB)
  - Ψ: log-normal standard deviation (dB)

Note that the values of the mean and standard deviation used to generate the lognormal variable are expressed in dB instead of in decimal format.

Finally, the generation of the log-normal variable given in equation (H-29) provides independent log-normal samples. Therefore, these independent samples have to be related to the channel mathematical model presented by Perez-Fontan [PEREZ-FONTAN et al, 1998]. The mathematical model define that two independent log-normal samples are found separated by a distance called the channel correlation distance, L_{Corr}; and this distance altogether with the mobile receiver velocity determine the time between two independent log-normal samples. This means that if the simulator wants to generate more input samples between the two independent log-normal samples, the simulator has to correlate the new generated input samples with the previous independent log-normal samples. The reason is that in doing so, the simulator will obtain input samples following the desired log-normal distribution in time where two samples separated the channel correlation distance will still be independent.

In fact, the rate of the input samples generated by the simulator is defined by the multipath component and thus, the simulator needs to generate more input samples between the two independent log-normal samples. Therefore, the log-normal generation is completed by adding an interpolator which generates from two independent log-normal samples, a series of correlated log-normal samples at the desired sampling rate between the two independent log-normal samples.

The typical interpolator block scheme is given below.

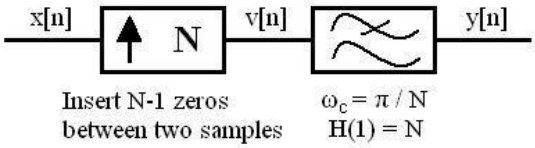

Figure H-6: Generic Interpolator [MARIÑO et al, 1999]

Note that if the required rate of the input samples is too high to be attained with only one interpolator –filter is too restrictive to be successfully implemented-, the simulator can always implement more than one interpolator.

H.3.2.1.3. Generation scheme of the $c_1[k]$ coefficient

The final generation scheme of the $c_1[k]$ coefficient is shown in Figure H-7. This scheme shows all the elements presented in the previous two sections.

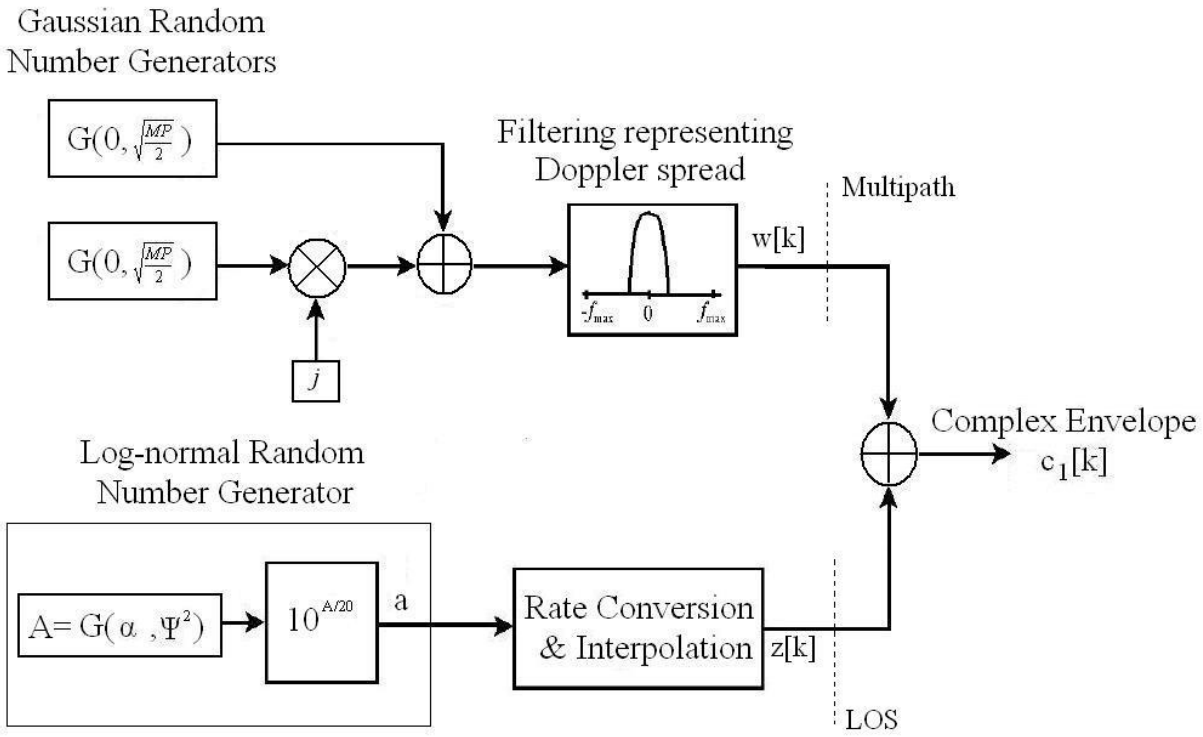

Figure H-7: Generation of the $c_1[k]$ coefficient following a Loo distribution with Butterworth multipath Doppler shaping [BURZIGOTTI et al, 2008]

Finally, note that whereas the log-normal mean and standard deviation are used in dB, the MP parameter is used in its natural form.

H.3.2.2. AWGN generation

The generation of the AWG noise can be ambiguous because, in the mobile channel case, the received signal power varies. Moreover, the simulated signal is the received equivalent baseband signal which is complex. Therefore, the noise also has to be complex.

The complex noise generation consists in generating two white Gaussian variables as indicated in annex H.3.1.3. One Gaussian variable represents the real noise component and the other represents the imaginary one. And the standard deviation of the Gaussian variables is calculated by using equation (H-24). Finally, the C/N_0 used to calculate the standard deviation values is the C/N_0 value when it is not taken into account the possible signal attenuation introduced by the mobile channel, the $c_1[k]$ coefficient. In other words, the C/N_0 used to calculate the noise sigma is the C/N_0 that would be received if the transmission was made through an AWGN channel as has been specified in subsection 3.2.2.4.

Annex I. Code Shift Keying

In this annex, the CSK mapping between the input bits and the transmitted CSK symbol is presented and the calculation of the variance of the number of errors of a packet defined by a given CSK source packets mapping is described.

I.1. CSK mapping between the input bits and the transmitted symbol

In this section, the mapping between the k bits transmitted into a single CSK symbol and the CSK symbol selected to transmit the bits is presented. This mapping is different from the orthogonal CSK or M-ary orthogonal signaling to the bi-orthogonal CSK or M-ary bi-orthogonal signaling.

I.1.1. Orthogonal CSK mapping

The orthogonal CSK signaling method calls the symbol represented by the smallest circular shift of the fundamental code –even if this shift is equal to 0 chips-, the first circularly shifted version of the fundamental code or symbol 0. The transmission of this symbol would be reflected only at the y_0 component of Y vector, if no noise was introduced during the signal transmission through the channel. The other components of the vector should be 0. However, since the channel transmission introduces thermal noise to the received signal, the value of the y_0 component is corrupted and the values of the other Y vector components are not 0. The symbol represented by the second smallest circular shift of the fundamental code is called the symbol 1 and its transmission through a noiseless channel would be reflected only on the y_1 component of the Y vector. The remaining association between circular shifts of the fundamental code, symbols and components of the Y vector is done as explained with the two first symbols. Therefore, the reception of the CSK symbol represented by the largest circular shift of the fundamental code should be inspected at the last component of the Y vector, the y_{M-1} component.

The mapping between the input information bits and the CSK symbols is a natural mapping, with the first bit fed to the demodulator considered as the bit with the smallest weight, and the last bit fed to the demodulator considered as the bit with the biggest weight. The input information bits are denoted as:

$$B^k = \{b_{k-1}, b_{k-2}, \ldots, b_0\} \quad \text{(I-1)}$$

Where:

- B^k: Set of k input bits for a orthogonal CSK symbol
- b_x: Value of the x^{th} bit of the set B^k

For example, if the bits values are B = {b_2=1, b_1=0, b_0=1}, the circularly shifted version of the fundamental code having the sixth smallest shift –symbol 5– is transmitted. Its reception is observed at the y_5 component of the Y vector.

I.1.2. Bi-orthogonal CSK mapping

The mapping of the bi-orthogonal CSK signaling method between the input information bits and the bi-orthogonal CSK symbols is very similar to the orthogonal CSK mapping. The only difference is that for a bi-orthogonal mapping one of the input bits of the set of k bits determines the polarity –or sign– of the CSK symbol to transmit. The exact mapping process is given below.

The first k-1 bits fed to the modulator are used to select the symbol to transmit. The selection is made as if the k-1 bits have to be mapped for an orthogonal CSK signal. Therefore, the component of the Y vector in which the reception of a bi-orthogonal CSK symbol is observed is determined by the first k-1 bits fed to the modulator. The k^{th} bit fed to the modulator is used to select the polarity of the symbol. The mathematical notation for a bi-orthogonal CSK mapping is:

$$R^k = \{\beta, b_{k-2}, b_{k-2}, \ldots, b_0\} \quad \text{(I-2)}$$

$$R^k = \{\beta, B^{k-1}\} \quad \text{(I-3)}$$

Where:

- R^k: Set of k input bits for a bi-orthogonal CSK symbol
- β: Value of the bit controlling the polarity of the bi-orthogonal CSK symbol
- B^{k-1}: Set of k-1 input bits for a orthogonal CSK symbol
- b_x: Value of the x^{th} bit of the set B^{k-1}

Note that a M-ary bi-orthogonal signaling method is equivalent to a M/2-ary orthogonal signaling method where the two polarities –or signs– of the M/2 different symbols are used.

I.2. Calculation of the variance of the number of bit errors of a packet defined by a CSK source packet mapping

The calculation of the variance of the number of errors of a packet defined by a given CSK source packet mapping is achieved by applying the typical formula of the variance of a random variable. This means that we need to find the probability of a packet having X errors after its transmission through the channel, where X goes from 0 to the packet size.

The only difficulty of this proposed calculation of the variance is that the theoretical formula of the probability of a packet having X errors after its transmission through a channel changes its mathematical expression depending on the used CSK source packet mapping. Therefore, the solution employed in this dissertation consists in encoding a program which provides the desired results for each given CSK source packet mapping rather than to determine a specific formula for each CSK source packet mapping. Nevertheless, note that this program does not consist of channel transmission simulations but it consists of applied theoretical formulas.

The probability of a packet having X errors after its transmission through a channel when the packet is transmitted following a given CSK source packet mapping depends on two main parameters. The first parameter is the number of bits mapped by each CSK symbol which belong to the same packet. In this case, in order to simplify the analysis, it has been assumed that the number of bits mapped by a CSK symbol which belong to the same packet is the same for all the packets. For example, if a CSK symbol maps 6 bits, the 4 different configurations are: 6 packets of 1 bit, 3 packets of 2 bits, 2 packets of 3 bits, and 1 packet of 6

bits. The second parameter is the packet size. The number of different probabilities or, in other words, the number of errors X which can have a packet depends on the packet size. And the larger the packet is, the more number of errors has and the larger the variance is.

The method presented in the next subsections follows two steps. First, the probability of having n erroneous bits out of x bits belonging to the same packet when a CSK symbol mapping K bits, $x \leq K$, is transmitted with a CSK symbol probability of error equal to P_M is calculated. Second, the previous probabilities for different values of n which are associated to a single CSK symbol are propagated to the total number of CSK symbols necessary to transmit a complete packet. Note that the parameter x defines the different CSK source packet mappings.

I.2.1. Probability of having n erroneous bits among x bits belonging to the same packet when the transmitted CSK symbol maps K bits

The first step of the calculation of the variance consists in calculating the probability of having n erroneous bits out of x bits belonging to the same packet when the transmitted CSK symbol maps K bits and has a probability of error equal to P_M.

The number of different probabilities to calculate in this first step depends on the number of bits, x, belonging to each packet. For example, the number of probabilities to calculate is 7, from 0 erroneous bits to 6 erroneous bits, when a CSK symbol maps 6 bits and all the bits belong to the same packet; and the number is 4, from 0 erroneous bits to 3 erroneous bits, when a CSK symbol maps 6 bits with 3 bits belonging to a packet and 3 bits belonging to another one. In this last case, the probabilities calculated are valid for both packets.

The method used to calculate the probabilities consists in first finding the probability that an erroneous estimated CSK symbol mapping K bits results into the erroneous determination of the values of n bits, and second, in weighting the previous probability with the probability of demodulating incorrectly a CSK symbol.

The probability is found by dividing the number of cases where the CSK symbol has n errors by the total number of possible cases. A case is defined as a pattern of bit errors generated by the erroneous estimation of the received CSK symbol. Therefore, since a CSK symbol mapping K bits can have from 0 to K erroneous bits when the CSK symbol has been estimated incorrectly, the total number of cases is 2^K-1. The number of cases where the erroneous CSK symbol results into n erroneous bits is calculated by using a combinatorial number.

Nevertheless, the previous method has to be slightly modified in order to take into account that only x bits out of K bits mapped by the CSK symbol belong to the same packet. Therefore, the first analyzed CSK source packet mapping is the mapping where all the bits belong to the same packet since it is the simplest calculation and its calculation method is the method defined previously. This CSK source packet mapping is called the basic CSK source packet mapping from now on. Moreover, the results of the basic CSK source packet mapping are used to simplify the calculations of remaining CSK source packet mappings.

Assuming a CSK symbol carrying K bits, the probabilities of the basic CSK source packet mapping of having n erroneous bits when the CSK symbol is erroneously demodulated are:

$$p\left(0\middle|K_K\right) = 1 - P_M \tag{I-4}$$

$$p\left(n\middle|K_K\right)=P_M\cdot\frac{\binom{K}{n}}{2^K-1}\qquad K\geq n\geq 1 \tag{I-5}$$

Where:

- $p(n|Y_y)$: Probability of having n bit errors belonging to the same packet when transmitting a CSK symbol mapping Y bits out of which y bits belong to this packet.
- P_M: Probability of error of a CSK symbol

Once the probabilities of the basic CSK source packet mapping have been found, the probabilities for CSK source packet mappings defining packets which share bits mapped by the same CSK symbol are calculated. The process consists of two steps.

First, for each different number of bit errors resulting from the erroneous estimate of the CSK symbol and its associated probability -the probability of the main CSK source packet mapping-, the probabilities of the different CSK source packet mappings where the bits mapped by a CSK symbol belong to more than one packet are calculated. More specifically, each probability is calculated by dividing the number of patterns of m bits errors out of which n erroneous bits belong to the analyzed packet by the total number of patterns of m bit errors. For example, assuming that a CSK symbol carries 6 bits where 3 bits belong to one packet and the 3 other bits belong to another packet, some representative particular cases are: if the erroneous estimate of the CSK symbol results in to 1 bit error out of the K bits mapped by the CSK symbol, the particular cases are that the packet of 3 bits has either 0 or 1 errors; however, when the erroneous estimate of the CSK symbol results in to 4 bit errors, the particular cases are that the packet of 3 bits has from 1 to 3 errors. The mathematical expressions are:

$$p\left(n,m\middle|K_x\right)=\begin{cases}p\left(m\middle|K_K\right)\cdot\dfrac{\binom{x}{n}\cdot\binom{K-x}{m-n}}{\binom{K}{m}} & m\geq n\geq\left(m-x\right)\\ 0 & otherwise\end{cases} \tag{I-6}$$

Where:

- $p(n,m|K_x)$: probability of having m erroneous bits out of which n bits belong to the same packet when the CSK symbol mapping K bits out of which x bits belong to the same packet, $m \leq x$, has been erroneously estimated.

The second step consists of adding the probabilities of the particular cases having the same number n of bit errors but found from different numbers of m bit errors. The m bit errors are the result of the erroneous estimate of the CSK symbol.

$$p\left(n\middle|K_x\right)=\left(1-P_M\right)\cdot\delta(n)+\sum_{m=1}^{K}p\left(n,m\middle|K_x\right) \tag{I-7}$$

Where:

- $\delta(n)$: Dirac delta function

Once the calculation has been explained, the mathematical expression is simplified and adapted to include the basic CSK source packet mapping. The general expression is:

$$p\left(n\middle|K_x\right)=\left(1-P_M\right)\cdot\delta(n)+P_M\frac{\binom{x}{n}}{2^K-1}\sum_{w=\delta(n)}^{K-x}\binom{K-x}{w}\qquad n\le x \tag{I-8}$$

Finally, this expression is valid and exact for an orthogonal CSK or MOS signaling method. Nevertheless, this expression can also be used for a bi-orthogonal CSK because, in this case, the expression is a very accurate approximation.

I.2.2. Calculation of the probability error of the bits of a packet from the probability of error of the bits mapped by a CSK symbol

The second step of the calculation of the variance of the number of errors of the packets defined by a given CSK source packets mapping consists in calculating the probability of having a given number of bit errors inside a packet from the probability of having n erroneous bits out of x bits belonging to the same packet when the transmitted CSK symbols maps K bits.

The bits mapped by a single CSK symbol which belongs to one or more packets do not have to be the only bits belonging to these packets. In fact, since the amount of bits which can be mapped by a single CSK symbol is rather small, the most common configuration is to group several CSK symbols in order to generate a packet. This means that the probabilities calculated in the previous section are not sufficient to calculate the variance of the number of bit errors of the packets of a given CSK source packet mapping. More specifically, if the packet has a size of P bits, the probabilities necessary to calculate the variance go from the probability of having 0 bit errors during the complete packet transmission to the probability of having P bit errors during the complete packet transmission. And all these probabilities can be calculated from the probabilities calculated in the previous section by propagating them. The definition of a generic propagation process is given below.

A generic propagation process consists in knowing the bit error probabilities of an element of L bits, denoted as the element-to-propagate, in order to calculate which will be the bit error probabilities of this element if its size is increased to T. The new element of T bits of size is called the propagated-element, where T=L+L' and L' is the size of the element used to extend the original element of L bits of size. The element of L' bits of size is called the propagation-element. Therefore, applying the propagation process, the probability of having n erroneous bits belonging to the same packet when the bits of the packet are carried by one CSK symbol which maps L bits can be extended to the probability of having n' erroneous bits belonging to the same packet when the bits of the packet are carried by two CSK symbols where each one maps L bits. Moreover, note that the element-to-propagate and the propagation-element are not limited to a single CSK symbol but they can be any element of any possible size.

The propagation process is achieved by making a combination of all the different bit error probabilities of the element-to-propagated with all the different bit error probabilities of the propagation-element. More specifically, each bit error probability of the element-to-propagate is multiplied by each bit error probability of the propagation-element. The result is a partial bit error probability of the propagated-element where the number of bit errors associated to this probability is calculated by adding the number of bit errors of this bit error probability of the element-to-propagate to the number of bit errors of this bit error probability of the propagation-element. Finally, once all the multiplications have been made, the partial bit error probabilities of the propagated-element having the same number of bit errors are added together in order to obtain the total bit error probabilities of the propagated-element.

The algorithm implementing the propagation process is given below:

- Element-to-propagate: {Size: L, $P^e=(P_0^e,\ldots, P_L^e)$}
  - P_x^e: Probability of having x bit errors of the element-to-propagate
- Propagation-element: {Size: L', $P^p=(P_0^p,\ldots, P_{L'}^p)$}
  - P_x^p: Probability of having x bit errors of the propagation-element
- Propagated-element: {Size: T, $P^{ep}=(P_0^{ep},\ldots, P_T^{ep})$}
  - P_x^{ep}: Probability of having x bit errors of the propagated-element
- Initializing propagated-element probabilities to 0.
- For i = 0 to L
  - For j = 0 to L'
    - $P_{(i+j)}^{ep} = P_{(i+j)}^{ep} + P_i^e \cdot P_j^p$