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, Optimisation*

Présentée et soutenue par *Pierre NERI*

Le 10/11/2011

Titre : *Use of GNSS signals and their augmentations for Civil Aviation navigation during Approaches with Vertical Guidance and Precision Approaches*

Utilisation des signaux GNSS et de leurs augmentations pour l'Aviation Civile lors d'approches avec guidage vertical et d'approches de précision

---

JURY
Professeur Washington OCHIENG
Professeur Bernd EISSFELLER
Professeur Emmanuel DUFLOS
Docteur Christophe MACABIAU
Laurent AZOULAI

Ecole doctorale : *Mathématiques, Informatique et Télécommunications de Toulouse*

Unité de recherche : Laboratoire de traitement du signal et des télécommunications (LTST)

Directeur(s) de Thèse : *Christophe Macabiau*

Rapporteurs : Pr. Washington OCHIENG, Pr. Bernd EISSFELLER, Pr. Emmanuel DUFLOS

Abstract

The Global Navigation Satellite System (GNSS) has been identified as a very promising mean to provide navigation services to civil aviation users. In recent years, GNSS has become one of the reference mean of navigation thanks to its worldwide coverage area. This global trend can be observed aboard civil aviation aircraft since a majority of them are now equipped with GNSS receivers. However, civil aviation requirements are so stringent in terms of accuracy, integrity, availability and continuity, that GPS standalone receivers cannot be used as a sole mean of navigation.

This result led to the definition of several architectures aiming at augmenting the GNSS constellations. We can distinguish SBAS (Satellite Based Augmentation Systems), GBAS (Ground Based Augmentation Systems), and ABAS (Aircraft Based Augmentation Systems). In fact, this PhD investigates the behaviour of the position error at the output of receiver architectures which have been identified as very promising for civil aviation applications.

A state of the art study of the current civil aviation requirements and of the on-going studies at standardization level concerning future civil aviation receivers is first presented. This section was used to select two particularly interesting solutions. In fact, we decided to focus on the two following receivers:

- combined dual frequency E1/E5 GPS-GALILEO and RAIM (Receiver Autonomous Integrity Monitoring) for APV and CAT-I operations
- GPS L1 C/A and GBAS for CAT-I and CAT-II/III precision approaches.

A description of the different GNSS signals currently available or that will be available in the future thanks to the modernization of GPS as well as the implementation of the GALILEO constellation is given.

The different models used to represent the errors affecting the GNSS signals as well as the pseudorange measurements made by GNSS receivers are then described. First we present the error models which are intended to represent the impact of the main sources of errors on GNSS receivers measurements. These sources of errors are propagation errors due to ionosphere and troposphere, multipath, interference, noise, satellite clock error, satellite position estimation error. Then, we present the pseudorange measurement error models used for the combined GPS-GALILEO and RAIM receiver model as well as for the GPS L1 C/A and GBAS receiver model. Finally, we present the structure and the different modules of the GNSS receiver simulator developed in the course of this PhD work and in particular we give a description of the receiver signal processing, the position computation as well as the integrity monitoring.

Concerning the combined GPS-GALILEO and RAIM receiver, we propose a microscopic analysis of the behaviour of the output NSE which is based on simulations made using the previous simulator. Particular attention is dedicated to the study of the position steps induced by nominal GNSS constellation changes including rising and setting GPS and GALILEO satellites.

Simulation results show that this position solution has very good performance mainly due to the combination of two GNSS constellation. However, the impact of constellation changes has been highlighted, as the induced vertical position error steps can reach amplitudes of around 2.0 meters. Moreover, we also observe steps on the vertical protection levels of more than 4.0 meters. To deal with these steps, an algorithm is defined so as to avoid their effect during the most critical part of the

approaches, between the Final Approach Fix and the touchdown on the runway. It consists in predicting satellites which will probably rise or set during this final approach and to automatically exclude them from the position solution when passing the FAF. This algorithm allows avoiding every NSE step due to constellation changes without degrading the position solution performance thanks to the high number of satellites available.

We finish this report with the presentation of the outcomes of our study on the GPS L1 C/A and GBAS receiver. In fact, the approach is different from the one adopted for the RAIM receiver since we propose here a GBAS NSE model for autoland simulations for CAT-I and CAT-II/III autoland capability demonstrations to certification authorities. In this context, the analysis is based on a previously published model concerning GBAS CAT-I autoland simulations which is completely reviewed and analysed. The conclusion is that the methodology used in this model is innovative and well adapted to autoland simulations. However, a number of weaknesses are identified. In particular, the state of the art model is based on a number of distributions derived statistically during simulations which were based on pseudorange measurement models which do not reflect latest evolutions of GBAS requirements and standard.

Thus, an alternate model is proposed in this report for GBAS CAT-I autoland simulations. This model is defined using new experimental distributions which were computed in simulations on the basis of updated models for the simulation of the GPS constellation and the GPS pseudorange measurements. We thus obtain a complete GBAS CAT-I NSE model. The final objective of our study has been to propose evolutions of the model in order to be able to use it for GBAS CAT II-III autoland simulations.

The main contributions of this thesis are the development of a complete GNSS receiver simulator including multi-constellation and multi-frequency position solution, the study of the temporal behaviour of a combined GPS/GALILEO and Weighted Least Square Residuals (WLSR) RAIM NSE as well as the assessment of the impact of constellation changes, the design of an algorithm to prevent the impact of constellation changes on combined GPS/ GALILEO and WLSR RAIM position solution during approaches. We can also mention the complete analysis of a state of the art GBAS GAST-C (GBAS Approach Service Type for CAT-I precision approaches as defined by ICAO) NSE model for GBAS CAT-I autoland simulations, the proposal of an alternate model for GBAS CAT-I autoland simulations, and the proposal of a GBAS GAST-D (GBAS Approach Service Type for CAT-II/III precision approaches as defined by ICAO) NSE model for GBAS CAT-II/III derived the previous one.

Résumé

La navigation par satellite, Global Navigation Satellite System, a été reconnue comme une solution prometteuse afin de fournir des services de navigation aux utilisateurs de l'Aviation Civile. Ces dernières années, le GNSS est devenu l'un des moyens de navigation de référence, son principal avantage étant sa couverture mondiale. Cette tendance globale est visible à bord des avions civils puisqu'une majorité d'entre eux est désormais équipée de récepteurs GNSS. Cependant, les exigences de l'Aviation Civile sont suffisamment rigoureuses et contraignantes en termes de précision de continuité, de disponibilité et d'intégrité pour que les récepteurs GPS seuls ne puissent être utilisés comme unique moyen de navigation.

Cette réalité a mené à la définition de plusieurs architectures visant à augmenter les constellations GNSS. Nous pouvons distinguer les SBAS (Satellite Based Augmentation Systems), les GBAS (Ground Based Augmentation Systems), et les ABAS (Aircraft Based Augmentation Systems). Cette thèse étudie le comportement de l'erreur de position en sortie d'architectures de récepteur qui ont été identifiées comme étant très prometteuses pour les applications liées à l'Aviation Civile.

Pour commencer, nous présentons une revue de l'état de l'art concernant les exigences de performance de l'aviation civile ainsi que les études actuelles sur les futurs récepteurs GNSS pour l'aviation civile dans les différents groupes de standardisation concernés. A l'issue de cette revue, nous avons pu sélectionner deux solutions particulièrement intéressantes, que nous avons décidé d'étudier pendant cette thèse :

- Les récepteurs combinés GPS-GALILEO bi-fréquence augmentés par un algorithme RAIM (Receiver Autonomous Integrity Monitoring) pour les approches APV et CAT-I
- Les récepteurs GPS L1 C/A augmentés par un système GBAS pour les approches de précision CAT-I et CAT-II/III

Les différents signaux GNSS qui sont actuellement disponibles ainsi que ceux qui le seront dans le futur suite à la modernisation de la constellation GPS et le lancement de la constellation GALILEO sont ensuite décrits.

Nous poursuivons avec la présentation des modèles utilisés pour simuler les différentes erreurs qui affectent les signaux GNSS ainsi que la présentation des modèles de mesures de pseudodistance réalisées par les récepteurs GNSS. Tout d'abord, nous commençons par établir les modèles qui servent à représenter l'impact induit par les principales sources d'erreurs sur les mesures GNSS qui sont principalement les erreurs de propagation dues à la traversée de la ionosphère et de la troposphère, les multi-trajets, les interférences, le bruit, les erreurs d'horloge satellite ainsi que les erreurs d'estimation de position des satellites. Nous décrivons ensuite les modèles d'erreurs des mesures de pseudodistance utilisés respectivement dans le modèle de récepteur combiné GPS-GALILEO augmenté par un algorithme RAIM et dans le modèle de récepteur GPS L1 C/A augmenté par le GBAS. Finalement, nous introduisons l'architecture ainsi que les différents modules qui constituent le simulateur de récepteur GNSS qui a été développé lors de notre étude, ce qui inclue en particulier les modules de traitement du signal, de calcul de position et de contrôle d'intégrité.

En ce qui concerne le récepteur combiné GPS-GALILEO and RAIM, nous proposons ici une étude du comportement de la NSE (Navigation Sensor Error) en sortie du récepteur qui est basée sur les résultats obtenus à partir du simulateur évoqué précédemment. Nous nous concentrons en particulier

sur une étude des sauts de position induits par les changements de constellation nominaux qui incluent les levers ainsi que les couchers de satellite GPS et GALILEO.

Les résultats de simulation montrent que ce type de solution de position a de très bonnes performances, principalement grâce à la combinaison de deux constellations GNSS différentes. Cependant, nous avons pu illustrer l'impact des sauts de constellation puisque nous avons observé des sauts d'erreur de position pouvant atteindre des amplitudes de l'ordre de 2.0 mètres. De plus, nous avons également constaté des sauts de rayon de protection vertical allant jusqu'à 4 mètres. Afin de ne pas subir ce genre de phénomène, nous proposons un algorithme permettant d'éviter les sauts pendant la phase la plus critique des approches, que nous avons définie comme allant du FAF (Final Approach Fix) jusqu'au toucher des roues. Le principe de cet algorithme est de prédire les satellites qui vont probablement se lever ou se coucher lors de cette approche finale, et de les exclure automatiquement du calcul de position au passage du FAF. Cet algorithme permet ainsi d'éviter l'ensemble des sauts de position dus aux changements de constellation sans pour autant dégrader la performance de positionnement grâce au nombre important de satellites disponibles.

Nous terminons ce rapport en présentant les résultats de notre étude concernant les récepteurs GPS L1 C/A augmentés par un système GBAS. L'approche GBAS CAT-I dont nous proposons une analyse complète. La conclusion de cette revue est que la méthode utilisée dans ce modèle est innovante et tout à fait adaptée aux simulations autoland. Cependant, un certain nombre de limitations sont relevées. En particulier, ce modèle est basé sur des distributions obtenues par simulation qui sont basées sur des modèles d'erreur de mesures de pseudodistance qui ne prennent pas en compte les dernières évolutions des exigences et des standards GBAS.

C'est pourquoi un modèle alternatif est proposé dans ce rapport pour les simulations autoland GBAS CAT-I. Ce modèle repose sur de nouvelles distributions qui ont été obtenues en simulation sur la base de modèles mis à jour en ce qui concerne la simulation de la constellation GPS et des mesures de pseudodistance GPS L1 C/A. La finalité de cette étude est de proposer des évolutions de ce modèle afin de pouvoir l'utiliser également dans le cadre de simulations autoland GBAS CAT II-III.

Les contributions principales de cette thèse sont le développement d'un simulateur de récepteur GNSS complet incluant la possibilité de simuler des solutions de position multi-constellation et multifréquences, l'étude du comportement temporel de la NSE d'un récepteur combiné GPS-GALILEO augmenté par un algorithme RAIM des moindres carrés pondérés (WLSR RAIM), ainsi que l'analyse de l'impact des changements de constellation et la définition d'un algorithme permettant d'éviter ce genre d'impact pendant les parties critiques des approches. Il faut également mentionner l'analyse complète d'un modèle publié de NSE GBAS GAST-C (GBAS Approach Service Type pour les approches de précision CAT-I comme défini par l'OACI), la proposition d'un modèle alternatif pour les simulations autoland GBAS CAT-I ainsi que la proposition d'un modèle de NSE GBAS GAST-D (GBAS Approach Service Type pour les approches de précision CAT-II/III comme défini par l'OACI) dérivé du modèle précédent.

Acknowledgements

Table of contents

List of Figures

13

List of Tables

Chapter 1: Introduction

1.1 Motivation

Since many years, civil aviation has identified GNSS as an attractive mean to provide navigation services. In fact, in recent years, GNSS slowly became one of the reference mean of navigation for civil aviation users due to its wide coverage area. Since a majority of commercial aircraft is now equipped with GNSS receivers, aeronautical rules have evolved so as to switch from classical radionavigation to RNAV or Area Navigation in order to be less dependent from ground facilities.

This evolution was possible using the US Global Positioning System which currently broadcasts in particular the GPS L1 C/A signal. This signal is of particular interest for civil aviation since it is emitted in an Aeronautical Radio Navigation Services (ARNS) band which is reserved for aeronautical applications and protected from interferences.

However, since civil aviation requirements can be very stringent in terms of accuracy, integrity, availability and continuity, GPS standalone receivers cannot be used as a sole mean of navigation. This fact has led the ICAO to define and develop standard augmentation systems to correct the GPS L1 C/A and to monitor the quality of the received Signal-In-Space (SIS). Different solutions exist depending on where and how the augmentation is implemented. We can distinguish SBAS (Satellite Based Augmentation Systems), GBAS (Ground Based Augmentation Systems), ABAS (Aircraft Based Augmentation Systems). Each system can allow fulfilling applicable SIS requirements up to a certain level. Two types of ABAS systems can be distinguished: Receiver Autonomous Integrity Monitoring (RAIM) when only GNSS information is used, Aircraft Autonomous Integrity Monitoring (AAIM) when information from other on-board sensors is also used. It is important to notice that GBAS and SBAS provide the possibility to correct GPS L1 C/A signal pseudorange measurements by transmitting adequate corrections to the receivers as well as monitoring the integrity of the signal. On the other hand, ABAS mainly permits to monitor the integrity of GPS L1 C/A. In addition, improvements of the accuracy, availability and continuity of the position can be obtained in the case of AAIM thanks to the integration in the position solution of external sensors measurements.

In fact, the most demanding phases of flight in terms of SIS performance are the approaches which have been categorized by the ICAO as following:

- Non precision approaches (NPA)
- Approaches with vertical guidance (APV)
- Precision approaches (CAT I, CAT II, CAT III a/b/c)

Current GNSS standards published by ICAO cover every phase of flight from Oceanic down to CAT I precision approaches thanks to GBAS. The aim of this study is to investigate and characterize the behaviour of the position error of promising positioning solutions for civil aviation users in the context of the on-going development of new GNSS constellations as well as new GNSS signals. In fact, during approaches and in particular precision approaches, the behaviour of the GNSS NSE (Navigation Sensor Error) has to be precisely known so that autoland guidance laws can be adapted to anticipate and cancel the impact of undesired or spurious position errors.

Airbus as an aircraft manufacturer is particularly interested in increasing its knowledge of the impact of the different errors affecting pseudorange measurements on the NSE and in evaluating the benefits

brought by future evolutions of the GNSS systems which is why a partnership was created between Airbus and ENAC to investigate these subjects and which resulted in the co-funding of this PhD.

In this context, we decided to focus our attention on two promising position solutions which are GPS L1 C/A and GBAS as well as GPS/GALILEO and RAIM.

It was felt that no software models were already available for GPS L1 C/A and GBAS and GPS/GALILEO and RAIM and suited to the needs of Airbus and in particular the needs for evolution of the models, and the need to increase the knowledge through participation to the development.

GBAS is composed of two main elements:

- The ground station which includes an active VDB (VHF Broadcast) transmitter and several reference receivers which location is precisely known. This station is able to compute differential pseudorange corrections and to monitor the quality of the GPS SIS.
- The airborne receiver which includes the capability to receive and process the GBAS SIS in addition to GPS SIS. Using the data sent by the ground station, the user receiver is able to correct its own measurements but also to exclude some of them and to compute protection levels which are an evaluation of the confidence that the user can have in the final position solution.

GBAS is currently foreseen as an important source of innovation for civil aviation since it has already been certified for CAT-I precision approaches and may allow reaching ICAO requirements down to CAT-II/III minima, then providing an alternative to classical landing system ILS. This possibility is actively investigated and ICAO and Industry standardization bodies are currently deriving requirements for GBAS CAT II-III.

However, mandatory regulations for the certification of CAT II/III Autoland capability of a navigation mean require numerous simulations to assess statistically the aircraft capability to autoland. Therefore, it is necessary to identify the GBAS GLS (GPS Landing System) behaviour with sufficient fidelity, taking into account errors affecting the SIS performance. The outcomes of these simulations are intended to feed an autoland simulator dedicated to airworthiness assessment of aircraft guidance laws.

A model has previously been proposed in the past for CAT I GBAS autoland simulations but the evolution of CAT II/III requirements and the lack of information on the validation process used made it necessary to analyze and improve if necessary the existing model. The aim of the study on GBAS is then to derive the rationale for the architecture of the state of the art model, to propose possible improvements and to develop necessary evolutions to extend the use of this model to CAT II/III autoland simulations. The expected outcome is then to obtain a simulation tool to be used in the frame of certification of GBAS CAT II/III.

RAIM and GPS/GALILEO is the other position solution which is studied. Today RAIM and/or AAIM are commonly used to provide integrity monitoring for phases of flight down to Non Precision Approaches using GPS L1 C/A measurements and the associated redundancy. However, current performance associated to GPS L1 C/A and GPS constellation is not sufficient to meet civil aviation requirements for more stringent phases of flight and in particular, the vertical requirements associated to APV and CAT-I.

The introduction of new signals and constellations - such as GPS L5 and GALILEO for example – will significantly increase the number of available signals and satellites, the quality of the

measurements as well as the quality of constellation geometries. Thus, RAIM may provide a simple mean to monitor the quality of the SIS and to reach more stringent phases of flights such as APV or CAT-I operations.

This possibility has been investigated by civil aviation community during recent years as in [MARTINEAU, 2008], [LEE, 2007] and [WALTER, 2008]. RAIM is now foreseen as an interesting candidate to provide integrity monitoring for CAT-I. Different algorithms have been studied and their performance in terms of availability has been published. It results that RAIM alone does not seem to provide sufficient protection for CAT-I approaches. New possibilities to extend RAIM to APV and CAT-I are under study and it can be reasonably assumed that RAIM using GPS and GALILEO constellations could be used in the near future (Advanced RAIM) [GEAS, 2010].

Therefore, it appears necessary to address the specific phenomena that will result from the combination of two different constellations in one positioning solution. The interest of this study was thus to provide a microscopic analysis of the temporal behaviour of GPS and GALILEO NSE and RAIM, for CAT-I type of operations. Indeed, the use of different types of satellites, the constellation changes, potential loss of frequencies may imply unexpected behaviour of NSE. The impact of the introduction of this new positioning solution has to be investigated so as to limit unwanted effects for aircraft flight control systems which use the GNSS computed position. A particular attention has been paid to constellation changes. So as to make this study possible, a GPS/GALILEO and RAIM receiver model has been developed.

1.2 Original Contribution

This section gives a brief insight in the original contributions obtained thanks to this study. Each point is extensively described all along this document:

- Development of a GNSS receiver simulator including multi-constellation and multi-frequency position solution.
- Study of the temporal behaviour of GPS/GALILEO and LSR RAIM NSE and assessment of the impact of constellation changes.
- Design of an algorithm to prevent the impact of constellation changes on GPS/GALILEO and LSR RAIM position solution during final approaches.
- Analysis of a state of the art GBAS GAST-C NSE model for GBAS CAT-I autoland simulations
- Proposal of an alternate model for GBAS CAT-I autoland simulations
- Proposal of a GBAS GAST-D NSE model for GBAS CAT-II/III based on the previous so as to reflect evolution of civil aviation requirements concerning GBAS GAST-D service.

1.3 Dissertation organization

The following report is organized as follows.

First, Chapter 2 gives an overview of requirements applicable to GNSS use for navigation of civil aviation aircraft. It describes the different categories of phases of flight with an emphasis on approaches. The concept of Performance Based Navigation (PBN) is then introduced. The operational criteria of ICAO for GNSS based navigation are defined and the associated Signal In Space requirements are reminded for each phase of flight. Finally, a review of the different categories of GNSS receivers classified by RTCA and agreed by EUROCAE is presented which is followed by the presentation of initial studies led by EUROCAE on possible multi-constellation multi-signals GNSS receivers.

The state of the art of GNSS signals structure and the associated signal processing is presented in Chapter 3. Thus, the different GNSS signals currently available for civil aviation or which will be operational with the introduction of new GPS signals as well as the GALILEO constellation are described. Classical architectures of the signal receiver for processing these signals are then given.

The purpose of Chapter 4 is to highlight the different pseudorange measurement models considered in this study as well as the receiver simulator developed and used to derive the results obtained in the course of this work. First, models used to represent the errors affecting the pseudorange measurements are developed. Then, the pseudorange measurement error models applicable to civil aviation GNSS receivers for position computation are reminded in the case of RAIM and GPS/GALILEO and GBAS GAST-C and GAST-D with GPS L1 C/A. The next sections present the architecture and the implementation of the multi-constellation and multi-signal GNSS receiver simulator which includes in particular the receiver signal processing, the position computation as well as the integrity monitoring.

Chapter 5 gathers the results obtained during simulation concerning the RAIM and GPS/GALILEO combination model and more precisely the behaviour of the NSE as well as the impact of constellation changes on this NSE.

Chapter 6 presents the outcomes of our study on a GBAS GAST-D NSE model for CAT II/III autoland simulations. This chapter thus addresses a review and a mathematical analysis of an existing GBAS GAST-C NSE model for CAT-I autoland simulations. It then presents our proposal of an equivalent GBAS GAST-C NSE model for CAT-I autoland simulations which takes into account the conclusions derived from the simulation run in the frame of this study. The proposed model is moreover enhanced so as to provide a solution for simulating GBAS GAST-D NSE for CAT II/III autoland simulations by taking into account recent evolutions of GBAS CAT II/III standards.

Chapter 2: Civil Aviation Requirements

2.1 Civil Aviation Authorities

So as to be authorized for use aboard aircraft, navigation equipments have to fulfil a number of requirements so as to ensure their capability to perform their function. The aim of this section is to briefly present and describe the main organisations which issue these requirements.

2.1.1 International Civil Aviation Organization (ICAO)

The International Civil Aviation Organization (ICAO) is the agency of the United Nations, which codifies the principles and techniques of international air navigation and fosters the planning and development of international air transport to ensure safe and orderly growth. The ICAO Council adopts standards and recommended practices (SARPs) concerning air navigation, prevention of unlawful interference, and facilitation of border-crossing procedures for international civil aviation. In addition, the ICAO defines the protocols for air accident investigation followed by transport safety authorities in countries signatory to the Convention on International Civil Aviation, commonly known as the Chicago Convention [ICAO, 2008].

In particular, The International Civil Aviation Organization (ICAO) is responsible for establishing the standards for radio navigation aids, including the ones concerning GNSS. They are mainly defined in the Annex 10 to the Convention on International Civil Aviation.

2.1.2 RTCA, Inc.

RTCA, Inc. is a private, not-for-profit Corporation that develops consensus-based recommendations regarding communications, navigation, surveillance, and air traffic management (CNS/ATM) system issues. RTCA functions as a Federal Advisory Committee. Its recommendations are used by the Federal Aviation Administration (FAA) as the basis for policy, program, and regulatory decisions and by the private sector as the basis for development, investment and other business decisions [RTCA, 2010].

In particular, the working group SC-159 of RTCA focuses on GNSS systems. His task is to develop minimum standards that form the basis for FAA approval of equipment using GPS as primary means of civil aircraft navigation.

According to [RTCA, 2006], RTCA's objectives include but are not limited to:

- coalescing aviation system user and provider technical requirements in a manner that helps government and industry meet their mutual objectives and responsibilities;
- analysing and recommending solutions to the system technical issues that aviation faces as it continues to pursue increased safety, system capacity and efficiency;
- developing consensus on the application of pertinent technology to fulfill user and provider requirements, including development of minimum operational performance standards for electronic systems and equipment that support aviation; and
- assisting in developing the appropriate technical material upon which positions for the International Civil Aviation Organization and the International Telecommunication Union and other appropriate international organizations can be based.

Different RTCA publications have been used during this PhD the main one being:

- DO-229D – *Minimum Operational Performance Standards for Global Positioning System/Wide Area Augmentation System Airborne Equipment.*
- DO-245A – *Minimum Aviation System Performance Standards for Local Area Augmentation System*

- DO-253C – *Minimum Operational Performance Standards for GPS Local Area Augmentation System Airborne Equipment*

2.1.3 EUROCAE

The European Organization for Civil Aviation Equipment (EUROCAE) is a non-profit making organization which was formed to provide a European forum for resolving technical problems with electronic equipment for air transport. EUROCAE deals exclusively with aviation standardization (Airborne and Ground Systems and Equipments) and related documents as required for use in the regulation of aviation equipment and systems [EUROCAE, 2010]. Its work programme is principally directed to the preparation of performance specifications and guidance documents for civil aviation equipment, for adoption and use at European and world-wide levels.

EUROCAE is composed of manufacturers, service providers, national and international aviation authorities as well as users (airlines, airports). EUROCAE can be considered as the equivalent of RTCA for Europe.

To develop EUROCAE documents, EUROCAE organizes Working Groups (WG). In particular, the WG-62 is responsible for the preparation of aviation use of GALILEO and the development of initial Minimum Operation Performance Specifications (MOPS) for the first generation of GALILEO airborne receivers.

The use of dual constellation receivers (GPS+GALILEO) should be standardized jointly by EUROCAE and RTCA in a future MOPS.

Now that the different organisations involved in the elaboration of the standards related to the use of GNSS systems for civil aviation have been presented, we will present the standardized phases of flight for civil aircraft flights.

2.1.4 FAA and EASA

The official authorities which publish mandatory requirements to be respected by aircraft manufacturers and airliners to fly an aircraft are the FAA and EASA. The FAA is an agency of the United States Department of Transportation. The EASA can be considered as the equivalent structure for the European Commission. Their main goal is to ensure the safety of the civil aviation air traffic.

Many of their publications are based on or refer to the standardization publications emitted by the previous organisations.

We can mention here publications which are directly related to this study and which are the airworthiness criteria for landing operations. These can be found in FAA Advisory Circular AC 120-28D [FAA, 1999] and EASA CS AWO [EASA, 2003].

2.2 Phases of Flight

2.2.1 Categories of flight phases

The flight of an aircraft consists of six major phases [CICTT, 2006]:

- Take-Off: From the application of takeoff power, through rotation and to an altitude of 35 feet above runway elevation or until gear-up selection, whichever comes first.
- Departure: From the end of the Takeoff sub-phase to the first prescribed power reduction, or until reaching 1000 feet above runway elevation or the VFR pattern (Visual Flight Rules), whichever comes first.

- Cruise: Any level flight segment after arrival at initial cruise altitude until the start of descent to the destination.
- Descent:
  - Instrument Flight Rules (IFR): Descent from cruise to either Initial Approach Fix (IAF) or VFR pattern entry.
  - Visual Flight Rules (VFR): Descent from cruise to the VFR pattern entry or 1000 feet above the runway elevation, whichever comes first.
- Final Approach: From the FAF (Final Approach Fix) to the beginning of the landing flare.
- Landing: Transition from nose-low to nose-up attitude just before landing until touchdown.

As we are particularly interested in Approaches operations we will give a more detailed description in the following section.

2.2.2 Approaches

Categories of aircraft approaches are defined according to the level of confidence that can be placed by the pilot into the system he is using to help him land the plane safely. Approaches are divided in two main segments: the aircraft first follows the indication provided by the landing system, and then the pilot takes over in the final part and controls the aircraft using visual outside information. As the reliability of the aircraft, the crew and the landing system increases, the height of the aircraft over the ground at the end of the interval of use of the information provided by the system can be decreased [MACABIAU, 1997].

Three classes of approaches and landing operation have been defined by the ICAO in the Annex 6 [ICAO, 2001] and are classified as follows:

- Non Precision Approaches and landing operations (NPA): an instrument approach and landing which utilizes lateral guidance but does not utilize vertical guidance.
- Approaches and landing operations with vertical guidance (APV): an instrument approach and landing which utilizes lateral and vertical but does not meet the requirements established for precision approach and landing operation.
- Precision approaches and landing operation: an instrument approach and landing using precision lateral and vertical guidance with minima as determined by the category of operation.

The figure below presents the different phases of flight - and in particular the different type of approaches - along with the type of GNSS augmentations which allow conducting navigation operations for civil aviation during the corresponding phase of flight.
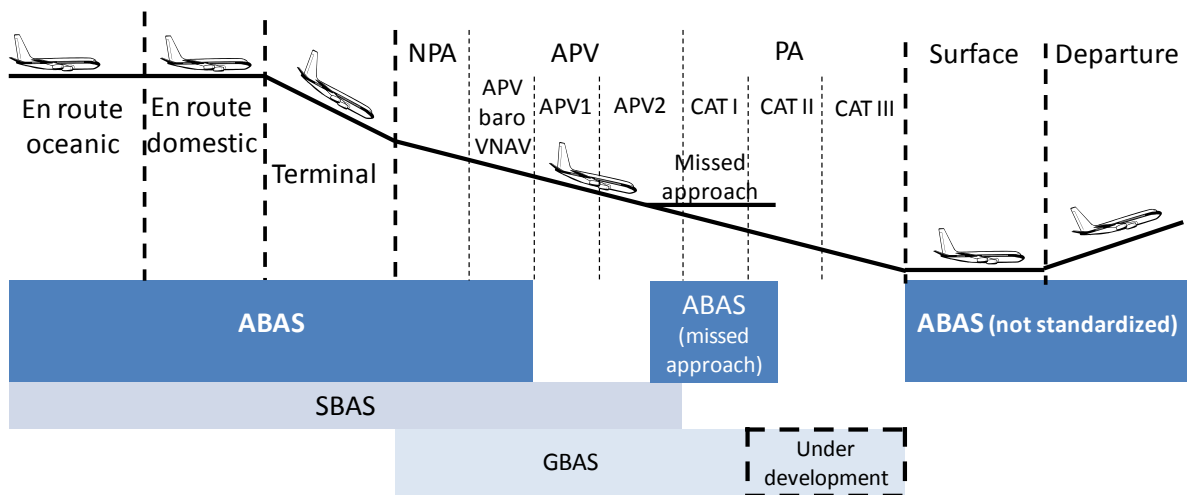
Figure 1: Phases of flight and GNSS augmentations [MONTLOIN, 2011]

The approaches can be defined using three different operational parameters which are the Decision Height (DH), the Distance of Visibility and the Runway Visual Range (RVR). These parameters are defined as follows [CABLER, 2002]:

Decision Height (DH) is the minimal height above the runway threshold at which as missed approach procedure must be executed if the minimal visual reference required in order continuing the approach has not been established.

Distance of Visibility is the greatest distance, determined by atmospheric conditions and expressed in units of length, at which it is possible with unaided eye to see and identify, in daylight a prominent dark object, and at night a remarkable light source.

Runway Visual Range (RVR) is the maximum distance in the landing direction at which the pilot on the centre line can see the runway surface markings, runway lights, as measured at different points along the runway and in particular in the touchdown area.

Category			Minimum Descent Altitude (MDA) Minimum Descent Height (MDH) Decision Altitude (DA) Decision Height (DH)	Visual requirements
NPA			MDA $\geq$ 350 ft	Depending on the airport equipment
APV			DA $\geq$ 250 ft	
LPV 200			DH $\geq$ 60 m (200ft)	
Precision Approaches	CAT – I		DH $\geq$ 60 m (200ft)	Visibility $\geq$ 800 m
				Or RVR $\geq$ 550 m
	CAT – II		30 m (100ft) $\leq$ DH $\leq$ 60 m (200ft)	RVR $\geq$ 300 m
	CAT – III	A	0 m $\leq$ DH $\leq$ 30 m (100ft)	RVR $\geq$ 175m
		B	0 m $\leq$ DH $\leq$ 15 m (50ft)	50 m $\leq$ RVR $\leq$ 175 m
		C	DH = 0 m	RVR = 0 m

Table 1: Decision heights and Visual requirements [ICAO, 2001]

2.3 Performance Based Navigation

The Performance Based Navigation (PBN) concept specifies that aircraft RNAV system performance requirements be defined in terms of the accuracy, integrity, availability, continuity and functionality, which are needed for the proposed operations in the context of a particular airspace concept. The PBN concept represents a shift from sensor-based to performance-based navigation. Performance requirements are identified in navigation specifications, which also identify the choice of navigation sensors that may be used to meet the performance requirements. These navigation specifications are defined at a sufficient level of detail to facilitate global harmonization by providing specific implementation guidance for States and operators [ICAO, 2008].

PBN offers a number of advantages over the sensor-specific method of developing airspace and obstacle clearance criteria:

- Reduces the need to maintain sensor-specific routes and procedures, and their associated costs
- Avoids the need for developing sensor-specific operations with each new evolution of navigation systems, which would be cost-prohibitive
- Allows for more efficient use of airspace (route placement, fuel efficiency and noise abatement)
- Clarifies how RNAV systems are used
- Facilitates the operational approval process for operators by providing a limited set of navigation specifications intended for global use.

The concept of PBN relies on RNAV systems and Required Navigation Performance (RNP) procedures. Here are reminded the definitions of these key terms [ICAO, 2008]:

Area Navigation (RNAV): A method of navigation which permits aircraft operation on any desired flight path within the coverage of station-referenced navigation aids or with the limits of the capability of self-contained aids, or a combination of these.

Area Navigation Equipment: Any combination of equipment used to provide RNAV guidance.

Required Navigation Performance (RNP) Systems: An area navigation system which supports on-board performance monitoring and alerting.

Required Navigation Performance (RNP): A statement of the navigation performance necessary for operation within a defined airspace.

According to [ICAO, 2008] the required navigation performance can be defined by the Total System Error (TSE) which is illustrated in Figure 2:
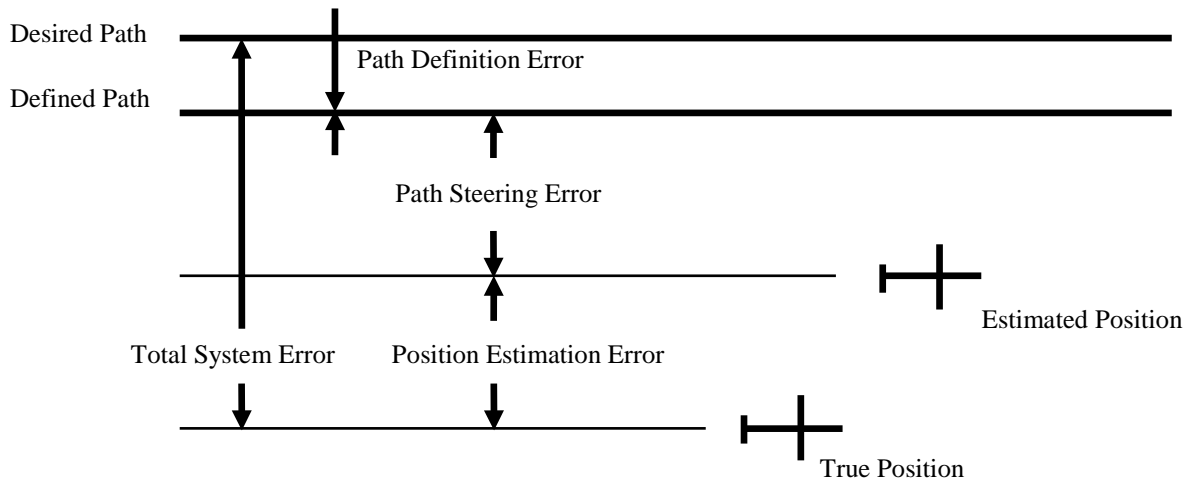
Figure 2: Total System Error [ICAO, 2008]

Defined Path: The output of the path definition function.

Desired Path: The path that the flight crew and air traffic control can expect the aircraft to fly, given a particular route or leg or transition.

Estimated Position: The output of the position estimation function.

Path Steering Error (PSE): The distance from the Estimated Position to the Defined Path. The PSE includes both FTE and display error.

Flight Technical Error (FTE): The accuracy with which the aircraft is controlled as measured by the indicated aircraft position with respect to the indicated command or desired position.

Path Definition Error (PDE): The difference between the Defined Path and the Desired Path at a specific point.

Position Estimation Error (PEE): The difference between the true position and Estimated Position.

Total System Error (TSE): The difference between true position and Desired Position. This error is equal to the vector sum of the Path Steering Error, Path Definition Error, and Position Estimation Error.

To support the PBN concept a reduced set of performance criteria have been identified so as to characterize the performance of navigation equipments. These criteria are presented in the next section.

2.4 Operational Criteria for navigation performance

Operational requirements for GNSS based navigation are defined using four criteria which are accuracy, availability, continuity and integrity. The associated definitions which are reminded here can be found in [ICAO, 2006].

2.4.1 Accuracy

Accuracy is the degree of conformance between the estimated or measured position and/or velocity of a platform at a given time and its true position and/or velocity. In order to characterize the accuracy on the estimated quantity, ICAO has defined a 95% confidence level. It means that for any estimated

position at a specific location, the probability that the position error is within the former requirement should be at least 95%.

2.4.2   Availability

The availability of a navigation system is the ability of the system to provide the required function and performance at the initiation of the intended operation. Availability is an indication of the ability of the system to provide usable service within the specified coverage area. Signal availability is the percentage of time that navigational signals transmitted from external sources are available for use.

2.4.3   Continuity

Continuity is the capability of the total system (comprising all elements necessary to maintain aircraft position within the defined airspace) to perform its function without interruption during the intended operation. Continuity relates to the capability of the navigation system to provide a navigation output with the specified accuracy and integrity throughout the intended operation, assuming that it was available at the start of the operation. The occurrence of navigation system alerts, either due to rare fault-free performance or to failures, constitute continuity failures. For en-route, since the durations of these operations are variable, the continuity requirement is specified as a probability on a per-hour basis. For approach and landing operations, the continuity requirement is stated as a probability for a short exposure time.

2.4.4   Integrity

Integrity is a measure of the trust that can be placed in the correctness of the information supplied by the total system. Integrity includes the ability of a system to provide timely and valid warnings to the user (alerts) when the system must not be used for the intended operation.

Integrity risk: It is the probability of an undetectable (latent) failure of the specified accuracy. If the equipment is aware of the navigation mode/alert limit, a positioning failure is defined to occur whenever the difference between the true position and the indicated position exceeds the applicable alert limit (HAL or VAL). If the equipment is not aware of the navigation mode/alert limit, positioning failure is defined to occur whenever the difference between the true position and the indicated position exceeds the applicable protection level (HPL or VPL).

Time-to-Alert: Time-to Alert is the maximum allowable elapsed time from the onset of a positioning failure until the equipment annunciates the alert.

Alert limits: For each phase of flight, to ensure that the position error is acceptable, alert limits are defined that represent the largest position error which results in a safe operation.

- The Horizontal Alert Limit (HAL) is the radius of a circle in the horizontal plane (the local plane tangent to the WGS-84 ellipsoid), with its centre being at the true position, that describes the region that is required to contain the indicated horizontal position with the required probability for a particular navigation mode.
- The Vertical Alert Limit (VAL) is half the length of a segment on the vertical axis (perpendicular to the horizontal plane of WGS-84 ellipsoid), with its centre being at the true position, that describes the region that is required to contain the indicated vertical position with the required probability for a particular navigation mode.

While Horizontal Alert Limits requirements are defined for all the phases of flight, Vertical Alert Limits are only defined for phases of flight under NPA.

2.5   Annex 10 Signal in Space performance requirements

The Signal in Space (SIS) is the aggregate of guidance signals arriving at the antenna of an aircraft [RTCA, 2004].

The combination of GNSS elements and a fault-free GNSS user receiver shall meet the signal-in-space requirements defined in Table 2. It is important to notice that these requirements have been recently updated to take into account LPV 200 approaches mainly by the CAT I approach vertical alert limit.

The concept of a fault-free user receiver is applied only as a means of defining the performance of combinations of different GNSS elements. The fault-free receiver is assumed to be a receiver with nominal accuracy and time-to-alert performance. Such a receiver is assumed to have no failures that affect the integrity, availability and continuity performance [ICAO, 2006].

Typical operation	Accuracy Horizontal 95% (Notes 1 and 3)	Accuracy Vertical 95% (Notes 1 and 3)	Integrity (Note 2)	Time-to-alert (Note 3)	Continuity (Note 4)	Availability (Note 5)
En-route	H 3.7km	N/A	$1-10^{-7}$ $/h$	5 min	$1-10^{-4}/h$ to $1-10^{-8}/h$	0.99 To 0.99999
En-route, Terminal	H 0.74km	N/A	$1-10^{-7}$ $/h$	15 s	$1-10^{-4}/h$ to $1-10^{-8}/h$	0.99 To 0.99999
Initial approach, Intermediate approach, Non-Precision Approach (NPA), Departure	H 220m	N/A	$1-10^{-7}$ $/h$	10 s	$1-10^{-4}/h$ to $1-10^{-8}/h$	0.99 To 0.99999
Approach operations with vertical guidance (APV-I)	H 16m	V 20m	$1-2.10^{-7}$ $/app$	10 s	$1-8.10^{-6}$ $/15\ s$	0.99 To 0.99999
Approach operations with vertical guidance (APV-II)	H 16m	V 8m	$1-2.10^{-7}$ $/app$	6 s	$1-8.10^{-6}$ $/15\ s$	0.99 To 0.99999
Category I precision approach	H 16m	4≤V≤6 m	$1-2.10^{-7}$ $/app$	6 s	$1-8.10^{-6}$ $/15\ s$	0.99 To 0.99999

Table 2: SIS performance requirements [ICAO, 2006]

Notes associated to Table 2 and Table 3 [ICAO, 2006]:

1. The 95th percentile values for GNSS position errors are those required for the intended operation at the lowest height above threshold (HAT), if applicable.

2. The definition of the integrity requirement includes an alert limit against which the requirement can be assessed. For Category I precision approach, a vertical alert limit (VAL) greater than 10 m for a specific system design may only be used if a system-specific safety analysis has been completed. Further guidance on the alert limits is provided in [ICAO, 2006] Attachment D. These alert limits are:

Typical operation	Horizontal alert limit	Vertical alert limit
En-route (oceanic/continental low density)	7.4 km (4 NM)	N/A
En-route (continental)	3.7 km (2 NM)	N/A
En-route, Terminal	1.85 km (1NM)	N/A
NPA	556 m (0.3 NM)	N/A
APV – I	40 m (130 ft)	50 m (164 ft)
APV – II	40 m (130 ft)	20 m (66 ft)
CAT – I precision approach	40 m (130 ft)	35 to 10 m (115 to 33 ft)

Table 3: Alert Limits associated to typical operations [ICAO, 2006]

3. The accuracy and time-to-alert requirements include the nominal performance of a fault-free receiver.

4. Ranges of values are given for the continuity requirement for en-route, terminal, initial approach, NPA and departure operations, as this requirement is dependent upon several factors including the intended operation, traffic density, and complexity of airspace and availability of alternative navigation aids. The lower value given is the minimum requirements for areas with low traffic density and airspace complexity. The higher value given is appropriate for areas with high traffic density and airspace complexity. Continuity requirements for APV and Category I operations apply to the average risk (over time) of loss of service, normalized to a 15-second exposure time.

5. A range of values is given for the availability requirements as these requirements are dependent upon the operational need which is based upon several factors including the frequency of operations, weather environments, the size and duration of the outages, availability of alternate navigation aids, radar coverage, traffic density and reversionary operation procedures. The lower values given are the minimum availabilities for which a system is considered to be practical but are not adequate to replace non-GNSS navigation aids. For en-route navigation, the higher values given are adequate for GNSS to be the only navigation aid provided in an area. For approach and departure, the higher values given are based upon the availability requirements at airports with a large amount of traffic assuming that operations to or from multiple runways are affected but reversionary operational procedures ensure the safety of the operation.

6. A range of values is specified for Category I precision approach. The 4.0 m (13 feet) requirement is based upon ILS specifications and represents a conservative derivation from these specifications.

7. GNSS performance requirements for Category II and III precision approach operations are under review and will be included at a later date.

8. The terms APV-I and APV-II refer to two levels of GNSS approach and landing operations with vertical guidance (APV) and these terms are not necessarily intended to be used operationally.

2.6 Equipment Classes

The different GNSS receivers to be used for civil aviation can be classified according to their implemented functionalities as well as to their operational capabilities. This type of classification has been published in [RTCA, 2006] for GPS receivers. We recall here these definitions.

2.6.1 Operational Classes

Class 1: Equipment that supports oceanic and domestic en-route, terminal, approach (LNAV), and departure operation. When in oceanic and domestic en route, terminal, LNAV and departure operations, this class of equipment can apply the long-term and fast SBAS differential corrections when they are available.

Class 2: Equipment that supports oceanic and domestic en route, terminal, approach (LNAV, LNAV/VNAV), and departure operation. When in LNAV/VNAV, this class of equipment applies the long-term, fast, and ionospheric corrections. When in oceanic and domestic en route, terminal, approach (LNAV), and departure operations, this class of equipment can apply the long-term and fast SBAS differential corrections when they are available.

Class 3: Equipment that supports oceanic and domestic en route, terminal, approach (LNAV, LNAV/VNAV, LP, LPV), and departure operation. When in LPV, LP, or LNAV/VNAV, this class of equipment applies the long-term, fast, and ionospheric corrections. When in oceanic and domestic en route, terminal, approach (LNAV), and departure operations, this class of equipment can apply the long-term and fast SBAS differential corrections when they are available.

2.6.2 Functional Classes

Class Beta: Equipment consisting of a GPS/SBAS sensor that determines position (with integrity) and provides position and integrity data to an integrated navigation system (e.g., flight management system, multi-sensor navigation system). This equipment also provides integrity in the absence of the SBAS signal through the use of Fault Detection and Exclusion (FDE).

Class Gamma: Equipment consisting of both the GPS/SBAS position sensor (defined by Class Beta) and a navigation function, so that the equipment provides path deviations relative to a selected path. The equipment provides the navigation function required of a stand-alone navigation system. This equipment also provides integrity in the absence of the SBAS signal through the use of FDE. In addition, this class of equipment requires a database, display outputs and pilot controls.

Class Delta: Equipment consisting of both the GPS/SBAS position sensor (defined by Class Beta) and a navigation function, so that the equipment provides path deviations relative to a selected final approach path, similar to Class Gamma. However, not all of the functions provided by Class Gamma equipments are available from Class Delta. In particular, Class Delta does not provide an RNAV capability and is not required to provide a FAS database or direct pilot controls. It is understood that Class Delta equipment does provide means to be controlled. The Delta class of equipment is only

applicable to Class 4 that is intended to provide an ILS alternative. Aircraft that install Delta class equipment are expected to have a separate RNAV capability, as RNAV (GPS) approaches assume an RNAV capability up to the Final Approach Waypoint (FAWP) and for the missed approach (after the LTP/FTP). The integration of these systems is outside the scope of this document.

2.7 Combined Receivers

A sub-group of the WG-62 of EUROCAE, called the Ad-Hoc group is working on initial studies concerning GPS-GALILEO combined receivers used for civil aviation so as to prepare the future development of a GPS/GALILEO MOPS with RTCA SC-159. They are working on a Concept of Operations (ConOps) for combined GPS/GALILEO receivers [EUROCAE, 2009].

The main objective of this document is to provide a preliminary study on the operational benefits brought when using combined GPS/GALILEO receivers with respect to legacy GPS L1 receivers (including GPS L1 standalone, GPS L1 augmented by ABAS and GPS L1 augmented by SBAS). Due to the high number of GNSS signals that could be considered, this document focuses on a subset of selected GPS and GALILEO signals and presents a benefits trade-off study between various constellations/frequencies combinations against aircraft operations so as to make an initial selection of the promising solutions. This ConOps is forecasted to be used as an input to the development of a future GPS-GALILEO joint MOPS in coordination with RTCA [EUROCAE, 2009].

It can be noticed thanks to this document that the main foreseen benefits of combined receivers are [EUROCAE, 2009]:

- Greater resistance to unintentional interference
- Greater availability of satellites
- Robustness to the impact of control segment failures
- Improved accuracy and integrity

The concept of operations covers the following phases of flight: en-route, terminal area, Non Precision Approach, instrument approach with vertical guidance (LPV), and Precision Approach (Category I, II and III) [EUROCAE, 2009].

The concept of operations covers the following GNSS elements: GALILEO, GPS, Aircraft Based Augmentation Systems (ABAS), Satellite Based Augmentation Systems (SBAS) and Ground Based Augmentation Systems (GBAS) [EUROCAE, 2009].

According to [EUROCAE, 2009] a combination solution can be defined as:

- A set of available pseudorange measurement among GPS L1, GPS L5, GALILEO E1, GALILEO E5A and GALILEO E5B
- An associated integrity scheme which apply on the PVT solution to deliver the integrity information. It could be RAIM, SBAS, SoL, GBAS, INS-FDE, MRS or aRAIM.

This definition has been used during our study. Some of the combinations selected in the ConOps and concerning use of RAIM augmentations are presented below:

- GPS L1/L5 + RAIM
- GALILEO E1/E5A + RAIM
- GALILEO E1/E5B + RAIM
- GPS L1/L5 + GALILEO E1/E5A + RAIM

- GPS L1/L5 + GALILEO E1/E5B + RAIM

As we can see, RAIM may be used to implement several different combinations.

2.8 Synthesis

This section is a review of the requirements that a GNSS receiver has to fulfil so as to be authorized as a mean of navigation in a civil aircraft. Therefore, future GPS/GALILEO receivers will have to demonstrate their ability to fulfil these requirements so as to be considered reliable for specific phases of flight. Moreover, it presents the parameters that will allow evaluating the efficiency of new navigation solutions and proving their advantages upon legacy receivers.

Chapter 3: GNSS signals structure and signal processing for civil aviation

3.1   GNSS signals available for Civil Aviation

This section is a review of the different GPS and GALILEO signals that will be available for civil aviation applications with GALILEO setting up and GPS modernization.

GPS and GALILEO use a primary RNSS (Radio Navigation Satellite Service) allocation by International Telecommunication Union (ITU) in L1/E1, L2 and L5/E5 [ITU, 2012]. Use of L1/E1 and L5/E5 is however coordinated in accordance with primary allocation of L1/E1 and L5/E5 to ARNS systems [ITU, 2012]. ARNS band (Aeronautical Radio Navigation Services) is allocated by ITU to aeronautical radionavigation systems, but not eligible to satellite navigation because it is reserved for ground station to aircraft service.

The fact that ARNS allocation necessitates coordination with aeronautical services and that the use of GNSS for RNSS is dedicated to safety services, guarantee that L1/E1 and L5/E5 are protected against interferences.

As of today:
- very few systems use L1/E1 band
- DME-TACAN-JTIDS-MIDS and secondary radars use L5/E5 in a coordinated manner.

Use of GPS L2 cannot request protection against interferences from services in the 1215-1240 MHz, which makes it inappropriate for civil aviation applications.

ARNS bands are represented in Figure 3 along with the different GPS and GALILEO signals.
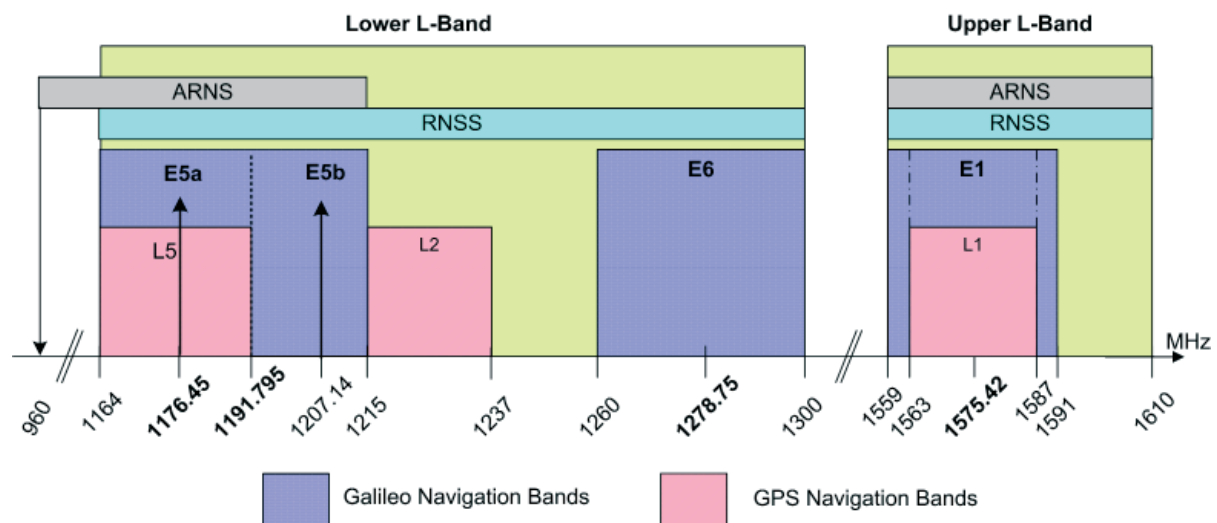
Figure 3: GALILEO frequency plan [GJU, 2010]

As we can see, the two different ARNS bands occupied by GNSS signals are the 960-1215 MHz band and the 1559-1610 MHz band. We will focus on the concerned signals which are:

- GPS L1 C/A
- GPS L1C
- GPS L5
- GALILEO E1
- GALILEO E5A
- GALILEO E5B

The characteristics of these signals are reminded in Table 4.

Constellation	Signal	Modulation	Code Length (ms)	Chip rate (Mcps)	Navigation Data (sps)	Secondary Code Length
GALILEO	E1B	CBOC(6,1,1/11)	4092	1,023	250	No
	E1C		4092	1,023	Pilot	Primary x 25 (100 ms)
	E5A-I	QPSK(10)	10230	10,23	50	Primary x 20 (20 ms)
	E5A-Q		10230	10,23	Pilot	Primary x 100 (100 ms)
	E5B-I	QPSK(10)	10230	10,23	250	Primary x 4 (4 ms)
	E5B-Q		10230	10,23	Pilot	Primary x 100 (100 ms)
GPS	L1 C/A	BPSK(1)	1023	1.023	50	No
	L1C-I	TMBOC(6,1,4/33)	10230	1.023	100	No
	L1C-Q		10230	1.023	Pilot	1800 bits
	L5-I	QPSK(10)	10230	10.23	1000	NH-10 (10bits)
	L5-Q		10230	10.23	Pilot	NH-20 (20 bits)

Table 4: GNSS signals for civil aviation

3.1.1 Structure of transmitted GNSS signals
This section presents the structure of the transmitted GNSS signals in the time and frequency domains.

3.1.1.1 Structure of transmitted BPSK signals
BPSK signals can be modelled as:

$$s_{BPSK}(t) = A.d(t).c(t).\cos(2\pi f_0 t)$$

(3-1)

With:

$$d(t) = \left[\sum_{k=-\infty}^{+\infty} d_k.\delta(t - k.T_D) \right] * rect\left(\frac{t - \frac{T_D}{2}}{T_D} \right)$$

(3-2)

$$c(t) = \sum_{i=-\infty}^{+\infty} c_i.m(t - i.T_C)$$

(3-3)

$$m(t) = rect\left(\frac{t - \frac{T_C}{2}}{T_C}\right)$$

<div align="right">(3-4)</div>

Where:

- f_0 is the signal carrier frequency
- $d(t)$ the data sequence waveform
- $c(t)$ the spreading code sequence waveform
- $d_k = \pm 1$ is the data bit, with T_D the data bit length
- $c_k = \pm 1$ is the Pseudo Random Noise (PRN) spreading code bit, with T_C the spreading code bit length
- $m(t)$ is the Non Return to Zero (NRZ) code materialization
- $rect\left(\frac{t}{T_C}\right) = \begin{cases} 0 \; if \; |t| > \frac{T_C}{2} \\ 1 \; if \; |t| \leq \frac{T_C}{2} \end{cases}$

Assuming the PRN code sequence is a random sequence of independent bits, it can be shown that the normalized power spectrum density expression of $c(t)$ can be expressed as:

$$G_c(f) = \frac{1}{T_C}.|M(f)|^2 = T_C.\left(\frac{\sin\left(\pi.\frac{f}{f_C}\right)}{\pi.\frac{f}{f_C}}\right)^2$$

<div align="right">(3-5)</div>

With f_C is the rate of the spreading code and $f_C = \frac{1}{T_C}$.

The same expression can be computed for $d(t)$ assuming that the data sequence is a random sequence of independent bits:

$$G_d(f) = T_D.\left(\frac{\sin\left(\pi.\frac{f}{f_D}\right)}{\pi.\frac{f}{f_D}}\right)^2$$

<div align="right">(3-6)</div>

With f_D is the rate of the spreading code and $f_D = \frac{1}{T_D}$.

If we consider the baseband BPSK signal:

$$s_{bpsk}(t) = A.d(t).c(t)$$

<div align="right">(3-7)</div>

Then we have:

$$G_{bpsk}(f) = G_d(f) * G_c(f)$$

<div align="right">(3-8)</div>

3.1.1.2 Structure of transmitted BOC signals

The Binary Offset Carrier modulation is very interesting for improving the spectral occupation of a particular frequency band since it is a simple way to shift the energy away from the frequency band centre. It also offers a high degree of spectral separation with signals centred in the band considered.

Offset Carrier modulations consist in the product of the NRZ materialization of a PRN code with a subcarrier. The properties of the resulting signal depend on the type of subcarrier chosen. In the case of BOC modulations, the subcarrier is a NRZ signal equal to the sign of a sine or cosine waveform. This solution has been chosen for GALILEO signals since it has a constant complex envelope which is much easier to emit from navigation satellites and avoids distortions due to non linear amplification.

BOC signals are in general referred to as BOC(p,q) where p defines the subcarrier rate f_s and q defines the spreading code rate f_c as following:

$$f_s = p \times 1.023 \ MHz$$

$$f_c = q \times 1.023 \ MHz$$

(3-9)

Two different notations are used in the literature to describe BOC modulations. We will use here the model exposed in [BETZ, 2002] for a sine subcarrier:

$$x(t) = c(t). \, sign(\sin(2\pi f_s t))$$

(3-10)

Where:

- $c(t)$ is the code sequence waveform of period $T_C = \frac{1}{f_c}$:

$$c(t) = \sum_{k=-\infty}^{+\infty} c_k. \, m(t - k. T_c)$$

- $m(t)$ is the NRZ code materialization:

$$m(t) = rect\left(\frac{t - \frac{T_C}{2}}{T_C}\right)$$

The ratio $n = 2\frac{f_s}{f_c} = 2\frac{p}{q}$ represents the number of half periods of the sub carrier during one code chip. It can be either odd or even. Depending on it, the power spectrum density will have a different expression. It can be shown, that, assuming the PRN code sequence is random and independent, and assuming that code-carrier coherency is random, the normalized power spectrum densities of base-band sine-phase BOC signals is expressed as:

$$G_{BOC}(f) = \frac{1}{T_C}\left(\frac{\sin\left(\frac{\pi f T_c}{n}\right)\sin(\pi f T_c)}{\pi f \cos\left(\frac{\pi f T_c}{n}\right)}\right)^2 \quad \text{for n even}$$

(3-11)

$$G_{BOC}(f) = \frac{1}{T_C}\left(\frac{\sin\left(\frac{\pi f T_c}{n}\right)\cos(\pi f T_c)}{\pi f \cos\left(\frac{\pi f T_c}{n}\right)}\right)^2 \quad \text{for n odd}$$

(3-12)

To illustrate these expressions we have represented below the power spectrum density and autocorrelation of sine BOC(1,1) signal.
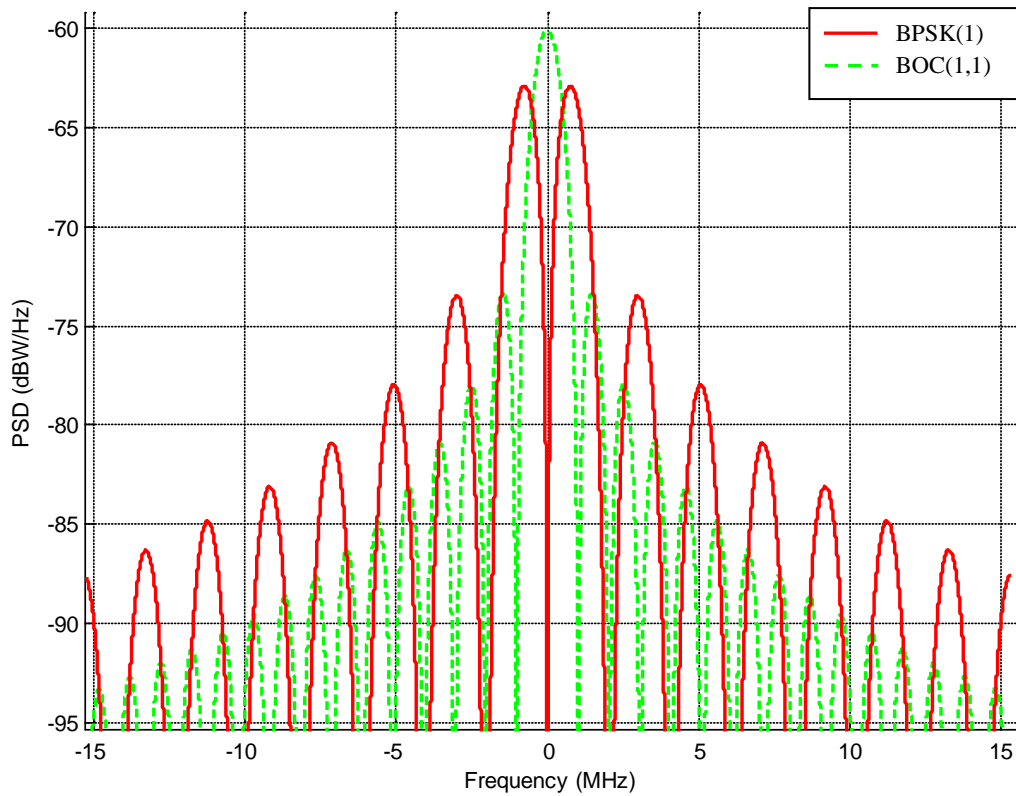
Figure 4 : BPSK and BOC(1,1) Power Spectrum Density

We can see that this spectrum is split on each side of the central frequency. Moreover we can remark that the main lobes of the BPSK are placed just in between the lobes of the BOC(1,1) allowing a good spectral separation.
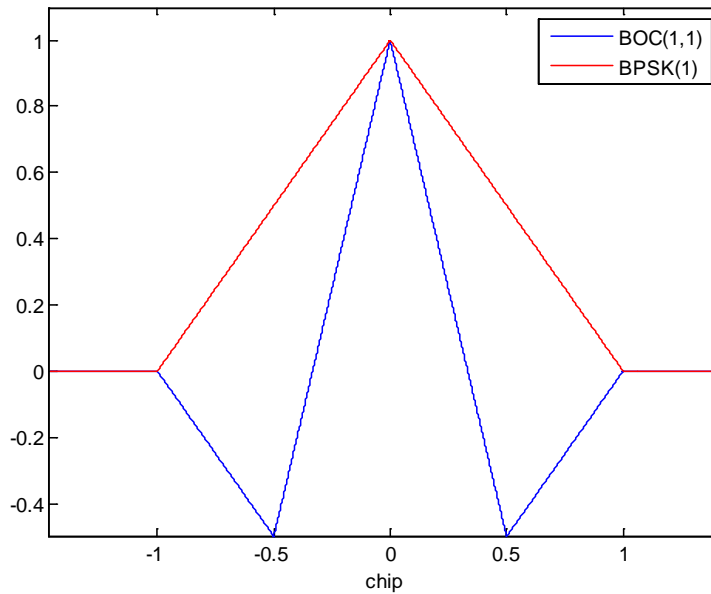
Figure 5 : BOC(1,1) Autocorrelation Function

We can see in figure 5 that the autocorrelation of the baseband BOC(1,1) signal exhibits secondary peaks at -0.5 and +0.5.

3.1.1.3 *Structure of transmitted MBOC signals*

So as to maintain a high interoperability between future GPS and GALILEO civil signals in L1 band that is to say GALILEO E1 OS and GPS L1C, United States and Europe relevant authorities agreed to share a common normalized power spectral density referred to as Multiplexed Binary Offset Carrier (MBOC) PSD [GODET, 2006] and [HILBRECHT, 2006]. It is important to understand that it is only defined spectrally, and therefore it can be using different implementations. This section only presents the spectral definition of MBOC and the implementations chosen for GALILEO and GPS will be presented hereafter in relevant sections of this document.

The MBOC PSD is the PSD of all the components of the civil signals, which means data and pilot components, and it has the following expression [JULIEN, 2007]:

$$G(f) = \frac{10}{11} G_{BOC(1,1)}(f) + \frac{1}{11} G_{BOC(6,1)}(f)$$

(3-13)

A particular implementation of the MBOC which is called TMBOC - Time Multiplexed Binary Offset Carrier - is represented in Figure 6.

The main reason for this design was to put a fraction of the power away from the carrier frequency so as to improve the tracking performance in presence of thermal noise and multipath compared to the previous pure BOC(1,1) baseline signal [JULIEN, 2007].

As we said previously different implementations allow obtaining this PSD and therefore several parameters have to be determined and in particular:

- The respective power of the data and pilot channels

- The respective weights of the BOC(1,1) and BOC(6,1) components within the data and pilot channels.
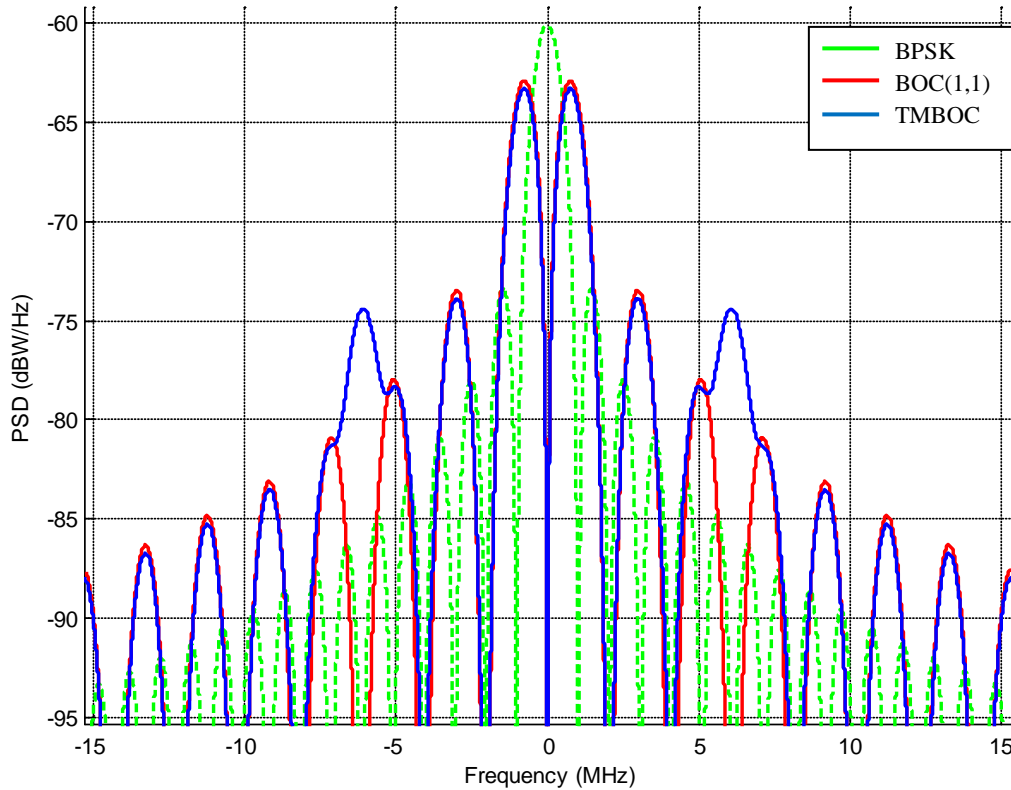
Figure 6: MBOC Power Spectral Density

3.1.1.4   *Structure of transmitted ALTBOC signals*

This section is intended to give a brief overview of the Alternate Binary Offset Carrier (ALTBOC) modulation and therefore only the basic principle will be presented.

As we have seen previously, the BOC modulation is a square subcarrier modulation. The principle of ALTBOC modulation is to multiply the signal to be transmitted with a complex subcarrier. In that way, the baseband signal spectra is not split up but is shifted in the upper or lower frequencies. Therefore, it allows transmitting one different component in each side-lobe of the Offset Carrier where in the case of BOC, each side-lobe contains the same information. To transmit two different signals c_a and c_b we can use the following signal:

$$x(t) = c_a(t).\,[sign(\cos(2\pi f_s t)) + j.\,sign(\sin(2\pi f_s t))]$$
$$+ c_b(t).\,[sign(\cos(2\pi f_s t)) - j.\,sign(\sin(2\pi f_s t))]$$

(3-14)

The main disadvantage of this solution is that it has a constant envelope only if the baseband signals are BPSK. Therefore, it does not permit to transmit signals composed of a data and a pilot channel (QPSK). We will see in a further section that a modified version of the ALTBOC modulation has been proposed for GALILEO E5 signals broadcasting which is called Constant Envelope ALTBOC. We remind here the definition of this modified ALTBOC modulation [REBEYROL, 2007].

$$x_{ALTBOC}(t) = \begin{cases} \left(c_L(t) + j.c_L{}'(t)\right).\left[sc_{as}(t) - j.sc_{as}\left(t - \dfrac{T_S}{4}\right)\right] + \\[2mm] \left(c_U(t) + j.c_U{}'(t)\right).\left[sc_{as}(t) + j.sc_{as}\left(t - \dfrac{T_S}{4}\right)\right] + \\[2mm] \left(\overline{c_L(t)} + \overline{j.c_L{}'(t)}\right).\left[sc_{ap}(t) - j.sc_{ap}\left(t - \dfrac{T_S}{4}\right)\right] + \\[2mm] \left(\overline{c_U(t)} + \overline{j.c_U{}'(t)}\right).\left[sc_{ap}(t) + j.sc_{ap}\left(t - \dfrac{T_S}{4}\right)\right] + \end{cases}$$

<div align="right">(3-15)</div>

With:

- T_S is the subcarrier period with $T_S = \dfrac{1}{f_S}$
- c_U the data upper code
- $c_U{}'$ the pilot upper code
- c_L the data lower code
- $c_L{}'$ the pilot lower code
- $\overline{c_L} = c_U c_U{}' c_L{}', \overline{c_L{}'} = c_U c_U{}' c_L, \overline{c_U} = c_L c_U{}' c_L{}'$ and $\overline{c_U{}'} = c_L c_L{}' c_U$
- $sc_{as}(t) = \dfrac{\sqrt{2}}{4} sign\left[cos\left(2\pi f_S t - \dfrac{\pi}{4}\right)\right] + \dfrac{1}{2} sign[cos(2\pi f_S t)] + \dfrac{\sqrt{2}}{4} sign\left[cos\left(2\pi f_S t + \dfrac{\pi}{4}\right)\right]$
- $sc_{ap}(t) = -\dfrac{\sqrt{2}}{4} sign\left[cos\left(2\pi f_S t - \dfrac{\pi}{4}\right)\right] + \dfrac{1}{2} sign[cos(2\pi f_S t)] - \dfrac{\sqrt{2}}{4} sign\left[cos\left(2\pi f_S t + \dfrac{\pi}{4}\right)\right]$

The normalized power spectrum density of the baseband constant envelope ALTBOC signal for n odd is:

$$G_{ALTBOC}(f) = \frac{4}{\pi^2 f^2 T_C} \frac{cos^2(\pi f T_C)}{cos^2\left(\pi f \dfrac{T_C}{n}\right)}$$
$$\times \left[cos^2\left(\pi f \frac{T_S}{2}\right) - cos\left(\pi f \frac{T_S}{2}\right) - 2 cos\left(\pi f \frac{T_S}{2}\right) cos\left(\pi f \frac{T_S}{4}\right) + 2\right]$$

<div align="right">(3-16)</div>

3.1.2 Civil Aviation signals in L1 band

3.1.2.1 GPS L1 C/A

GPS L1 C/A is the legacy GPS signal and it is a BPSK signal. We will use this signal as a basis to present the classical principles of GNSS. The following model represents the L1 signal radiated by the antenna of a GPS satellite i [SPILKER, 1996]:

$$s^i{}_{L1}(t) = \sqrt{2 * P}.C_i(t)D_i(t) cos(2\pi f_{L1} t + \theta) + \sqrt{P}.P_i(t)D_i(t) sin(2\pi f_{L1} t + \theta)$$

<div align="right">(3-17)</div>

Where:

- P is the mean radiated power
- D_i is the P/NRZ materialization of the navigation message
- P_i is the P/NRZ materialization of the military P(Y) code of the satellite
- C_i is the P/NRZ materialization of the public C/A code of the satellite

- f_{L1} = 1575.42 MHz is the L1 carrier frequency
- θ is the phase on L1
- t is the time in seconds, expressed in the satellite generated time scale $t = t_{GPS} - \Delta t_S^i$
- t_{GPS} is the time expressed in GPS time scale
- Δt_S^i is the satellite clock offset

Civilian users like civil aviation users can only track the C/A component of the L1 signal since they have no knowledge of the military code. This signal since it is relatively easy to track is a reference to GPS mass-market. Nonetheless, this signal structure can be improved to provide better acquisition, tracking and demodulation performance. New signals structures were proposed in the frame of GPS modernization GPS III, GALILEO, QZSS for example which include high rate PRN codes, data/pilot components, BOC modulations and new data delivery techniques.

We remind here the ideal model of the autocorrelation function of the materialization of the C/A spreading code:

$$K_C(\tau) = \begin{cases} 1 - \dfrac{|\tau|}{T_C} & if\ |\tau| < T_C \\ 0\ elsewhere \end{cases}$$

(3-18)

It can be seen in Figure 5.

Assuming that the PRN code sequence is a random sequence of independent bits, the power spectral density of the signal is a sinus cardinal. This envelope corresponds to the PSD of the code wave form which is rectangular. It has already been discussed in section 3.1.1.

3.1.2.2   GPS L1C

As we said previously US accepted to share the GPS L1C PSD for the MBOC PSD definition. However, different implementations are possible to obtain this PSD. The main candidate for GPS L1C is the Time Multiplexed BOC modulation. It uses a time-multiplexing of the two MBOC components which are a BOC (1,1) and a BOC(6,1).

The general expression of the TMBOC is given in [JULIEN, 2007]:

$$s_{TMBOC(6,1,p)}(t) = \begin{cases} x(t) & if\ t \in S_1 \\ y(t) & if\ t \in S_2 \end{cases}$$

(3-19)

With:

- $c(t)$ the code sequence waveform
- $x(t)$ and $y(t)$ respectively are the BOC (1,1) and BOC(6,1) signals resulting of the product of the code sequence waveform $c(t)$ and the BOC(1,1) and BOC(6,1) subcarrier waveforms:
$$x(t) = c(t).BOC_1(t)$$
$$y(t) = c(t).BOC_6(t)$$
- $BOC_1(t) = sign(\sin(2\pi.f_0 t))$ the BOC(1,1) subcarrier waveform
- $BOC_6(t) = sign(\sin(6 \times 2\pi.f_0 t))$ the BOC(6,1) subcarrier waveform
- $f_0 = 1.023\ MHz$

46

- S_1 union of the temporal slots where $x(t)$ is used
- S_2 union of the temporal slots where $y(t)$ is used

The predicted power distribution according to [BETZ, 2006] is 25% of power on the data channel and 75% of power on the pilot channel. Moreover, it has been decided that for the data channel, the signal modulation is a pure BOC(1,1). It implies that the pilot signal will be a TMBOC(6,1,4/33) meaning that there will be 4/33 of the power for the BOC(6,1) component and 29/33 of power for the BOC(1,1) component. This leads to the following baseband PSD for the two channels [HEIN, 2006]:

$$G_{data}(f) = G_x(f)$$

$$G_{pilot}(f) = \frac{29}{33} G_x(f) + \frac{4}{33} G_y(f)$$

(3-20)

Therefore, the temporal implementation proposed is to select 4 given chips every 33 chips of the spreading code and modulate these with a BOC(6,1) subcarrier, while the others will be modulated by a BOC(1,1). This scheme is thus far more complex than for legacy GPS L1 C/A signal.

We have already presented the MBOC PSD in section 3.1.1.3. We remind here the expression of the autocorrelation functions of the TMBOC designed for GPS L1C.

For the data channel, the model of the autocorrelation function is the ideal model of the autocorrelation function of the BOC(1,1) signal $x(t)$:

$$R_x(\tau) = tri\left(\frac{\tau}{1}\right) - \left(tri\left(\frac{\tau - {}^1/_2}{{}^1/_2}\right) + tri\left(\frac{\tau + {}^1/_2}{{}^1/_2}\right) \right)$$

(3-21)

With $tri\left(\frac{\tau - a}{T}\right)$ the triangular function of width $2T$ centered in a and of magnitude 1.

For the pilot channel we have:

$$R_{TMBOC(6,1,4/33)}(\tau) = \frac{1}{33}\left(29. R_x(\tau) + 4. R_y(\tau)\right)$$

(3-22)

Where R_x and R_y are the autocorrelation functions of $x(t)$ and $y(t)$.

3.1.2.3 GALILEO E1 OS
As already said GALILEO E1 OS will use the MBOC PSD as well as GPS L1C, but with a different implementation which is called Composite BOC (CBOC).

For the moment, it is likely that the power will be equally split between the pilot and data channels where as we have seen previously for GPS it has been decided to put more power for the pilot channel. In the case of GALILEO, it is also likely that both data and pilot channels will carry an equal amount of BOC(6,1) [JULIEN, 2007].

The GALILEO CBOC signal can be written as [AVILA-RODRIGUEZ, 2006]:

$$\begin{cases} s_{data}(t) = c_D(t)d(t)\left(\sqrt{(1-p)}.BOC_1(t) + \sqrt{p}.BOC_6(t)\right) \\ s_{pilot}(t) = c_P(t)\left(\sqrt{(1-p)}.BOC_1(t) - \sqrt{p}.BOC_6(t)\right) \end{cases}$$

(3-23)

Where:

- c_D and c_P are respectively the data and pilot channels spreading code sequences
- d is the navigation message
- BOC_1 and BOC_6 are respectively the BOC(1,1) and BOC(6,1) subcarrier waveforms
- $P = \sqrt{1/11}$ the weight of the BOC(1,1) waveform
- $Q = \sqrt{10/11}$ the weight of the BOC (1,1) waveform
- p the amount of BOC(6,1) power within the whole civil channel power with $p = \frac{Q^2}{P^2+Q^2}$

The sign inversion of the BOC(6,1) component between the data and pilot channels allows removing the cross correlation terms between BOC(1,1) and BOC(6,1) which appear on each channel. Therefore the GALILEO E1 OS tracking can be done using a CBOC(6,1,p,"+") on the data channel and a CBOC(6,1,p,"-") on the pilot channel which are defined as:

$$\text{CBOC}(6,1,p," + ") = c(t)\left(\sqrt{(1-p)}.BOC_1(t) + \sqrt{p}.BOC_6(t)\right)$$

$$\text{CBOC}(6,1,p," - ") = c(t)\left(\sqrt{(1-p)}.BOC_1(t) - \sqrt{p}.BOC_6(t)\right)$$

As we can see, CBOC signal subcarriers are a weighted sum of the BOC(1,1) and BOC(6,1) subcarriers, so the CBOC local replica will be represented by four amplitude value levels instead of two for classic BOC signals. It implies that the receiver will have to be able to generate a muti-level local replica.

The PSD of s_{OS} as explained previously is the MBOC PSD. The autocorrelation function of the GALILEO OS data and pilot waveforms can be expressed as [JULIEN, 2006]:

$$R_{\text{CBOC}(6,1,p,"+")}(\tau) = \frac{1}{11}\left(10.R_x(\tau) + R_y(\tau) + \frac{20}{11}R_{x/y}(\tau)\right)$$

$$R_{\text{CBOC}(6,1,p,"-")}(\tau) = \frac{1}{11}\left(10.R_x(\tau) + R_y(\tau) - \frac{20}{11}R_{x/y}(\tau)\right)$$

Where:

- R_x and R_y are the autocorrelation functions of the BOC(1,1) and BOC(6,1) signals defined as the product of the code sequence and the subcarrier waveforms:
$$x(t) = c(t).BOC_1(t)$$
$$y(t) = c(t).BOC_6(t)$$

- $R_{x/y}$ is the cross correlation function between the BOC(1,1) and BOC(6,1) waveforms

3.1.2.4 *Civil Aviation considerations*

As we have seen in this section, the future new GNSS signals GPS L1C and GALILEO E1 OS are introducing a higher degree of complexity in the receivers. It has been anticipated that the increased performance may not balance the increased complexity. That's why it is anticipated that Civil Aviation users may, as a first step, use an intermediate solution which consists in tracking only BOC(1,1) part of the signals. Studies have been led which show that the loss of accuracy is in this case not so constraining. Therefore, in our study we will consider the two different possibilities.

3.1.3 Civil Aviation signals in L5

In this band, both GPS and GALILEO will broadcast wideband signals which offer very high tracking accuracy. We briefly present here these signals.

3.1.3.1 *GPS L5*

The introduction of GPS L5 aims at providing a completely redundant signal to the L1 signal so as to enhance signal diversity and as a consequence robustness in case of interference. Moreover, it will enable civilian GPS users to perform dual frequency measurements and so to compute the iono-free hybrid pseudorange measurements which are supposed to be free of ionospheric error. The transmitted GPS L5 signal is a QPSK signal including a pilot and a data channel. Therefore, it is equivalent to two BPSK signals. We have already presented the characteristics of this type of signals in the previous sections.

3.1.3.2 *GALLILEO E5A and E5B*

GALILEO E5A and E5B have the same characteristics and are components of the E5 whole signal. These two components can be modelled as two separate BPSK channels. These two signals are broadcasted using the constant envelope ALTBOC modulation which has been described in section 3.1.1.4. Therefore, different configurations are available to users. They can either track the whole signal E5 (E5A+E5B) coherently, or separately E5A or E5B.

Tracking the whole E5 signal allows for extremely high code tracking accuracy but to do so it is necessary to use an extra-wide band front end filter which is very challenging and expensive. Moreover, by processing the entire E5 signal, there is a risk to propagate degradations experienced only on one frequency band to the other. Typically we could imagine interferences or jamming on E5B that would impact the signal processing of E5A.

3.2 Receiver Signal Processing

3.2.1 Receiver Architecture

This section is intended to present an overview of a classical GNSS receiver architecture. The structure of the receiver is represented in the following block diagram:
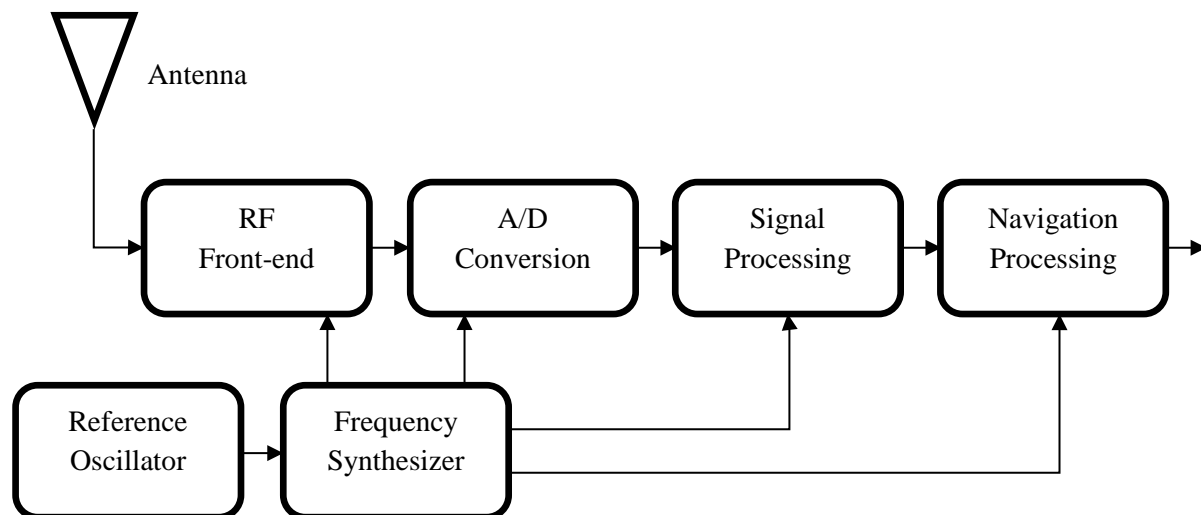
Figure 7: GNSS receiver structure

Reference oscillator, Frequency Synthesizer: The measurements made by GNSS receivers are estimates of Time Of Arrival (TOA) and received carrier phase and frequency. It is therefore necessary to have a local reference oscillator to drive the local replicas of GNSS signals. That's why the reference oscillator accuracy and stability is crucial to GNSS receiver measurements accuracy. The output of the reference oscillator is used by the frequency synthesizer to derive local oscillators needed within the receiver.

Antenna and RF font end: The antenna is designed to receive GNSS signals and to have as much as possible the following characteristics: frequency selectivity, constant gain pattern above a given elevation, good multipath and interference rejection capabilities, low gain at negative elevation angles, stable phase and group centre, stable group delay. 'Then, the RF front-end is in charge of processing the analog signals at the output of the antenna, which consists mainly in the following steps: amplifications, filtering and down-conversion to an appropriate frequency (Intermediate Frequency (IF) or baseband).

Analog to Digital Converter (ADC) / Automatic Gain Control (AGC): For signal processing, it is necessary to digitize the analog signals. The A/D converter samples and quantizes the incoming signal. The AGC aims at adapting the power of the received signal to the quantization range of the ADC so as to avoid undesired distortions.

Signal Processing and Navigation Processing is detailed in further sections of this document.

3.2.2   Receiver Signal Processing

In this section we present the signal processing of GPS L1 C/A since it is the legacy signal but the general principles are applicable to the other GPS/GALILEO signals. The signal processing includes three different functions which are acquisition, tracking and demodulation which are achieved through the use of two different tracking loops:

- A Delay Lock Loop (DLL) that continuously controls a local replica of the signal code synchronized with the incoming code.
- A Carrier Tracking Loop (PLL or FLL) that continuously controls a local replica of the signal carrier synchronized with the incoming carrier.

In this study we assume that acquisition and demodulation are functioning nominally. Thus, we only describe the tracking function.

We will consider the following model for the GPS L1 C/A received signal, sampled and quantized:

$$r(k) = Ad(kT_e - \tau).c(kT_e - \tau)\cos(2\pi f_0 kT_e - \varphi) + b(k)$$

(3-24)

Where:

- A is the amplitude of the received signal
- $d(t)$ the navigation message materialization
- $c(t)$ the PRN code sequence materialization
- T_e the sampling period of the signal at the output of the ADC.
- τ the group propagation delay
- φ the phase of the received carrier
- $b(t)$ the Gaussian noise affecting the measurements

3.2.2.1 *Correlator outputs model*

The correlator outputs can be modelled as [BASTIDE, 2002]:

$$
\begin{cases}
I_E(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c(\varepsilon_\tau + \dfrac{C_s}{2}).sinc(\pi\varepsilon_f T_I)\cos(\varepsilon_\varphi) + n_{I_E}(k) \\[2mm]
I_P(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c(\varepsilon_\tau).sinc(\pi\varepsilon_f T_I).\cos(\varepsilon_\varphi) + n_{I_P}(k) \\[2mm]
I_L(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c(\varepsilon_\tau - \dfrac{C_s}{2}).sinc(\pi\varepsilon_f T_I).\cos(\varepsilon_\varphi) + n_{I_L}(k) \\[2mm]
Q_E(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c(\varepsilon_\tau + \dfrac{C_s}{2}).sinc(\pi\varepsilon_f T_I).\sin(\varepsilon_\varphi) + n_{Q_E}(k) \\[2mm]
Q_P(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c(\varepsilon_\tau).sinc(\pi\varepsilon_f T_I).\sin(\varepsilon_\varphi) + n_{Q_P}(k) \\[2mm]
Q_L(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c(\varepsilon_\tau - \dfrac{C_s}{2}).sinc(\pi\varepsilon_f T_I).\sin(\varepsilon_\varphi) + n_{Q_L}(k)
\end{cases}
$$

(3-25)

- $\widetilde{K}_c$ is the correlation function between the locally generated spreading code with the filtered incoming spreading code
- $\varepsilon_\varphi = \varphi - \hat{\varphi}$ the carrier phase error between the incoming carrier phase φ and the locally generated carrier phase $\hat{\varphi}$
- $\varepsilon_\tau = \tau - \hat{\tau}$ is the code delay error between the incoming code delay τ and the locally generated code delay $\hat{\tau}$
- ε_f is the frequency error between the incoming carrier and the locally generated carrier,
- T_I is the correlation time
- C_s is the chip spacing
- $n_{I_E}(k)$, $n_{I_P}(k)$, $n_{I_L}(k)$ are respectively the Early, Prompt and Late In-Phase post-correlation thermal noise components.

- $n_{Q_E}(k)$, $n_{Q_P}(k)$, $n_{Q_L}(k)$ are respectively the Early, Prompt and Late Quadra-Phase post-correlation thermal noise components. They are uncorrelated with In-Phase thermal noise components.
- The power of the prompt thermal noise components can be computed as [JULIEN, 2010]:

$$Var\big[n_{I_P}\big] = Var\big[n_{Q_P}\big] = \frac{N_0}{4T_I}\beta$$

- $N_0/2$ is the equivalent double-sided thermal noise PSD at the correlator input.
- $\beta = \int_{-\infty}^{+\infty}|H(f)|^2\,|F_L(f)|^2df$ is the loss of noise power due to the front-end filter.
- H is the equivalent front-end filter transfer function
- $F_L(f)$ is the Fourier transform of the locally generated code sequence

3.2.2.2 *Carrier tracking loop*

There are two types of carrier tracking loops which are the Phase Lock Loop (PLL) and the Frequency Lock Loop. The structure of a classic PLL is represented in

Figure **8**. The principle is to estimate the phase shift between the received carrier and the locally generated carrier so as to maintain it as close to zero as possible. This allows wiping off the carrier of the received signal by multiplying the incoming signal with the locally generated carrier and then filtering the result so as to remove the undesired high frequency component.
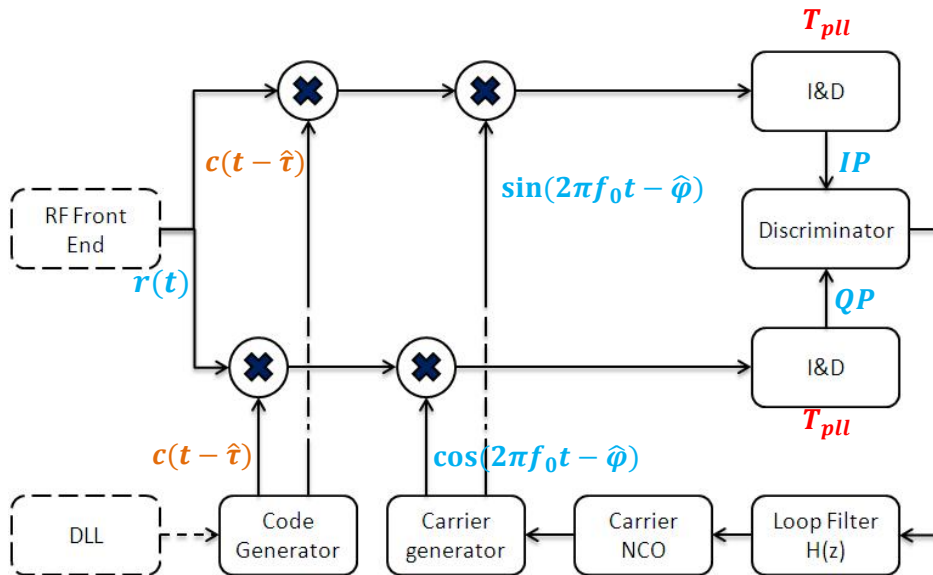
Figure 8: Phase Lock Loop structure (with T_{pll} the coherent integration time)

To estimate the carrier phase error between the incoming carrier phase and the locally generated carrier phase different discriminators can be used. Here are some examples:

$$D_{Costas} = Q_P \times I_P$$

(3-26)

$$D_{Atan} = \text{atan}\left(\frac{Q_P}{I_P}\right)$$

(3-27)

It is important to notice that Costas discriminator depends on the power of the incoming signal, it is thus necessary to normalize it.

We can see in figure 6 that discriminator outputs are then filtered by the PLL loop filter. The resulting output is integrated so as to obtain the locally generated carrier phase at the output of the NCO.

3.2.2.3 *Delay Lock Loop*
We present here the structure of a classical DLL in Figure 9.
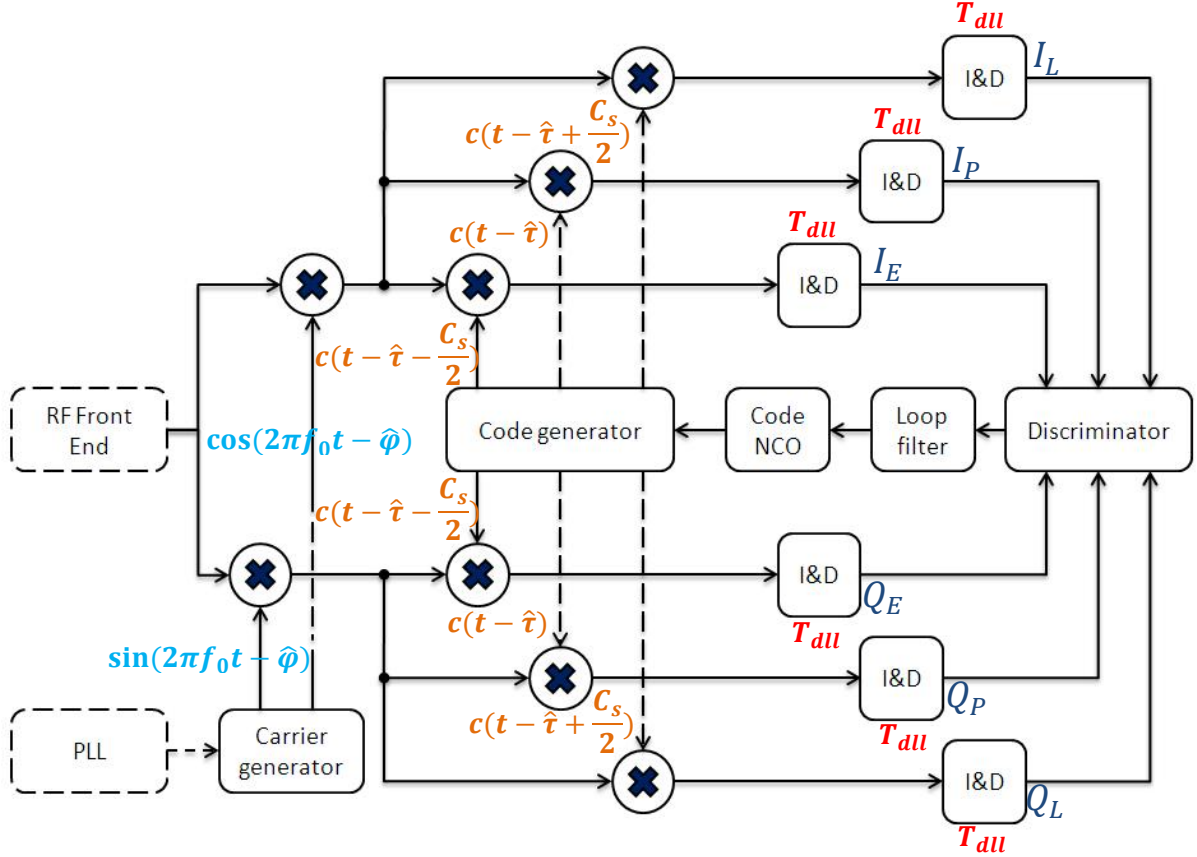
Figure 9: Structure of a classic DLL (with T_{dll} the correlator integration time)

As we can see, the early, prompt and late correlator outputs are fed into a code delay discriminator so as to estimate the code delay tracking error between the incoming code sequence delay and the locally generated code sequence delay. The most commonly used discriminators are the Early-Minus Late Power (EMLP) and the Dot-Product (DP):

$$D_{EMLP} = \left(I_E{}^2 + Q_E{}^2\right) - \left(I_L{}^2 + Q_L{}^2\right)$$

(3-28)

$$D_{DP} = (I_E - I_L)I_P + (Q_E - Q_L)Q_P$$

(3-29)

It is important to understand that these discriminators depends on the signal power and must therefore be normalized. The discriminator outputs are then filtered by the DLL loop filter. The DLL loop filter output is then integrated and fed into the NCO so as to determine the phase of the locally generated code replicas.

3.2.2.4 *Tracking error*

The error variance of the tracking loops due to noise has been studied and expressed depending on the type of discriminator used. In our study we used two different models.

[HOLMES, 1982] formulas are based on the assumption of infinite RF bandwidths. They provide a good approximation for the standard deviation of the code tracking error, and a precise estimation for the standard deviation of the carrier tracking error since the latter is not dependent on the receiver RF/IF bandwidth. These can be found in [SPILKER, 1996]:

- Code tracking error:

$$\sigma_{\tau,DP} = \sqrt{\frac{B_{DLL}.C_s}{2.C/N_0}\left(1 + \frac{1}{C/N_0.T_{DLL}}\right)}$$

(3-30)

$$\sigma_{\tau,EMLP} = \sqrt{\frac{B_{DLL}.C_s}{2.C/N_0}\left(1 + \frac{2}{(2-C_s).C/N_0.T_{FLL}}\right)}$$

(3-31)

- Carrier tracking error (for COSTAS discriminator):

$$\sigma_\theta = \sqrt{\frac{B_{PLL}}{2.C/N_0}\left(1 + \frac{1}{2.C/N_0.T_{DLL}}\right)}$$

(3-32)

Where:

- B_{DLL} and B_{PLL} are respectively the bandwidth of the equivalent linear model of the code and carrier tracking loops
- T_{DLL} and T_{FLL} are respectively the coherent integration time of the code and carrier loops

[BETZ, 2000] formulas take into account the RF bandwidth, but considering an ideal rectangular filter. They provide a better estimate of the code tracking error. They can be found in [JULIEN, 2005]:

$$\sigma_{\tau,DP} = \sqrt{\frac{B_{DLL}\int_{-B/2}^{B/2} G(f)sin^2(\pi f C_s)df}{\frac{C}{N_0}\left(2\pi\int_{-B/2}^{B/2} fG(f)sin(\pi f C_s)df\right)^2}\left(1 + \frac{1}{\frac{C}{N_0}T_{DLL}\left(\int_{-B/2}^{B/2} G(f)df\right)}\right)} \; (m)$$

(3-33)

$$\sigma_{\tau,EMLP} = \sqrt{\frac{B_{DLL}\int_{-B/2}^{B/2} G(f)sin^2(\pi f C_s)df}{\frac{C}{N_0}\left(2\pi\int_{-B/2}^{B/2} fG(f)sin(\pi f C_s)df\right)^2}\left(1 + \frac{\int_{-B/2}^{B/2} G(f)cos^2(\pi f C_s)df}{\frac{C}{N_0}T_{DLL}\left(\int_{-B/2}^{B/2} G(f)cos(\pi f C_s)df\right)^2}\right)} \; (m)$$

(3-34)

Where:

- B is the RF front-end filter bandwidth (Hz)
- C the received signal power over infinite signal bandwidths (W)
- N_0 the power spectral density of the noise (W/Hz)
- $G(f)$ is the power spectral density of the processed signal

3.3   Synthesis

This chapter presents the different GNSS signals that were considered in this study. It consists in the GNSS signals either currently available to civilian users (GPS L1 C/A) or that will be available in the mid-term thanks to the modernization of GPS or the introduction of the GALILEO constellation. The structure of these signals has been described as well as their main characteristics, and in particular their power spectral density and autocorrelation functions. This chapter also contains a brief description of the signal processing of the legacy GPS L1 C/A signal as well as the characterization of the associated tracking errors which are used as reference all along this work.

Chapter 4: GNSS measurement model

4.1   Introduction

As we explained in the previous chapter, the method used in GNSS receivers to elaborate the pseudorange measurements is to generate a local replica of the satellite code and to shift it to obtain the highest correlation possible between the received code and the local one. The phase shift of the code replica, if multiplied by the speed of light, corresponds to the satellite range biased by the user receiver clock error with respect to GNSS reference time. This measurement is called the pseudorange since it is biased with respect to the true distance between the user receiver antenna and the satellite antenna. This pseudorange is usually measured using a tracking loop such as the DLL.

However, the receiver is not able to measure the pseudorange. The measured pseudorange is in fact affected by several additional errors:

- Delays induced by the propagation of the signal through atmosphere
- Multipath
- Receiver errors and in particular thermal noise

So as to determine the distribution of errors affecting the GNSS position solutions it is necessary to evaluate and model the residual errors affecting the pseudorange measurements after application of corrections on the raw pseudorange measurements.

The main steps of the processes which allow building the final measurements used to compute a position in GNSS receivers is represented in Figure 10.
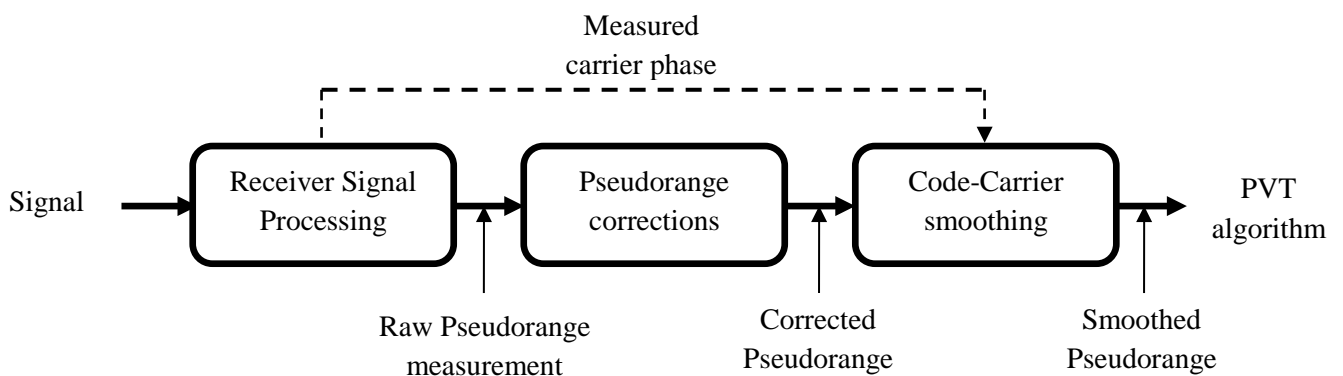
Figure 10: Elaboration of GNSS measurements

In the following sections we will describe these different quantities and we will present the error models which we used to represent the impact of errors affecting GNSS signals. As a general rule, we decided to favour as much as possible standardized models widely accepted by the international community when such models are available and adequate for the purpose of our studies.

We have to remark that in this study we only considered code pseudorange measurements made by the DLL even if another measurement is made in the receiver by the carrier tracking loops which is the carrier phase measurement.

The raw code and carrier phase measurements made by a receiver u for a given satellite i at epoch k can be modelled as:

$$P^i(k) = \rho^i(k) + c\left(\Delta t_u(k) - \Delta t_S^i(k)\right) + c\left(I^i(k) + T^i(k)\right) + D_{multi}^i(k) + n_P^i(k)$$

(4-1)

$$\varphi^i(k) = \rho^i(k) + c\left(\Delta t_u(k) - \Delta t_S^i(k)\right) - c\left(I^i(k) - T^i(k)\right) + \varphi_{multi}^i(k) + \lambda N^i + n_\varphi^i(k)$$

(4-2)

Where:

- P^i is the code pseudorange measurement in meters
- φ^i is the carrier phase measurement in meters
- ρ^i is the geometrical distance between the receiver and the satellite i.
- I^i is the ionospheric propagation delay in seconds.
- T^i is the tropospheric propagation delay in seconds.
- D_{multi}^i is the code pseudorange measurement error induced by multipath propagation.
- φ_{multi}^i is the phase measurement error due to multipath propagation.
- λ is the carrier wavelength.
- N^i is the carrier phase measurement ambiguity, constant over time as long as the PLL does not lose lock on the signal.
- n_P^i is the code phase thermal noise error
- n_φ^i is the carrier phase thermal noise error

In the next sections we will present the error models that we used in this study. First, we will present the models that were used to model the pseudorange received by the receiver. Then, we will present two different pseudorange measurement error models that we used depending on the type of augmentation considered which are:

- Receiver Autonomous Integrity Monitoring (RAIM)
- Ground Based Augmentation System (GBAS)

In fact, as we have seen previously, RAIM does not provide correction for pseudorange measurements while GBAS provide differential corrections. Thus, pseudorange measurement error models considered for RAIM are the models corresponding to GPS standalone.

Additional augmentation models of SBAS and GALILEO GIC have been developed but we did not have the opportunity to use them in the frame of the current study. However, they will be used for future researches.

4.2 Pseudorange measurement model

4.2.1 Pseudorange measurement error models

The models presented in this section are used to generate the pseudorange at the input of the receiver.

4.2.1.1 *Ionospheric delay*

4.2.1.1.1 General considerations

The ionospheric delay model is used to represent the total error affecting pseudorange measurements due to the propagation of the signal through the ionosphere. Ionospheric error is the most severe error affecting GNSS measurements since it is very complex to model it and therefore it is difficult to

predict the delay so as to correct measurements. However, we will see further in this document that some properties of the ionosphere can be used to nearly cancel its effects on GNSS measurements when dual frequency measurements are available.

In this study, we focus on the nominal effects of the ionosphere on the group delay of GNSS signals. Abnormal behaviour of the ionosphere is not considered and in particular ionospheric storms and ionospheric scintillations are not studied.

The ionosphere is a dispersive medium which is located between 60 km and 1000 km above the earth's surface, in the atmosphere. In this area ultraviolet rays coming from the sun ionize a portion of gas molecules and thus, it releases free electrons. These electrons influence the propagation of the electromagnetic waves and thus, the GNSS signals [LEICK, 1995].

More precisely, the propagation velocity of GNSS signals through the ionosphere depends on their frequency and the total electron content (TEC, in e/m2) integrated along the LOS. It represents the number of free electrons in a 1 m² column along the LOS. It can be modelled as [LEICK, 1995]:

$$TEC = \int_{LOS\,path} N_e . dS$$

(4-3)

With N_e the local electron density expressed in units of electron per cubic meters.

The group delay can be approximated at the first order as [LEICK, 1995]:

$$I_P \approx \frac{40.3}{f^2} TEC \quad m$$

(4-4)

For simulation purpose, it was necessary to model the total ionospheric delay so as to build the pseudorange. Different models have been identified and compared:

IRI-2001/IRI-2007:

> In [RTCA, 2004], Appendix R, it is specified that to model ionosphere error, the model IRI-2001 should be used:

> *"Ionospheric error shall be modelled using the International Reference Ionosphere 2001 (IRI-2001) model. The IRI-2001 model was developed and validated by IRI, an international project sponsored by the Committee on Space Research (COSPAR) and the International Union of Radio Science (URSI). It models the ionospheric daily variation but does not model storms. Since statistical data is needed from a deterministic model, the model inputs will be randomized in the test procedures. IRI-2001 accounts for temporal and spatial correlation between satellite measurements. Other iono models may be used but they must be validated."*

> The procedure that should be used to model storm conditions is the following:

> *"In addition to the ionospheric daily variation modelled by IRI, recorded storm data shall be used as the basis for the ionospheric component of pseudorange error in some trials."*

IRI-2007 is an updated version of IRI-2001. The main limitation of this model for us is that it is very complex to use and we do not have the necessary background to properly use it. Moreover, in the frame of our study and since it is a deterministic model it would be necessary to randomize the input parameters so as to obtain a statistical model (as it is mentioned above). Finally, the objective was to integrate a ionospheric error model in our simulator which would be very complex to do with the IRI-model and that's we decided to consider other solutions.

GPS/GALILEO correction algorithms:

For single frequency receivers, ionospheric models can be used so as to correct a fraction of the ionospheric delay, using as inputs parameters broadcasted in the navigation message.

In the case of GPS, the model specified in [ARINC, 2004] is called the Klobuchar algorithm. This model is reputed to allow removing at least 50% of the total ionospheric error. It is easy to find in the appropriate frame of the navigation message the input parameters of this model.

In the case of GALILEO, a more recent model should be used which is called the Nequick algorithm [ARBESSER, 2006]. It is supposed to have better performance than Klobuchar algorithm and is assumed to correct 70% of the ionospheric delay when operating on E5a, E5b, and E1 frequencies [GJU, 2010]. However, this model is a little more complex, and since GALILEO is not in service, there is not enough information available at the moment of the development of our software.

The final solution chosen in the frame of this study is to use Klobuchar model to generate ionospheric delays because of its simplicity and the availability of large amounts of data. We remind hereafter this algorithm.

4.2.1.1.2   Klobuchar Algorithm

Klobuchar algorithm can be found in [KLOBUCHAR, 1987] or [ARINC, 2004]. It is a thin shell model. Eight coefficients are transmitted to the users through the navigation message. The following notations are used to define the algorithm:

Satellite transmitted terms:

- α_n, β_n are the satellite transmitted data words with $n = 0, 1, 2 \text{ and } 3$

Receiver generated terms:

- E the elevation angle between the user and satellite (semi-circles)
- A the azimuth angle between the user and satellite, measured clockwise positive from the true north (semi-circles)
- ϕ_u user geodetic latitude (semi-circles) WGS-84
- λ_u user geodetic longitude (semi-circles) WGS-84
- $GPS\ time$ receiver computed system time

Computed terms:

- X phase (radians)
- F obliquity factor (dimensionless)
- t local time (sec)

- ϕ_m geomagnetic latitude of the earth projection of the ionospheric intersection point (mean ionospheric height assumed 350 km) (semi-circles)
- λ_i geodetic longitude of the earth projection of the ionospheric intersection point (semi-circles)
- ϕ_i geodetic latitude of the earth projection of the ionospheric intersection point (semi-circles)
- ψ earth's central angle between the user position and the earth projection of ionospheric intersection point (semi-circles)

$$\psi = \frac{0.0137}{E + 0.11} - 0.022 \quad (semi-circles)$$

$$\lambda_i = \lambda_u + \frac{\psi \sin A}{\cos \phi_i} \quad (semi-circles)$$

$$\phi_i = \begin{cases} \phi_u + \psi \cos A, & |\phi_u| \leq 0.416 \\ if\ \phi_i > +0.416, & then\ \phi_i = +0.416 \\ if\ \phi_i < -0.416, & then\ \phi_i = -0.416 \end{cases} \quad (semi-circles)$$

$$\phi_m = \phi_i + 0.064 \cos(\lambda_i - 1.617) \quad (semi-circles)$$

$$t = \begin{cases} 4.32(10^4)\lambda_i + GPS\ Time, & if\ 0 \leq t < 86400 \\ 4.32(10^4)\lambda_i + GPS\ Time - 86400, & if\ t \geq 86400 \\ 4.32(10^4)\lambda_i + GPS\ Time + 86400, & if\ t < 0 \end{cases} \quad (sec)$$

$$F = 1.0 + 16.0[0.53 - E]^3$$

$$PER = \begin{cases} \sum_{n=0}^{3} \beta_n \phi_m^n, & PER > 72000 \\ if\ PER < 72000, & PER = 72000 \end{cases} \quad (sec)$$

$$x = \frac{2\pi(t - 50400)}{PER} \quad (rad)$$

$$AMP = \begin{cases} \sum_{n=0}^{3} \alpha_n \phi_m^n, & AMP > 0 \\ if\ AMP < 0, & AMP = 0 \end{cases} \quad (sec)$$

We can then obtain the final ionospheric group delay as:

$$I_{L1}^p(t) = \begin{cases} cF\left[5 \times 10^{-9} + AMP\left(1 - \frac{x^2}{2} + \frac{x^4}{24}\right)\right], & if\ |x| < 1.57 \\ cF(5 \times 10^{-9}), & if\ |x| > 1.57 \end{cases} \quad (meters)$$

Table 5: Klobuchar algorithm [ARINC, 2004]

4.2.1.2 Tropospheric delay

Troposphere is a non-dispersive medium. It is the lowest part of the Earth's atmosphere and it is composed of dry gases and water vapour. These two elements affect the signal propagation velocity which results in range errors for pseudorange measurements. Water vapour density varies widely with position and time and is difficult to predict.

The impact of the propagation of GNSS signals through troposphere is a well known phenomenon and it can be modelled quite efficiently. The model used in civil aviation GPS receivers so as to correct the tropospheric delay can be found in [RTCA, 2006]. Moreover, for civil aviation GALILEO receivers, it is specified in [EUROCAE, 2007] that the GALILEO receiver shall apply a tropospheric correction which is at least as good as the one defined in [RTCA, 2006]. Consequently, this model is a reference for both GPS and GALILEO receivers. We will use it to model the true tropospheric delay.

It is considered that ever since tropospheric delay is a local phenomenon, each user shall compute its own tropospheric delay. The tropospheric delay related to the satellite i can be modeled as:

$$Tr_i = -(d_{hyd} + d_{wet}).m(El_i)$$

(4-5)

Where:

- d_{hyd} is the estimated vertical range delay (i.e. for a satellite at 90° elevation angle) induced by gases in hydrostatic equilibrium in meters
- d_{wet} is the estimated vertical range delay caused by water vapour in meters
- El_i the satellite elevation angle
- $m(El_i)$ is a mapping function that scales the delays to the actual satellite elevation angle

d_{hyd} and d_{wet} are computed from the receiver height's and estimates of five meteorological parameters: pressure P (mbar), temperature T, water vapor pressure e (mbar), temperature lapse rate β (K/m), and water vapor lapse rate λ (dimensionless).

The five meteorological parameters applicable to the receiver latitude ϕ and day of year D are computed on the basis of fixed average and seasonal variation values presented in Table 6 and Table 7.

Latitude (°)	P_0 (mbar)	T_0 (K)	e_0 (mbar)	β_0 (K/m)	λ_0
≤ 15	1013.25	299.65	26.31	6.30e-3	2.77
30	1017.25	294.15	21.79	6.05e-3	3.15
45	1015.75	283.15	11.66	5.58e-3	2.57
60	1011.75	272.15	6.78	5.39e-3	1.81
≥ 75	1013.00	263.65	4.11	4.53e-3	1.55

Table 6: Average meteorological parameters for tropospheric delay [RTCA, 2006]

Latitude (°)	ΔP (mbar)	ΔT (K)	Δe (mbar)	$\Delta \beta$ (K/m)	$\Delta \lambda$
≤ 15	0.00	0.00	0.00	0.00e-3	0.00
30	-3.75	7.00	8.85	0.25e-3	0.33
45	-2.25	11.00	7.24	0.32e-3	0.46
60	-1.75	15.00	5.36	0.81e-3	0.74
≥ 75	-0.50	14.50	3.39	0.62e3	0.30

Table 7: Seasonal variation of meteorological parameters for tropospheric delay [RTCA, 2006]

In the following equations we will use the general symbol ξ to represent one of the five meteorological parameters since the operations are the same for each.

For latitudes $|\phi| \leq 15°$ and $|\phi| \geq 75°$, values ξ_0 and $\Delta\xi$ are taken directly from Table 6 and Table 7. For latitudes in the range $15° < |\phi| < 75°$, values ξ_0 and $\Delta\xi$ are calculated by linear interpolation between values of the two closest latitudes ϕ_i and ϕ_{i+1} :

$$\xi_0(\phi) = \xi_0(\phi_i) + \big(\xi_0(\phi_{i+1}) - \xi_0(\phi_i)\big)\frac{(\phi - \phi_i)}{(\phi_{i+1} - \phi_i)}$$

$$\Delta\xi(\phi) = \Delta\xi(\phi_i) + \big(\Delta\xi(\phi_{i+1}) - \Delta\xi(\phi_i)\big)\frac{(\phi - \phi_i)}{(\phi_{i+1} - \phi_i)}$$

(4-6)

Values of the five meteorological parameters at the actual latitude and Day of the Year are then computed using the following equation:

$$\xi(\phi, D) = \xi_0(\phi) - \Delta\xi(\phi).\cos\left(\frac{2\pi(D - D_{min})}{365.25}\right)$$

(4-7)

With $D_{min} = 28$ for northern latitudes, $D_{min} = 211$ for southern latitudes.

Zero-altitude zenith delay can then be computed:

$$z_{hyd} = \frac{10^{-6}k_1 R_d P}{g_m}$$

(4-8)

$$z_{wet} = \frac{10^{-6}k_2 R_d}{g_m(\lambda + 1) - \beta R_d}.\frac{e}{T}$$

(4-9)

Where

- $k_1 = 77.604$ K/mbar
- $k_2 = 382000$ K²/mbar
- $R_d = 287.054$ J/(kg.K)
- $g_m = 9.784$ m/s²

d_{hyd} and d_{wet} are then derived from these values :

$$d_{hyd} = \left(1 - \frac{\beta H}{T}\right)^{\frac{g}{\beta R_d}} . z_{hyd}$$

(4-10)

$$d_{wet} = \left(1 - \frac{\beta H}{T}\right)^{\frac{g(\lambda+1)}{\beta R_d} - 1} . z_{wet}$$

(4-11)

With

- $g = 9.80665$ m/s²
- H is the receiver's height above mean-sea-level

To obtain the final delay, there is still to compute the mapping function. It has two different expressions:

- A simplified one valid for satellite elevation angle $> 4°$

$$m(E_i) = \frac{1.001}{\sqrt{0.002001 + \sin^2(E_i)}}$$

(4-12)

- An alternate calculation valid for satellite elevation angle $> 2°$

$$m(E_i) = \left(\frac{1.001}{\sqrt{0.002001 + \sin^2(E_i)}}\right) . \left(1 + 0.015 . \left(MAX\begin{bmatrix}0 \\ 4° - E_i\end{bmatrix}\right)^2\right)$$

(4-13)

4.2.1.3 *Clock errors*

The model used for Clock errors are based on [WINKEL, 2000]. This model is based on a stochastic differential equation modelling the oscillator errors. Readers can refer to this paper for detailed theory. A variety of oscillators and their associated characteristics are presented in this document and reported here:

Oscillator	White frequency noise h_0	Flicker h_{-1}	Integrated frequency noise h_{-2}
Standard quartz	2.10^{-19} s	7.10^{-21} s	2.10^{-20} Hz
TCXO	1.10^{-21} s	1.10^{-20} s	2.10^{-20} Hz
OCXO[1]	8.10^{-20} s	2.10^{-21} s	4.10^{-23} Hz
OCXO[2]	$2.51.10^{-26}$ s	$2.51.10^{-23}$ s	$2.51.10^{-22}$ Hz
Rubidium[1]	2.10^{-20} s	7.10^{-24} s	4.10^{-29} Hz
Rubidium[2]	1.10^{-23} s	1.10^{-22} s	$1.3.10^{-26}$ Hz
Cesium[1]	1.10^{-19} s	1.10^{-25} s	2.10^{-32} Hz
Cesium[2]	2.10^{-20} s	7.10^{-23} s	4.10^{-29} Hz

Table 8: Parameters for the Allan variance of several oscillators [WINKEL, 2000]

The Allan variance can be defined as [WINKEL, 2000]:

$$A_{\sigma_y^2(\tau)} = \frac{h_0}{2\tau} + 2\ln 2 h_{-1} + \frac{2\pi^2}{3}\tau h_{-2}$$

(4-14)

Where

- y is defined as

$$y = \frac{\delta f}{f}$$

- δf is the absolute frequency deviation
- f is the nominal frequency of the oscillator

All calculations made, the expression of the stochastic differential equation is [WINKEL, 2000]:

$$\frac{d}{dt}\begin{bmatrix} X^1 \\ X^2 \\ X^3 \\ X^4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{\omega_0^2}{\sqrt{3}} & -\sqrt{\frac{2}{\sqrt{3}}}\omega_0 \end{bmatrix}\begin{bmatrix} X^1 \\ X^2 \\ X^3 \\ X^4 \end{bmatrix} + \begin{bmatrix} \sqrt{h_0/2}\,\varepsilon_0 \\ \pi\sqrt{2h_{-2}}\varepsilon_{-2} \\ 0 \\ \frac{2}{\sqrt{3}}\sqrt{\pi h_{-1}\omega_0}\varepsilon_{-1} \end{bmatrix}$$

(4-15)

With:

- $\varepsilon_0, \varepsilon_{-1}, \varepsilon_{-2}$ independent Gaussian noise processes of variance 1
- $\omega_0 = 2\pi\sqrt{\frac{h_{-2}}{h_0}}$

Therefore, we obtain [WINKEL, 2000]:

- The total time fluctuation given by $X^1 + X^3$
- The total frequency fluctuation given by $X^2 + X^4$

4.2.1.4   Noise

We have seen in Chapter 3 the impact of thermal noise on the correlator outputs. The chosen approach in this study is not to model the noise on the pseudorange but to directly generate the associated correlated noise on the correlator outputs.

We can remind here the correlator outputs expressions for Early, Late, Prompt In-Phase and Quadra-Phase channels presented in Chapter 3 in the case where the Doppler residual is considered null assuming that the acquisition is successfully achieved:

$$
\begin{cases}
I_E(k) = \dfrac{A}{2} . d(k) . \widetilde{K}_c \left(\varepsilon_\tau + \dfrac{C_s}{2}\right) . \cos(\varepsilon_\varphi) + n_{I_E}(k) \\[2mm]
I_P(k) = \dfrac{A}{2} . d(k) . \widetilde{K}_c(\varepsilon_\tau) . \cos(\varepsilon_\varphi) + n_{I_P}(k) \\[2mm]
I_L(k) = \dfrac{A}{2} . d(k) . \widetilde{K}_c \left(\varepsilon_\tau - \dfrac{C_s}{2}\right) . \cos(\varepsilon_\varphi) + n_{I_L}(k) \\[2mm]
Q_E(k) = \dfrac{A}{2} . d(k) . \widetilde{K}_c \left(\varepsilon_\tau + \dfrac{C_s}{2}\right) . \sin(\varepsilon_\varphi) + n_{Q_E}(k) \\[2mm]
Q_P(k) = \dfrac{A}{2} . d(k) . \widetilde{K}_c(\varepsilon_\tau) . \sin(\varepsilon_\varphi) + n_{Q_P}(k) \\[2mm]
Q_L(k) = \dfrac{A}{2} . d(k) . \widetilde{K}_c \left(\varepsilon_\tau - \dfrac{C_s}{2}\right) . \sin(\varepsilon_\varphi) + n_{Q_L}(k)
\end{cases}
$$

$$(4\text{-}16)$$

- $\widetilde{K}_c$ is the correlation function between the locally generated spreading code with the filtered incoming spreading code
- $\varepsilon_\varphi = \varphi - \hat{\varphi}$ the carrier phase error between the incoming carrier phase φ and the locally generated carrier phase $\hat{\varphi}$
- $\varepsilon_\tau = \tau - \hat{\tau}$ is the code delay error between the incoming code delay τ and the locally generated code delay $\hat{\tau}$
- ε_f is the frequency error between the incoming carrier and the locally generated carrier,
- T_I is the correlation time
- C_s is the chip spacing
- $n_{I_E}(k)$, $n_{I_P}(k)$, $n_{I_L}(k)$ are respectively the Early, Prompt and Late In-Phase post-correlation thermal noise components.
- $n_{Q_E}(k)$, $n_{Q_P}(k)$, $n_{Q_L}(k)$ are respectively the Early, Prompt and Late Quadra-Phase post-correlation thermal noise components. They are uncorrelated with In-Phase thermal noise components.
- The power of the prompt thermal noise components can be computed as [JULIEN, 2010]:

$$
Var\big[n_{I_P}\big] = Var\big[n_{Q_P}\big] = \frac{N_0}{4T_I}\beta
$$

- $N_0/2$ is the equivalent double-sided thermal noise PSD at the correlator input.
- $\beta = \int_{-\infty}^{+\infty}|H(f)|^2 \, |F_L(f)|^2 df$ is the loss of noise power due to the front-end filter.
- H is the equivalent front-end filter transfer function

$F_L(f)$ is the Fourier transform of the locally generated code sequence

$$(4\text{-}17)$$

The idea is therefore to directly generate the noises $n_{I_E}, n_{I_P}, n_{I_L}, n_{Q_E}, n_{Q_P}, n_{Q_L}$. The method used is briefly described in [JULIEN, 2005] and it is based on the theory exposed in [HURST,1972] to generate correlated Gaussian noise. Indeed, it can be shown that a vector of correlated random variables can be related to a vector of non correlated random variables:

$$N_{Corr} = V_C \cdot \sqrt{D_C} \cdot N_{Norm}$$

(4-18)

With

- N_{Corr} the vector of correlated Gaussian noise
- C a positive definite matrix representing the expected correlation between noise on Early, Promp and Late channels
- V_C gathers the eigenvectors of C
- D_C is the diagonal matrix containing the eigenvalues of C so that $C \cdot D_C = D_C \cdot V_C$
- N_{Norm} represents Gaussian noise

According to [JULIEN, 2005] the expected correlation between noise on Early, Promp and Late channels is exactly given by the filtered autocorrelation function. Therefore, in the case of Early, Prompt and Late correlators we have for the in-phase components:

$$N_{corr} = \begin{bmatrix} n_{I_E} \\ n_{I_P} \\ n_{I_L} \end{bmatrix}$$

$$C = \begin{bmatrix} \widetilde{K}_c(0) & \widetilde{K}_c\left(\dfrac{C_s}{2}\right) & \widetilde{K}_c(C_s) \\ \widetilde{K}_c\left(-\dfrac{C_s}{2}\right) & \widetilde{K}_c(0) & \widetilde{K}_c\left(\dfrac{C_s}{2}\right) \\ \widetilde{K}_c(-C_s) & \widetilde{K}_c\left(-\dfrac{C_s}{2}\right) & \widetilde{K}_c(0) \end{bmatrix}$$

(4-19)

Where C_s is the chip spacing.

We can see that C is positive definite as expected. This model can therefore be used by generating samples of uncorrelated gaussian noise samples, then rotated by using the eigenvectors and the eigenvalues of the correlation matrix C. The appropriate variance is applied to the noise depending on the signal to noise ratio.

4.2.1.5 Multipath
This section addresses the impact of multipath on the GNSS signal observables during aircraft approaches. Indeed, the combination of the direct and reflected signals at the receiver antenna output induces measurements not representing solely the direct signal.

As our approach is to model the correlator outputs, we directly modelled the impact of multipath on the correlator outputs. The model used to determine the characteristics of the simulated multipath is the High Resolution Aeronautical Channel model developed by the DLR. [STEINGASS ET AL., 2004]

4.2.1.5.1 Correlator outputs in presence of multipath

The signal received at the receiver antenna in presence of multipath is the sum of the LOS (Line Of Sight) signal and of multipath rays.

If we consider that the LOS signal can be written as:

$$s_0(t) = a_0 . d(t) . c(t) . \cos(2\pi f_0 t + \varphi)$$

<div align="right">(4-20)</div>

Where:

- a_0 is the amplitude of the LOS signal
- $d(t)$ is the data sequence waveform
- $c(t)$ is the code sequence waveform
- $\cos(2\pi f_0 t)$ is the carrier of the signal

We can define each multipath ray s_m^i as follows:

$$s_m^i(t) = a^i . d\big(t + \delta\tau^i\big) . c\big(t + \delta\tau^i\big) . \cos\big(2\pi f_0\big(t + \delta\tau^i\big) + \delta\varphi^i\big)$$

<div align="right">(4-21)</div>

- a^i : the amplitude of the multipath component
- $\delta\tau^i$: relative delay
- $\delta\varphi^i$: relative phase shift

Then the composite received signal in presence of n multipath is:

$$s(t) = s_0(t) + \sum_{i=1}^{n} s_m^i(t)$$

<div align="right">(4-22)</div>

The idea now is to compute the correlation function of the composite signal $s(t)$ with the locally generated code and carrier replicas. In fact, we can see that the result will be the sum of the correlation of each component of $s(t)$ with the locally generated signals.

The correlation between a reflected signal with a relative delay $\delta\tau^i$ and the Early, Punctual and Late local replicas can be modelled as:

$$K_E^i \cong \widetilde{K}_c\left(\varepsilon_\tau + \delta\tau^i + \frac{C_s}{2}\right)$$

$$K_P^i \cong \widetilde{K}_c\big(\varepsilon_\tau + \delta\tau^i\big)$$

$$K_L^i \cong \widetilde{K}_c\left(\varepsilon_\tau + \delta\tau^i - \frac{C_s}{2}\right)$$

<div align="right">(4-23)</div>

Where:

- $\widetilde{K}_c$ is the correlation between the filtered incoming code and the locally generated replica
- K_E^i, K_P^i, K_L^i are respectively the early, prompt and late code correlation values for multipath i

Using expression (4-17) with the same assumptions, and since the correlation process is linear, if we consider that the received signal is the sum of the line of sight signal and the reflected signals we obtain:

$$
\begin{cases}
I_E = a_0.d(k).K_E.\cos(\varepsilon_\theta) + \displaystyle\sum_{i=1}^{n} a^i.d(k).K_E^i.\cos(\varepsilon_\varphi + \delta\varphi) \\[2mm]
I_P = a_0.d(k).K_P.\cos(\varepsilon_\theta) + \displaystyle\sum_{i=1}^{n} a^i.d(k).K_P^i.\cos(\varepsilon_\varphi + \delta\varphi^i) \\[2mm]
I_L = a_0.d(k).K_L.\cos(\varepsilon_\theta) + \displaystyle\sum_{i=1}^{n} a^i.d(k).K_L^i.\cos(\varepsilon_\varphi + \delta\varphi^i) \\[2mm]
Q_E = a_0.d(k).K_E.\sin(\varepsilon_\theta) + \displaystyle\sum_{i=1}^{n} a^i.d(k).K_P^i.\sin(\varepsilon_\varphi + \delta\varphi^i) \\[2mm]
Q_P = a_0.d(k).K_P.\sin(\varepsilon_\theta) + \displaystyle\sum_{i=1}^{n} a^i.d(k).K_L^i.\sin(\varepsilon_\varphi + \delta\varphi^i) \\[2mm]
Q_L = a_0.d(k).K_L.\sin(\varepsilon_\theta) + \displaystyle\sum_{i=1}^{n} a^i.d(k).K_E^i.\sin(\varepsilon_\varphi + \delta\varphi^i)
\end{cases}
$$

(4-24)

This model has been used to compute some figures in order to illustrate the impact of a unique multipath with different characteristics and on different signals. The results can be found in the following figures.
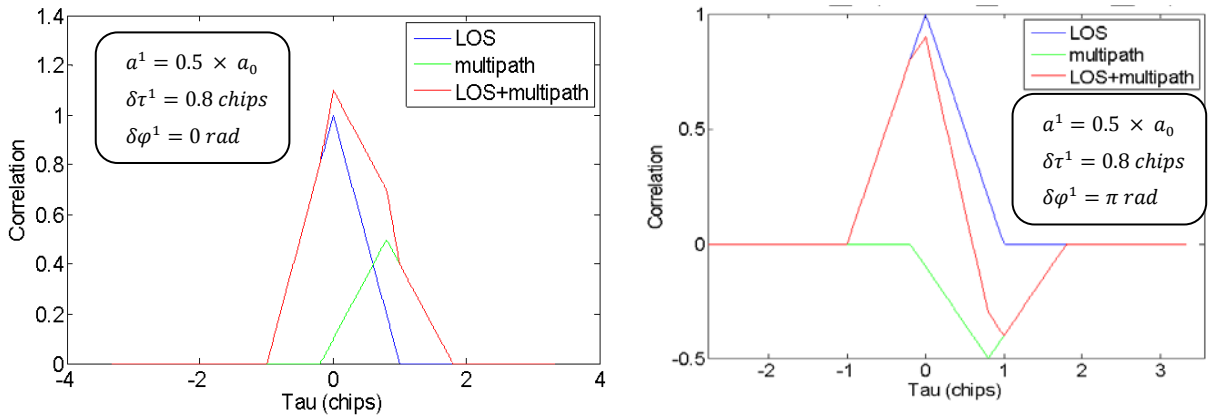
Figure 11 and Figure 12: Correlation function in presence of one multipath for a BPSK(1) signal
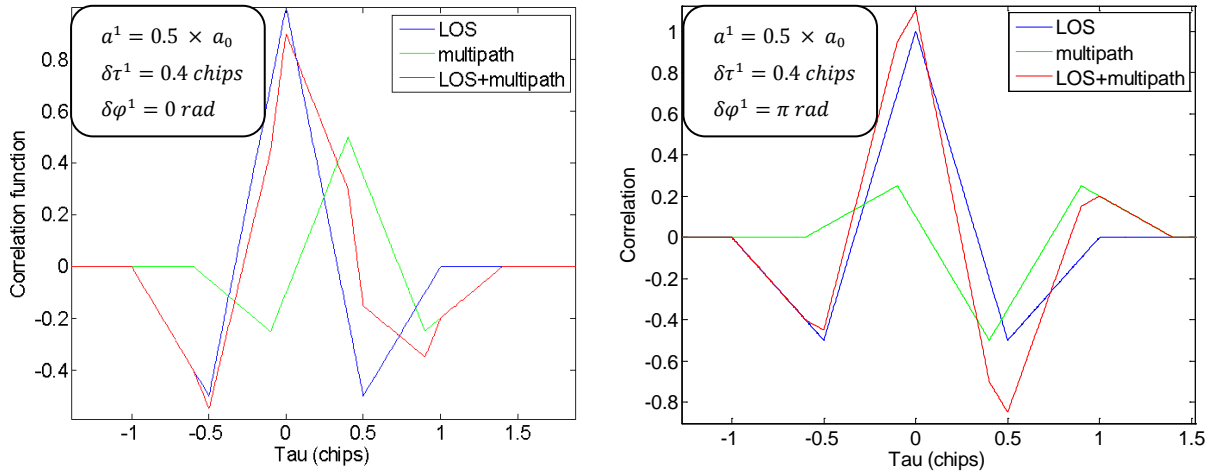
Figure 13 and Figure 14: Correlation function in presence of one multipath for a BOC(1,1) signal

We can clearly see here that a multipath lead to a bias in the correlation function and thus in the final pseudorange measurement. We have represented the impact of a known multipath on the correlator outputs. We now present the model used to predict the characteristics of multipath affecting the pseudorange measurements which are as already said, the relative amplitude, the relative delay, and the relative phase shift.

4.2.1.5.2   DLR model for multipath

The multipath error depends on the environment, the antenna and the receiver signal processing. The characterization of the multipath error is therefore very complex in particular because the geographic environment very difficult or impossible to model. However, to properly take it into account into error budgets, different studies have been led to compute the statistics of the error due to multipath.

This work was already tackled by RTCA for GPS L1 C/A code users, and the final result was a standard curve adopted in the ICAO SARPs stating the standard deviation of the error due to multipath as a function of the GPS satellite elevation angle [BOOTH, 2000]. This curve will be presented further in this document. The work done to validate this standard multipath error expression is described in [MURPHY, 2000].

However, a more recent study has been run to determine the impact of multipath on new GNSS signals. Preliminary results have shown that smaller results can be expected for new signals [MACABIAU et al., 2006]. To obtain these conclusions, they used the High Resolution Aeronautical Channel model developed for ESA and injected the output multipath parameters into a software receiver. The same method has been used during our work because it allows a more detailed representation of the multipath impact on the measurements. This method also allows a proper model representing the time evolution of the error due to multipath.

The model was initially presented in [STEINGASS ET AL., 2004]. We only make here a general presentation of the model. This model allows generating three multipath components which are:

- Path 0: A direct path
- Path 1: A refractive component of the direct path
- Path 2: A strong echo on the fuselage that is changing very slowly
- Path 3: A quickly changing ground echo

These multipath were used to generate multipath error on the correlator outputs of our software receiver using the expressions presented in the previous section.

71

4.2.1.6 Interferences

Other errors have been represented in the software receiver developed during this PhD which are Wide Band and CW interferences. However, as they have not been used for the simulations presented in this report, the models used will not be presented here but in the appendices.

4.2.2 RAIM Pseudorange measurement model

In this section we will present the models used to represent the errors affecting the RAIM pseudorange measurements. Systematic errors will be gathered in the fault free pseudorange measurement model whereas unusual errors that may cause hazardous positioning failure and which should be detected are described in the faulty pseudorange measurement model.

4.2.2.1 Fault free pseudorange measurement model

In the fault free case, we assume that the pseudorange measurement error components have a normal distribution with a zero mean. The resulting pseudorange error distribution is thus considered to be the result of the combination of all the errors and therefore, it is supposed to have a normal distribution with zero mean and a standard deviation which is the quadratic sum of all error standard deviations. We present here the different error components distributions as well as the final pseudorange error distribution model. We will also present assumptions concerning the temporal evolution of these errors.

4.2.2.1.1 Troposphere

We already presented the correction model used in civil aviation receivers for tropospheric error in section 4.2.1.2. The associated model for residual tropospheric error is:

$$\sigma_{i,tropo} = \sigma_{TVE}.m(E_i) \qquad \text{(Meters)}$$

$$\text{(4-25)}$$

With:

- the tropospheric vertical error $\sigma_{TVE} = 0.12 \ meters$.
- $m(E_i)$ is a mapping function that scales the delays to the actual satellite elevation angle (see section 4.2.1.2)

It is also specified in [RTCA, 2006] that tropospheric error should be modelled using a first-order Gauss-Markov process with a 30 minute correlation time and scaled per the tropospheric residual error sigma.

4.2.2.1.2 Ionosphere

For ionosphere error two cases have been taken into account:

- Single frequency measurements
- Dual frequency measurements

In the nominal mode, future civil aviation GNSS receivers will use dual frequency measurements. However, it is interesting to model single frequency measurements so as to model situations where a receiver will be forced to switch to degraded mode, in case of loss of a frequency due to satellite failure or signal outages for example.

4.2.2.1.2.1 *Single frequency measurements*

In the case of single frequency measurements, the receivers are supposed to use the ionospheric correction models presented in section 4.2.1.1.

GPS single frequency model: In the case of GPS L1 C/A receivers, the single frequency model used to correct ionospheric contribution to measurements error is the Klobuchar model. The residual error in this case can be modelled as:

$$\sigma^2_{iono,GPS} = MAX\left(\left(\frac{cT_{iono}}{5}\right)^2, \left(F_{pp}.\tau_{vert}\right)^2\right)$$

(4-26)

Where:

- C is the speed of light in a vacuum in m/s
- T_{iono} is the ionospheric correction in m
- F_{pp} is the obliquity factor (unitless)
- $\tau_{vert} = \begin{cases} 9\ m, & 0 \leq |\phi_m| \leq 20 \\ 4.5\ m, & 20 < |\phi_m| \leq 55 \\ 6\ m, & 55 < |\phi_m| \end{cases}$
- ϕ_m is the geomagnetic latitude in degrees

See section 4.2.1.1 for the definition of these terms.

GALILEO single frequency model: GALILEO single frequency receivers estimate the ionospheric delay using the NeQuick model. The error specifications of this model are not to exceed 30 % of the current ionospheric delay or the equivalent delay due to TEC (Total Electron Content) of 20 TECu, whichever is larger [SALOS, 2010]:

$$\sigma_{iono,Galileo} = max\left\{0.3\Delta_{iono}, \frac{40.3}{f^2}.20.10^{16}\right\}\ (m)$$

(4-27)

Substituting the ionospheric delay using the first order approximation model we obtain:

$$\sigma_{iono,Galileo} = \frac{40.3}{f^2}.10^{16}max\{0.3 \times F \times VTEC, 20\}\ (m)$$

(4-28)

Where:

- VTEC is the vertical TEC (e/m2)
- F is the obliquity factor defined in section 4.2.1.1 (unit less)

In [SALOS, 2010], numerical values of the residual ionospheric delay have been computed using the International GNSS Service (IGS) VTEC database. In our software we used these numerical values in the case of GALILEO single frequency receiver. These results are represented in Figure 15.
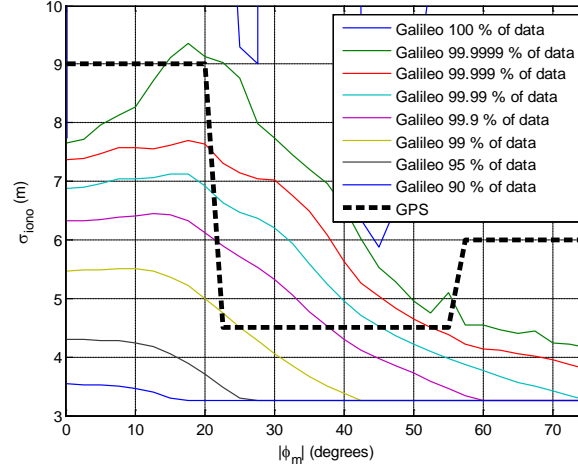
Figure 15: GALILEO and GPS ionospheric residual error model [SALOS, 2010]

4.2.2.1.2.2 Dual Frequency measurements

Ionosphere is a dispersive medium and therefore the ionosphere propagation delay is dependent on the frequency of the signal. Dual-frequency capable receivers can use this property to remove nearly completely the ionospheric error. To do so, the receiver computes a composite pseudorange which is called the iono-free measurement. If we consider GPS L1-L5 for GPS and GALILEO E1-E5B for GALILEO we obtain:

$$P_{L1-L5}(k) = \frac{f_{L1}^2}{f_{L1}^2 - f_{L5}^2} P_{L1}(k) + \frac{f_{L5}^2}{f_{L5}^2 - f_{L1}^2} P_{L5}(k)$$

$$P_{E1-E5B}(k) = \frac{f_{E1}^2}{f_{E1}^2 - f_{E5B}^2} P_{E1}(k) + \frac{f_{E5B}^2}{f_{E5B}^2 - f_{E1}^2} P_{E5B}(k)$$

$$\varphi_{L1-L5}(k) = \frac{f_{L1}^2}{f_{L1}^2 - f_{L5}^2} \varphi_{L1}(k) + \frac{f_{L5}^2}{f_{L5}^2 - f_{L1}^2} \varphi_{L5}(k)$$

$$\varphi_{E1-E5B}(k) = \frac{f_{E1}^2}{f_{E1}^2 - f_{E5B}^2} \varphi_{E1}(k) + \frac{f_{E5B}^2}{f_{E5B}^2 - f_{E1}^2} \varphi_{E5B}(k)$$

(4-29)

Where:

- $P(k)$ is the code measurement at instant k and $\varphi(k)$ is the phase measurement.
- $\frac{f_{L1}^2}{f_{L1}^2 - f_{L5}^2} = \frac{f_{E1}^2}{f_{E1}^2 - f_{E5A}^2} \approx 2.261$, $\frac{f_{L5}^2}{f_{L5}^2 - f_{L1}^2} = \frac{f_{E5A}^2}{f_{E5A}^2 - f_{E1}^2} \approx -1.261$
- $\frac{f_{E1}^2}{f_{E1}^2 - f_{E5B}^2} \approx 2.422$, $\frac{f_{E5B}^2}{f_{E5B}^2 - f_{E1}^2} \approx -1.422$

This correction allows removing most of the ionospheric delay and therefore the residual error is considered as negligible:

$$\sigma_{iono,dual}^2 \approx 0$$

(4-30)

74

4.2.2.1.3 Receiver noise residual error

4.2.2.1.3.1 Code tracking loop error

The receiver noise residual error is directly linked to code tracking loop error which has already been described in Chapter 3 of this PhD. We remind the expressions presented previously depending on the discriminator used:

$$\sigma_{\tau,DP} = \sqrt{\frac{B_{DLL}\int_{-B/2}^{B/2} G(f)sin^2(\pi f C_s)df}{\frac{C}{N_0}\left(2\pi\int_{-B/2}^{B/2} fG(f)sin(\pi f C_s)df\right)^2}\left(1 + \frac{1}{\frac{C}{N_0}T_{DLL}\left(\int_{-B/2}^{B/2} G(f)df\right)}\right)} \ (m)$$

(4-31)

$$\sigma_{\tau,EMLP} = \sqrt{\frac{B_{DLL}\int_{-B/2}^{B/2} G(f)sin^2(\pi f C_s)df}{\frac{C}{N_0}\left(2\pi\int_{-B/2}^{B/2} fG(f)sin(\pi f C_s)df\right)^2}\left(1 + \frac{\int_{-B/2}^{B/2} G(f)cos^2(\pi f C_s)df}{\frac{C}{N_0}T_{DLL}\left(\int_{-B/2}^{B/2} G(f)cos(\pi f C_s)df\right)^2}\right)} \ (m)$$

(4-32)

Where:

- B is the RF front-end filter bandwidth (Hz)
- C the received signal power over infinite signal bandwidths (W)
- N_0 the power spectral density of the noise (W/Hz)
- $G(f)$ is the power spectral density of the processed signal
- T_{DLL} is the coherent integration time of the DLL (s)
- Δ is the chip spacing (s)
- B_{DLL} is the DLL filter bandwidth (Hz)

The following assumptions have been used for nominal cases:

	GPS L1	GPS L5	GALILEO E1 BOC(1,1)	GALILEO E5A/E5B
B (MHz)	4	24	10	24
T_{DLL} (ms)	20	20	100	100
B_{DLL} (Hz)	1	1	1	1
Cs (chip)	0.2	0.25	0.2	0.25

Table 9: Parameters for DLL tracking error variance computation

4.2.2.1.3.2 Dual frequency measurements

We have seen that to correct the ionospheric delay, dual frequency receivers compute the iono-free pseudorange measurement.

For the other errors affecting the measurements, this technique can have various results on the computed iono-free measurements:

- Errors which are identical for both pseudorange measurements will be directly propagated in the iono-free measurement.

- Errors which are uncorrelated will be combined in the iono-free measurements and it can thus result in an increase of their impact. It is the case for noise and multipath for example.

Therefore, if we denote $\sigma_{noise,f1/f2}$ the standard deviation of the noise error on the iono-free measurement f1/f2 and using (4-29) we can write:

$$\sigma_{noise,L1/L5} = \sqrt{2.261^2 \sigma_{noise,L1}^2 + 1.261^2 \sigma_{noise,L5}^2}$$

(4-33)

$$\sigma_{noise,E1/E5A} = \sqrt{2.261^2 \sigma_{noise,E1}^2 + 1.261^2 \sigma_{noise,E5A}^2}$$

(4-34)

$$\sigma_{noise,E1/E5B} = \sqrt{2.422^2 \sigma_{noise,E1}^2 + 1.422^2 \sigma_{noise,E5B}^2}$$

(4-35)

4.2.2.1.3.3 *Code Carrier smoothing*
Code Carrier smoothing allows reducing the impact of noise and multipath on the pseudorange measurements. It is used either in single frequency receivers or in dual frequency receivers. The main difference is that smoothing in dual frequency receivers will be applied on the iono-free measurements.

The code carrier smoothing algorithm can be found in [RTCA, 2006]:

$$P_{proj} = P_{n-1} + \frac{\lambda}{2\pi}(\phi_n - \phi_{n-1})$$

(4-36)

$$P_n = \alpha \rho_n + (1 - \alpha)P_{proj}$$

(4-37)

Where:

- P_n is the carrier-smoothed pseudorange in meters
- P_{n-1} is the previous carrier-smoothed pseudorange in meters
- P_{proj} is the projected pseudorange in meters
- ρ_n is the raw pseudorange measurement in meters
- λ is the wavelength in meters
- ϕ_n is the accumulated carrier phase measurement in radians
- ϕ_{n-1} is the previous accumulated carrier phase measurement in radians
- α is the filter weighting function (a unit less parameter). Two definitions exist. In [RTCA, 2006] it is equal to the sample interval in seconds divided by the time constant T_{smooth}:

$$\alpha = \frac{T_s}{T_{smooth}}$$

(4-38)

In [RTCA, 2008], an alternative definition is given to be used for GBAS applications, which is:

$$\begin{cases} \alpha = \dfrac{T_s}{t} \ \text{if}\, t < T_{smooth} \\ \alpha = \dfrac{T_s}{T_{smooth}} \ \text{if}\, t > T_{smooth} \end{cases}$$

(4-39)

- t the time since initialization of the smoothing filter in seconds

Assuming the errors affecting the carrier phase measurements can be neglected, and assuming the errors affecting the code measurements can be modelled as white noise, the variance of the error affecting the smoothed code pseudorange can be modelled as [HEGARTY, 2006]:

$$\sigma_{\tilde{P}}^2 \approx \frac{\sigma_P^2}{2T_{smooth}}$$

(4-40)

Where:

- T_{smooth} is the time smoothing constant in seconds
- σ_P^2 is the raw code pseudorange measurement error variance
- $\sigma_{\tilde{P}}^2$ is the smoothed code pseudorange measurement error variance

Therefore we obtain the noise variance for single and dual frequency receivers:

Single frequency:

$$\sigma_{noise}^2 = \frac{\sigma_{noise,f_1}^2}{2T_{smooth}}$$

(4-41)

Dual frequency:

$$\sigma_{noise}^2 = \frac{\sigma_{noise,f_1/f_2}^2}{2T_{smooth}}$$

(4-42)

4.2.2.1.4 Multipath

As previously explained the characterization of multipath has been tackled by RTCA for GPS L1 C/A code users. This study resulted in a standard curve which has been then adopted in the ICAO SARPS. This curve describes the smoothed multipath error variance for airborne equipment as a function of the GPS satellite elevation angle. This model is described in:

$$\sigma_{multipath} = 0.13 + 0.53\, exp(-\theta/10\, deg) \quad \text{(meters)}$$

(4-43)

Where θ is the elevation angle in degrees of the considered satellite.

In the case of dual frequency measurements the same principle as in the case of thermal noise error can be applied. The iono-free measurement multipath error is thus a combination of the multipath error on each signal:

$$\sigma_{multipath,L1/L5} = \sqrt{2.261^2 \sigma^2_{multipath,L1} + 1.261^2 \sigma^2_{multipath,L5}}$$

(4-44)

$$\sigma_{multipath,E1/E5A} = \sqrt{2.261^2 \sigma^2_{multipath,E1} + 1.261^2 \sigma^2_{multipath,E5A}}$$

(4-45)

$$\sigma_{multipath,E1/E5B} = \sqrt{2.422^2 \sigma^2_{multipath,E1} + 1.422^2 \sigma^2_{multipath,E5B}}$$

(4-46)

4.2.2.1.5 Satellite Clock and ephemeris uncertainty

Satellite clock and ephemeris error components will depend on the considered system.

For GPS, User Range Accuracy (URA) is a bound of the standard deviation of the range component of clock and ephemeris error. We assume that the distribution of the ephemeris and clock range error of each satellite is over bounded by a zero mean Gaussian distribution with standard deviation equal to URA.

In the case of GALILEO, we assume that the clock and ephemeris error of each satellite is over bounded by a nonbiased Gaussian distribution with the minimum standard deviation called Signal In Space Accuracy (SISA). This parameter is assumed to have the same definition as the GPS URA. The integrity performance requirement specifies a SISA value for both nominal and degraded mode in [ESA, 2005]. This value is equal to 85 cm.

GPS URA depends on the assumed type of satellite and therefore on the considered modernization step of the GPS constellation. During our study we assumed future GNSS constellations and therefore we made the assumption that both GALILEO and GPS satellites will achieve at least the URA of 0.85 cm:

$$\sigma_{URA,GPS} = \sigma_{URA,GALILEO} = 0.85 \; meters$$

(4-47)

4.2.2.1.6 User Equivalent Range Error (UERE)

The UERE reflects the error budget and is based on the aggregation of the different contributions presented in the previous sections, assumed to be independent of each other:

$$\sigma^2_{UERE} = \sigma^2_{URA} + \sigma^2_{iono} + \sigma^2_{tropo} + \sigma^2_{multipath} + \sigma^2_{noise}$$

(4-48)

4.2.2.2 *Faulty pseudorange measurement model*

A fault is said to occur when a significantly large error in the range measurement (whether that error is due to an anomaly of the satellite itself or to environmental effects on the satellite ranging signal such as multipath or interference) may potentially cause an integrity failure [LEE, 2004] The GPS Standard Positioning Service Performance Standard [GPS SPS, 2008] specifies two parameters regarding the satellite integrity: the definition of a major service failure and the specification of the maximum rate of such a satellite fault.

A major service failure is defined to be a condition during which a healthy SPS SIS's instantaneous URE exceeds the SIS URE NTE tolerance without a timely alert being provided [GPS SPS, 2008]..

The NTE SPS SIS URE tolerance for a healthy SPS SIS is defined to be ± 4.42 times the upper bound on the URA value corresponding to the URA index *N* currently broadcast by the satellite [GPS SPS, 2008].

The probability of occurrence of such an event is 3 per year for a 24 GPS satellites constellation.

4.2.3 GBAS Pseudorange measurement model

This section presents the pseudorange measurement model used for simulations dedicated to GBAS NSE. To begin, we give an overview of the services that may be provided by GBAS equipments and we emphasize on the approach services.

4.2.3.1 *GBAS Service Types*

The GBAS service types (GST) represent matched sets of performance and functional requirements for airborne and ground subsystems allowing providing a particular service.

GBAS can support two different types of services:

- Positioning service [RTCA, 2008]: It allows the user to compute an accurate differential position solution with integrity. Only one type of positioning service is defined.
- Approach service [RTCA, 2008]: It enables the user to compute an accurate differentially corrected position solution, but also includes the definition of a reference path so that the airborne equipment can compute guidance information (deviations) relative to the reference path. Several different GBAS Approach Service Types (GASTs) are defined which provide different levels of performance.

As mentioned above, several GBAS approach services exist which are GASTs A, B, C and D. They correspond respectively to GBAS for APV I, APV II, CAT I and CAT II/III approaches [RTCA, 2008]. It is important to notice here that currently, only GBAS services types A to C are currently described in the SARPS [ICAO, 2007]. GAST D service type is described in [RTCA, 2008] and in ICAO NSP CSG proposed SARPs revisions for GBAS GAST D [ICAO, 2010a].

We focused our study on the GBAS approach services since we were interested in GBAS performance during precision approaches. Thus, we focused in particular on GAST C and GAST D approach services. We present here the pseudorange measurement error models used for these two categories of GBAS services.

GAST-C pseudorange measurement error models are standardized and have been published in [RTCA, 2008] whereas pseudorange measurement error models have not yet been officially published for GAST-D. One of the aims of this GBAS study is in fact to provide information on the statistics of

GBAS NSE for GAST D approach service and some assumptions have been made concerning GAST D pseudorange measurement error models.

4.2.3.2 *GAST-C pseudorange measurement error model*
In DO-245A, we can find the expressions of the standard deviation of the over bounding distributions of the pseudorange measurement errors in the case of GAST-C. The expression of the standard deviation of the pseudorange measurement error i can be written as:

$$\sigma_i^2 = \sigma_{pr\_gnd,i}^2 + \sigma_{pr\_air,i}^2 + \sigma_{tropo,i}^2 + \sigma_{iono,i}^2$$

(4-49)

Where:
- $\sigma_{pr\_gnd,i}^2$ is the total (post correction) fault-free noise term provided by the ground function (via the VDB) for satellite i
- $\sigma_{pr\_air,i}^2$ is the standard deviation of the aircraft contribution to the corrected pseudorange error for the ith satellite. It includes the receiver and airframe multipath contribution.
- $\sigma_{tropo,i}^2$ is the standard deviation of the residual tropospheric error for satellite i.
- $\sigma_{iono,i}^2$ is the standard deviation of the residual ionospheric delay (due to spatial decorrelation) uncertainty for satellite i.

The expressions used to represent these different terms are described below.

4.2.3.2.1 SIS pseudorange accuracy
The SIS pseudorange errors include both those errors due to the non-aircraft elements in the LAAS (Local Area Augmentation System) and the effects of the signal propagation through the atmosphere. The RMS of the total non-aircraft contribution to the GPS/LAAS error as a function of GPS satellite elevation angle at the GBAS reference point shall be:

$$\sigma_{pr\_gnd}(\theta_i) \leq \sqrt{\frac{\left(a_0 + a_1 e^{-\theta_i/\theta_0}\right)^2}{M} + (a_2)^2}$$

(4-50)

Where:

- $M \equiv$ number of ground reference receiver subsystems
- $i \equiv i^{th}$ ranging source
- $\theta_i \equiv$ elevation angle for the ith ranging source
- a_0, a_1, a_2 and θ_0 parameters are defined in Table 10.

The errors are divided into two categories, ground reference receiver errors (such as noise and multipath) and residual differential correction errors (such as spatial decorrelation, temporal decorrelation…):

Ground reference receiver errors:

- Thermal Noise: Derived using a validated analytical bound for carrier-smoothed delay-locked loop, assuming 100-sec smoothing. C/Nt modeled as a function of satellite elevation angle. GPS signal power as specified in the Global Position System Standard Positioning Service

Signal Specification. Ground antenna gain, implementation loss, and correlator spacing as defined in [RTCA, 2004]. Interference levels as defined in [RTCA, 2004] appendix H.
- Ground multipath: Derived from empirical data.

Residual LAAS SIS error:

- Pseudorange acceleration/Data broadcast latency
- Ephemeris
- Ground-to-airborne multipath
- Reference receiver relative survey

The coefficients presented in Table 10 are derived from a Curve fit to the Root-Sum-Square of the previous components.

Ground Accuracy Designator (GAD)	θ_i (degrees)	a_0 (meters)	a_1 (meters)	θ_0 (degrees)	a_2 (meters)
A	> 5	0.5	1.65	14.3	0.08
B	> 5	0.16	1.07	15.5	0.08
C	> 35	0.15	0.84	15.5	0.04
	≤ 35	0.24	0	-	0.04

Table 10: Non-aircraft Elements Accuracy Requirement [RTCA, 2004]

4.2.3.2.2 Airborne pseudorange accuracy
The RMS of the total airborne receiver contribution to the error in a corrected pseudorange for a GPS satellite as a function of satellite elevation angle above the local level plane shall be:

$$\sigma_{pr\_air}(\theta_i) \leq a_0 + a_1 e^{-\theta_i/\theta_0}$$

(4-51)

Where:

- $i = i^{th}$ ranging source
- $\theta_i = $ elevation angle for the ith ranging source
- a_0, a_1 and θ_0 parameters are defined in.

The thermal and interference budget shown in Table 11 is based on the performance of typical early-minus-late correlator receiver in the standard interference environment.

Airborne Accuracy Designator (AAD)	θ_0 (degrees)	a_0 (meters)	a_1 (meters)
A	6.9	0.15	0.43
B	4	0.11	0.13

Table 11: Airborne Accuracy Designator [RTCA, 2004]

$$\sigma_{mp}(\theta_i) \leq a_0 + a_1 e^{-\theta_i/\theta_0}$$

(4-52)

The same formula is used for multipath with different coefficients presented in Table 12.

The airframe multipath model AMD-A shown in Table 12 was developed and validated by the community, based largely on flight data collected by BOEING over the course of several years during normal production flight testing.

Airframe Multipath Designator (AAD)	θ_0 (degrees)	a_0 (meters)	a_1 (meters)
A	10.0	0.13	0.53
B	10.0	0.065	0.265

Table 12: Airframe Multipath Designator [RTCA, 2004]

4.2.3.2.3 Tropospheric residual uncertainty

The residual tropospheric uncertainty is defined by [RTCA, 2008]:

$$\sigma_{tropo}(\theta) = \sigma_N h_0 \frac{10^{-6}}{\sqrt{0.002 + \sin^2(\theta)}} \left(1 - e^{-\frac{\Delta h}{h_0}}\right)$$

(4-53)

Where:

- σ_N is the refractivity uncertainty transmitted by ground subsystem

For our simulations we had to model the residual tropospheric uncertainty which is computed using a transmitted parameter. We decided to use the assumptions proposed for predicting LAAS availability proposed in DO-245A Appendix F.

Therefore we assumed:

$$\sigma_{tropo}(\theta) \approx 0$$

4.2.3.2.4 Ionospheric residual uncertainty

The residual ionospheric uncertainty is defined as [RTCA, 2008]:

$$\sigma_{iono} = F_{pp}.\sigma_{vert\_iono\_gradient\_x}.(x_{air} + 2.\tau.v_{air})$$

(4-54)

- F_{pp} is the vertical-to-slant obliquity factor (unitless) for the given satellites and

$$F_{pp} = \left[1 - \left(\frac{R_e \cos\theta}{R_e + h_I}\right)^2\right]^{-\frac{1}{2}}$$

- $R_e = 6378.1363\ km$ is the radius of the earth
- $h_I = 350\ km$ is the ionospheric shell height

- θ is the elevation angle of the satellite
- $\sigma_{vert\_iono\_gradient\_x}$ is the Service Type dependent standard deviation of a normal distribution associated with the residual ionospheric uncertainty due to spatial decorrelation.
- x_{air} is the distance (slant range) between the aircraft and the GBAS reference point (in meters).
- τ is the Service Type dependent time constant of the smoothing filter.
- v_{air} is the horizontal speed of the aircraft (meters/sec).

Just as for tropospheric residual uncertainty we had to model some of the parameters necessary to compute ionospheric residual uncertainty which are x_{air} and v_{air}. We also used the assumptions presented in DO-245A Appendix F for LAAS availability prediction and which are reminded in Table 13.

	x_{air} (m)	v_{air} (m/s)
GAST C	6000	77
GAST D	5400	72

Table 13: Residual Ionospheric Uncertainty parameters assumptions [RTCA, 2004]

This choice is not optimal since to represent nominal error we should take into account different distances to the runway threshold and different air speeds. It may be impractical to do that since we do not model aircraft trajectory in our GBAS simulations.

4.2.3.3   *GAST-D pseudorange measurement error model*

One of the main evolutions brought by GAST-D with respect to GAST-C is the introduction of the code-carrier smoothing with $\tau = 30s$ instead of 100s. Therefore, the pseudorange measurement error is impacted.

At the time of this study, no expressions of the standard deviations of the pseudorange measurement error have been published in the case of GAST-D.

For Ionospheric and Tropospheric errors it is simple to compute GAST-D standard deviations:

- The expression (4-54) presented for ionospheric residual error for GAST-C contains explicitly the smoothing constant:

$$\sigma_{iono} = F_{pp} \cdot \sigma_{vert\_iono\_gradient\_x} \cdot (x_{air} + 2.\tau.v_{air})$$

- The expression (4-53) presented for tropospheric residual error for GAST-C is a slowly varying process and has a correlation time much higher than the smoothing constant. It can then be considered as not impacted by the new smoothing constant. We remind the equation here:

$$\sigma_{tropo}(\theta) = \sigma_N h_0 \frac{10^{-6}}{\sqrt{0.002 + \sin^2(\theta)}} \left(1 - e^{-\frac{\Delta h}{h_0}}\right)$$

However, for ground and airborne accuracy it is not that simple. The following assumptions have been made and are based on the recommendations of GBAS experts:

- The Airborne accuracy should be multiplied by the factor $\sqrt{100/30}$ to account for the change of carrier smoothing constant leading to:

$$\sigma_{pr\_air\_GASTD}(\theta_i) = \sqrt{100/30} \cdot \sigma_{pr\_air}(\theta_i)$$

(4-55)

With $\sigma_{pr\_air}(\theta_i)$ as presented for GAST C in equation (4-51)**.**

$$\sigma_{mp\_GASTD}(\theta_i) = \sqrt{100/30} \cdot \sigma_{mp}(\theta_i)$$

(4-56)

With $\sigma_{mp}(\theta_i)$ as presented for GAST C in equation (4-52).

- The ground accuracy used for GAST-C should not be impacted since it is supposed to include several error sources and therefore it is assumed to be sufficiently conservative to over bound the errors in the case of GAST-D measurements. However, to take into account advances in GBAS ground station antenna, it is recommended to use the C-curve defined in 4.2.3.2.1 for elevations between 0 and 35 ° all the way to 90 degrees.

4.3 Receiver simulator

4.3.1 Introduction

The first step of the development of the receiver model was to determine which kind of simulator would be implemented, one of the main aspects of our study being the temporal behaviour of the computed position.

Different strategies can be chosen when trying to represent the position error of GNSS solutions:

- Generating the whole GNSS signals and applying the complete operations of reception and processing of these signals. This option is very demanding in computational resources and may take a very long time that's why it is not very convenient for statistical studies which necessitates running several simulations.
- Generating the signals at the correlator outputs level. This solution allows reducing the computational burden while keeping a high degree of representativeness. However, to exploit these advantages it is necessary to know high fidelity models for the different components of the studied signals. Finally, the computational burden may still be important.
- Generating the pseudorange measurements by using pseudorange measurement models. This solution highly decreases the computational burden. However, it is not adapted to represent precisely the temporal behaviour of the GNSS signals even if assumptions can be taken. It is more convenient for statistical studies such as availability simulations.
- Generating directly the position errors. This solution is the fastest in terms of simulation time but it may also be the less precise representation. As we will see further in this document this solution has been chosen in the frame of our GBAS study.

However, as explained previously one of our initial objectives was to study the temporal behaviour of the position error for civil aviation receivers. To do so, it is thus necessary to implement a sufficient degree of representativeness. If our computational resources were unlimited the better solution would of course be the first one which consists in generating the whole GNSS signals. However, this solution

would be too heavy since we wanted to run our simulations on classical working stations. Thus we had two different possibilities left:

- The pseudorange generator
- I/Q correlator outputs generator: it consists as explained above in directly generating the correlator outputs. In fact, in a GNSS receiver, the input signals are correlated with local replicas of the code sequence and carriers of the GNSS signals. It results in the computation of correlator outputs. Models exist which allow to express the correlator outputs depending on the code phase shift and carrier phase shift between the received signals and the locally generated replicas. Moreover, the impact on the correlator outputs of most of the errors which affect the GNSS measurements are well known and have been modelled.

The final choice was the I/Q correlator outputs generator since it has many advantages: easier generation of errors at the correlator level such as noise, interference and multipath, more versatile since it is easier to modify a model or to add a new one. Moreover, we will see that this model can easily be turned into a pseudorange generator when necessary, for service volume simulations for example.

In the following sections we present the different modules which constitute the software receiver.

4.3.2   Simulator architecture

As previously explained, the final chosen model for the simulation of GNSS receiver signal processing behaviour is a correlator outputs generator. However, several other models are necessary to run it. We present here the architecture of the complete simulator. We can divide these modules in three different categories:

- Environmental models: these are the models which are external to the receiver and which determine the pseudorange behaviour
- Receiver signal processing: It gathers the models representing elaboration of the pseudorange measurement which are mainly the tracking loops.
- Position computation and monitoring: It contains the models used to compute a position using the GNSS measurements and the associated integrity monitoring.

It is important to notice that the simulator is capable of performing two different types of simulations:

- IQ simulations which are the most reliable simulations since all the modules are activated.
- Service Volume simulations in which the receiver signal processing part of the simulator is inhibited. In this case, the pseudorange measurements are directly generated using pseudorange measurement error models.
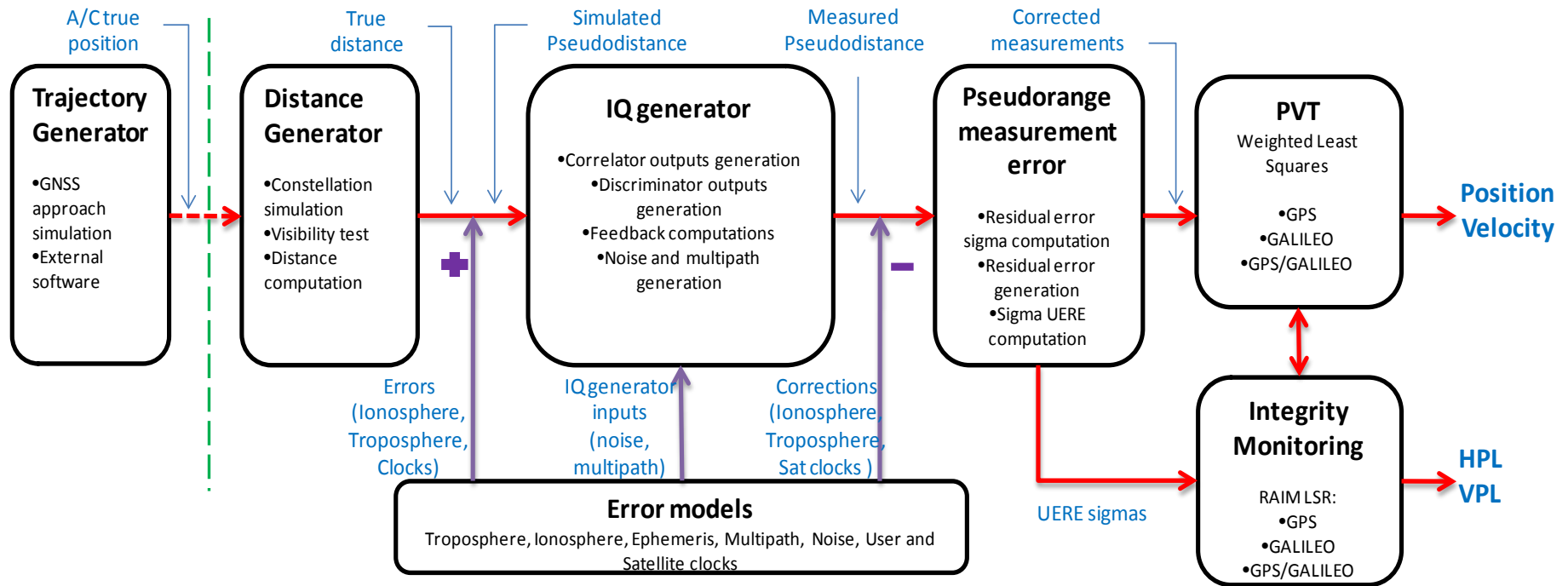
Figure 16: Receiver Simulator architecture

The global simulator architecture is represented in Figure 16. In the case of service volume simulations, the IQ generator and error models modules are inhibited and the true distance sent by the distance generator is directly transmitted without any additional processing to the pseudorange measurement error model.

The chosen architecture may appear to be a little weird at the beginning since the different errors impacting the pseudorange measurements are introduced at different levels of the simulator. In fact, this architecture is adapted to the type of model available for each pseudorange measurement error. In fact, the representativeness of the simulator is driven by the quality of these models since they are used to represent the measurement errors. The difficulty as we have seen already is that some errors are still not precisely modelled. Therefore, we had to deal with that constraint.

To do so our approach was the following. We can distinguish two different types of errors.

Corrected errors: In this category we put the errors for which it is possible to compute a correction at the receiver level. These are ionosphere, troposphere and satellite clock and ephemeris errors. The problem is that in general, it is really difficult to obtain valid models of the true values of these components in the pseudorange before tracking, as well as models for the associated pseudorange corrections. Therefore our strategy was to generate the absolute errors using known models which we knew to be imperfect. The absolute error is injected at the input of our receiver signal processing model. Therefore, the impact of the amplitude and dynamics of these errors on the tracking loops is taken into account.

What is important here is that we can statistically characterize the quality of the pseudorange error models after application of possible pseudorange corrections using sigmas of residual errors. Therefore, at the output of the receiver signal processing model we decided to exactly remove the error injected at the input and to generate a residual error on the pseudorange measurements.

This process is illustrated in the following figure.
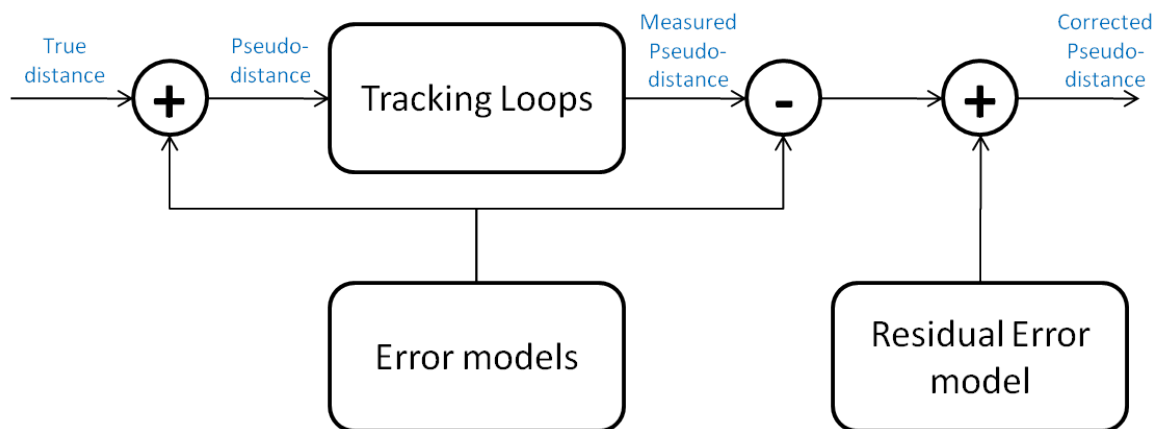
Other Errors: In this category we consider the errors which cannot be corrected using a classical correction model that is to say noise, multipath and interference. In this case, we know the impact of these errors on the correlator outputs. Thus using the appropriate models presented previously we can generate directly the corresponding correlator outputs. For these errors, the residual errors on the pseudorange measurements are linked to the behaviour of the tracking loops and thus their impact is directly taken into account by the IQ generator.

4.3.3 Environmental models

4.3.3.1 Trajectory generator

A trajectory generator has been developed with the help of a student so as to compute the position of the centre of gravity of a civil aviation aircraft along an RNAV/GNSS approach. An example of RNAV/GNSS procedure is shown in Figure 17.
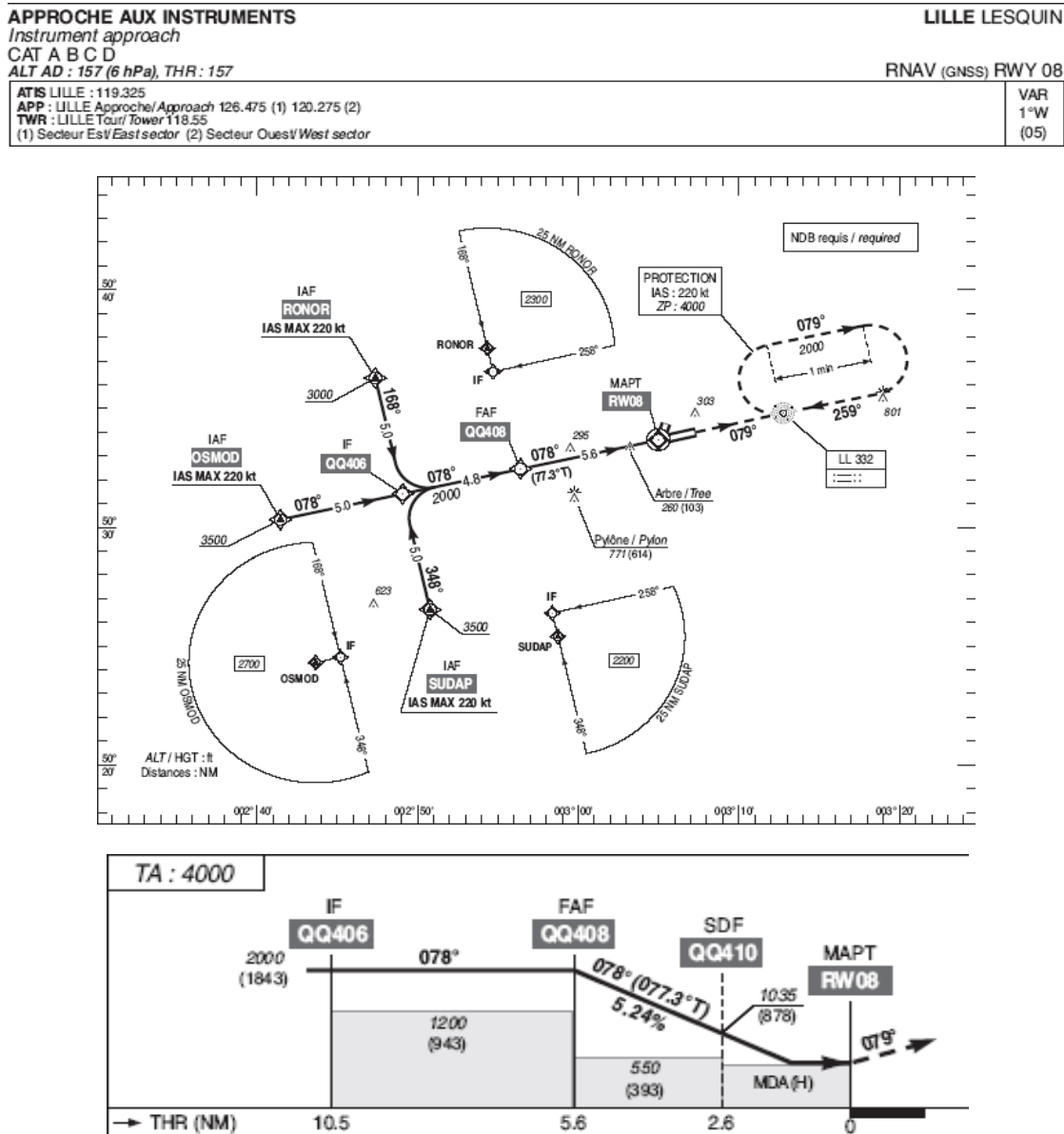
Figure 17: RNAV/GNSS procedure for LILLE Lesquin airport [SIA, 2010]

This type of procedures has standardized shapes which are "Y" and "T" approaches. The generator is currently only able to generate "T" approaches. The trajectory is expressed in the north-east-down coordinate frame centred at the runway threshold. The characteristics are fully customizable so as to allow representing a wide variety of airports. An example trajectory generated with this module is represented in Figure 18 and Figure 19.

The trajectory generated takes into account the velocity profile of the aircraft along the approach. Below can be found an example of generated trajectory:
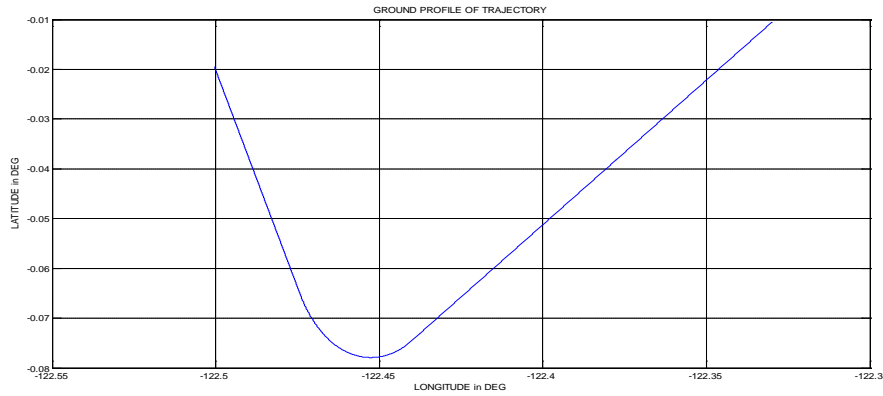
Figure 18: Lateral simulated trajectory profile
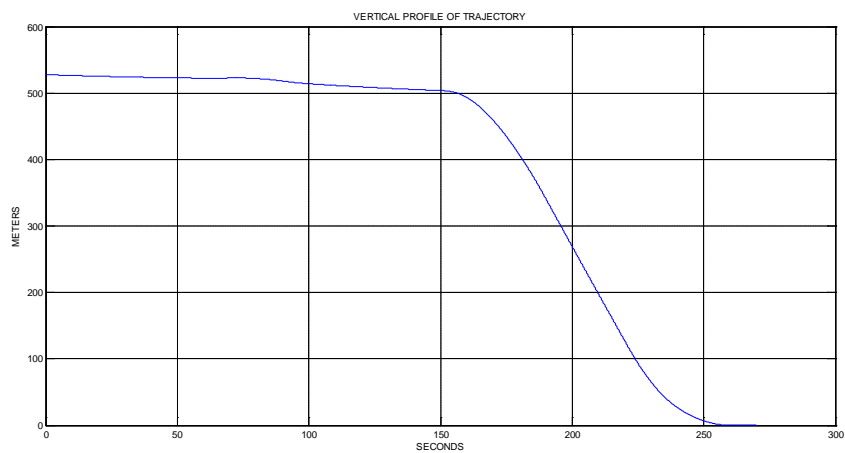
Figure 19: Vertical simulated trajectory profile

Aircraft Behaviour:

The behaviour of the aircraft along its trajectory has been taken into account. It means that we modelled the attitude of the aircraft that is to say pitch, roll and yaw angles. Raised cosine functions were used so as to generate continuous evolutions of these angles. An example is shown in Figure 20.

89
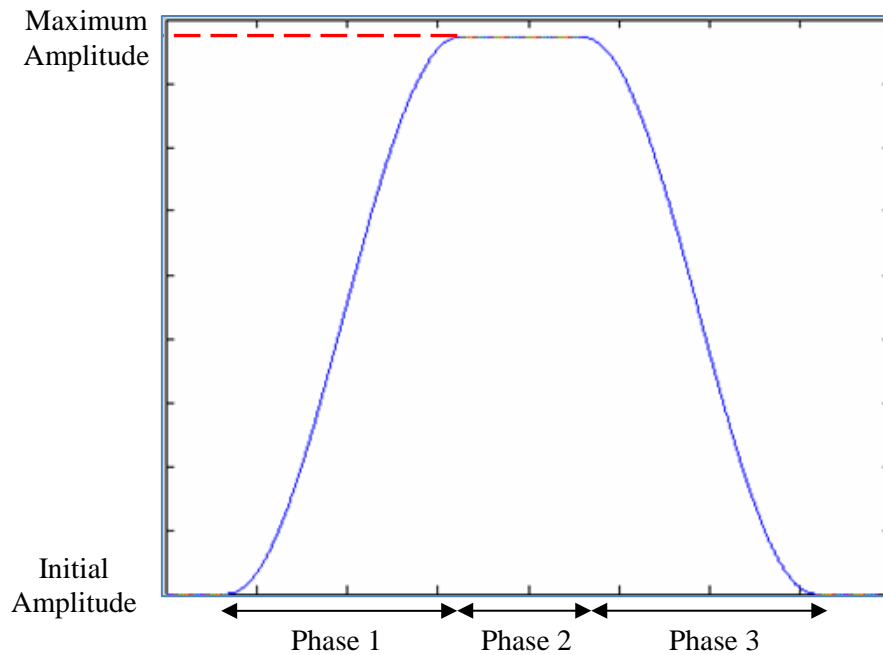
Figure 20: Raised Cosine for attitude angles simulation

Knowing the position of the centre of gravity of the aircraft, the attitude of the aircraft and the relative position of the receiver antenna with respect to the gravity centre, we are able to compute the position of the GNSS receiver antenna along the RNAV GNSS approach. This trajectory is used as the reference position in our simulations.

One additional feature of the trajectory generator is that it is also capable to compute the position of the wings of the aircraft. This information can be used to detect signal outages due to wings during turns for example.

4.3.3.2 Satellite Position

Satellites position and velocities are generated on the basis of GALILEO and GPS almanacs. As presented in [SPILKER, 1996], almanacs provide a reduced precision set of the ephemeris parameters used to compute position of GNSS satellites. For simulation purpose, it is sufficient to reproduce realistic positions and dynamics of the space vehicles that will generate the observation of the most significant geometry configurations. We remind here the algorithm used to compute GPS and GALILEO satellites position using almanacs.

The algorithm used in the case of almanacs is based on the user algorithm for ephemeris determination published in [ARINC, 2004]. Some of the parameters necessary for positioning using ephemeris are considered null in the case of almanacs thus resulting in far simpler equations and therefore an easier implementation.

Different formats of almanacs exist but in this PhD we used YUMA almanacs which have the following structure:

M_0	Mean Anomaly at Reference Time
Δn	Mean Motion Difference From Computed Value
e	Eccentricity
$\sqrt{A}$	Square Root of the Semi-Major Axis
Ω_0	Longitude of Ascending Node of Orbit Plane at Weekly Epoch
i_0	Inclination Angle at Reference Time
ω	Argument of Perigee
$\dot{\Omega}$	Rate of Right Ascension
IDOT	Rate of Inclination Angle
C_{uc}	Amplitude of the Cosine Harmonics Correction Term to the Argument of Latitude
C_{us}	Amplitude of the Sine Harmonics Correction Term to the Argument of Latitude
C_{rc}	Amplitude of the Cosine Harmonics Correction Term to the Orbit Radius
C_{rs}	Amplitude of the Sine Harmonics Correction Term to the Orbit Radius
C_{ic}	Amplitude of the Cosine Harmonics Correction Term to the Angle of Inclination
C_{is}	Amplitude of the Sine Harmonics Correction Term to the Angle of Inclination
t_{oe}	Reference Time Ephemeris
IODE	Issue of Data

Table 14: YUMA almanacs parameters

To compute SVs positions we have to use the ephemeris algorithm with the following additional assumptions:

- Sinusoidal corrections are null i.e. : $\delta u_k = \delta r_k = \delta i_k = 0$
- For the inclination angle, a nominal value of 0.30 semicircles is implicit, only the correction to the inclination δ_1 is transmitted.

We remind here the user algorithm for ephemeris determination which gives the space vehicle antenna phase centre position in WGS-84 Earth-centred, Earth fixed reference frame [ARINC, 2004]:

$\mu = 30986005.10^{14}\ m^3/s^2$	WGS-84 value of the Earth's universal gravitational parameter
$\dot{\Omega}_e = 7.2921151467.10^{-5}\ rad/s$	WGS-84 value of the Earth's rotation rate
$A = \left(\sqrt{A}\right)^2$	Semi major axis
$n_0 = \sqrt{\dfrac{\mu}{A^3}}$	Computed mean motion – rad/s
$t_k = t - t_{oe}$	Time from ephemeris reference epoch
$n = n_0 + \Delta n$	Corrected mean motion
$M_k = M_0 + n.t_k$	Mean anomaly
$\pi = 3.1415926535898$	GPS standard value for π
$M_k = E_k - e.\sin E_k$	Kepler's equation for the eccentric anomaly E_k – rad

91

$$v_k = tan^{-1}\left\{\frac{\sin v_k}{\cos v_k}\right\}=tan^{-1}\left\{\frac{\sqrt{1-e^2}\sin E_k/(1-e.\cos E_k)}{(\cos E_k-e)/(1-e.\cos E_k)}\right\}$$

True anomaly v_k as a function of the eccentric anomaly

$$E_k = cos^{-1}\left\{\frac{e + \cos v_k}{1 + e.\cos v_k}\right\}$$

Eccentric anomaly

$$\Phi_k = v_k + \omega$$

Argument of Latitude

$$\delta u_k = C_{us}\sin 2\Phi_k + C_{uc}\cos 2\Phi_k$$
$$\delta r_k = C_{rs}\sin 2\Phi_k + C_{rc}\cos 2\Phi_k$$
$$\delta i_k = C_{is}\sin 2\Phi_k + C_{ic}\cos 2\Phi_k$$

Argument of latitude correction
Radius correction
Inclination correction

$$u_k = \Phi_k + \delta u_k$$
$$r_k = A(1 - e.\cos E_k) + \delta r_k$$
$$i_k = i_0 + \delta i_k + (IDOT).t_k$$

Corrected argument of latitude
Corrected radius
Corrected inclination

$$x_k' = r_k.\cos u_k$$
$$y_k' = r_k.\sin u_k$$

Satellite position in orbital plane

$$\Omega_k = \Omega_0 + (\dot{\Omega} - \dot{\Omega}_e)t_k - \dot{\Omega}_e t_{oe}$$

Corrected longitude of ascending node

$$x_k = x_k'.\cos\Omega_k - y_k'.\cos i_k.\sin\Omega_k$$
$$y_k = x_k'.\sin\Omega_k + y_k'.\cos i_k.\cos\Omega_k$$
$$z_k = y_k'.\sin i_k$$

Satellite position in Earth-centred, Earth-fixed coordinates

Table 15: Ephemeris equations [ARINC, 2004]

To compute an estimate of the space vehicles velocities we derivate the expressions presented previously. Here are the equations allowing computing the Space Vehicle velocities in the WGS-84 coordinate frame.

$$\dot{M}_k = n$$
$$\dot{E}_k = \frac{\dot{M}_k}{(1 - e.\cos E_k)}$$
$$\dot{v}_k = \sin Ek.\dot{E}_k\frac{1 + e.\cos v_k}{\sin v_k.(1 - e.\cos E_k)}$$
$$\dot{u}_k=\dot{v}_k + 2(C_{us}.\cos 2u_k - C_{uc}.\sin 2u_k).\dot{v}_k$$
$$\dot{r}_k = A.e.n.\frac{\sin Ek}{1 - e.\cos Ek} + 2(C_{rs}.\cos 2u_k - C_{rc}.\sin 2u_k).\dot{v}_k$$
$$\dot{i}_k = IDOT + 2(C_{is}.\cos 2u_k - C_{ic}.\sin 2u_k).\dot{v}_k$$

$$(\dot{x_k'}) = \dot{r}_k\cos u_k - y_k'.\dot{u}_k$$

$$(\dot{y_k}') = \dot{r_k} \sin u_k + x_k^{'} . \dot{u_k}$$

$$\dot{\Omega}_k = (\dot{\Omega} - \dot{\Omega}_e)$$

Satellites velocity in Earth-centred, Earth-fixed coordinates:

$$\dot{x_k} = ((\dot{x_k^{'}}) - y_k' \cos(i_k) \dot{\Omega}_k) . \cos \Omega_k - (x_k^{'} . \dot{\Omega}_k + (\dot{y_k'}) \cos i_k - y_k' \sin(i_k) \dot{i_k}) \sin \Omega_k$$

$$\dot{y_k} = ((\dot{x_k^{'}}) - y_k' \cos(i_k) \dot{\Omega}_k) . \sin \Omega_k + (x_k^{'} . \dot{\Omega}_k + (\dot{y_k'}) \cos(i_k) - y_k' \sin(i_k) \dot{i_k}) \cos \Omega_k$$

$$\dot{z_k} = (\dot{y_k'}) \sin i_k + y_k' \cos(i_k) \dot{i_k}$$

Table 16: Ephemeris Velocity Equations

4.3.3.2.1 Distance computation

The different models gathered in this section are used so as to build a realistic true distance between an aircraft receiver and GNSS satellites. The true geometrical distance can be written as:

$$\rho_i(k) = \sqrt{(X_s^i(k) - X_r(k))^2 + (Y_s^i(k) - Y_r(k))^2 + (Z_s^i(k) - Z_r(k))^2}$$

(4-57)

Where:

- (X_s^i, Y_s^i, Z_s^i) is the position of satellite i
- (X_r, Y_r, Z_r) is the receiver position

We have already presented the two modules which provide the satellites positions and the user position. The next step is to provide the obtained distance to the IQ generator.

4.3.4 Receiver signal processing

This section briefly describes the principles used in the IQ generator module shown in Figure 16.

4.3.4.1 Mono-channel model

As a first step we present the structure of a single channel composed of one DLL and one PLL. The structure used in the simulator for the DLL is presented in Figure 21. The main difference with what happens in the case of a real receiver is that in the simulator we neither simulate the complete GNSS signals at the input of the receiver, nor generate local replicas of the PRN code sequences in the receiver. In fact, the principle is to directly generate the code delays and carrier phases of:

- the incoming GNSS signals
- the local replicas

Thus, we are able to compute the code phase and carrier phase tracking error and we can derive the corresponding correlator outputs using the expressions presented in chapter 3 and reminded here below:

$$
\begin{cases}
I_E(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c\left(\varepsilon_\tau + \dfrac{C_s}{2}\right).sinc(\pi\varepsilon_f T_I)\cos(\varepsilon_\varphi) + n_{I_E}(k) \\[2mm]
I_P(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c(\varepsilon_\tau).sinc(\pi\varepsilon_f T_I).\cos(\varepsilon_\varphi) + n_{I_P}(k) \\[2mm]
I_L(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c\left(\varepsilon_\tau - \dfrac{C_s}{2}\right).sinc(\pi\varepsilon_f T_I).\cos(\varepsilon_\varphi) + n_{I_L}(k) \\[2mm]
Q_E(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c\left(\varepsilon_\tau + \dfrac{C_s}{2}\right).sinc(\pi\varepsilon_f T_I).\sin(\varepsilon_\varphi) + n_{Q_E}(k) \\[2mm]
Q_P(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c(\varepsilon_\tau).sinc(\pi\varepsilon_f T_I).\sin(\varepsilon_\varphi) + n_{Q_P}(k) \\[2mm]
Q_L(k) = \dfrac{A}{2}.d(k).\widetilde{K}_c\left(\varepsilon_\tau - \dfrac{C_s}{2}\right).sinc(\pi\varepsilon_f T_I).\sin(\varepsilon_\varphi) + n_{Q_L}(k)
\end{cases}
$$

<div align="right">

(4-58)

</div>

As we can see the correlator outputs depend on:

- the amplitude of the signal A
- the navigation data bit $d(k)$
- the carrier phase tracking error ε_φ
- the code phase tracking error ε_τ
- the Doppler error ε_f
- the correlation of the locally generated spreading code with the filtered incoming spreading code $\widetilde{K}_c$
- the chip spacing C_s
- and a correlated noise

Therefore we are able to build directly the correlator outputs since these different quantities are well known. We consider here that the amplitude is equal to 2 and that all data bits are equal to 1.
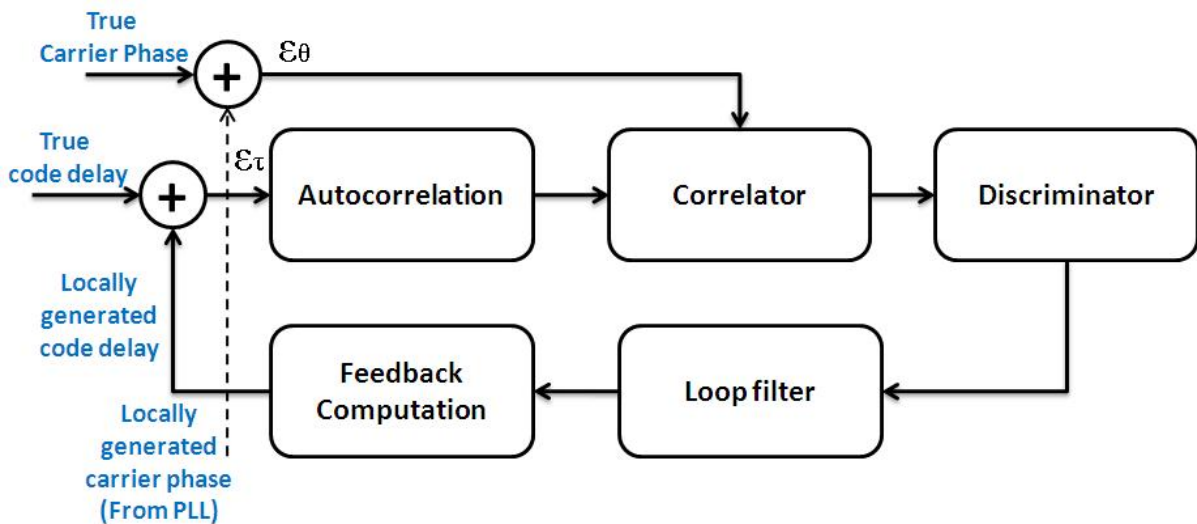
Figure 21: structure of the simulated DLL

4.3.4.1.1 Autocorrelation

In the simulator, depending on the type of signal tracked a sampled autocorrelation function is computed. In the current version of the software it is computed over the interval [-5chips ; +5chips] and it is considered as null out of this interval.

The first step is to compute the ideal autocorrelation function of the considered signal. The corresponding expressions have been presented in chapter 3. Then we have considered the impact of the front-end filter. This filter aims at limiting the impact of disturbances which degrade the useful signal reception such as noise or interference. The drawback is that the useful signal spectrum is filtered as well. This effect must be taken into account and can be represented by impacting the shape of the autocorrelation function accordingly to the front-end filter. If we note $h(t)$ the front-end filter impulse response and $H(f)$ its transfer function we can compute the correlation function of the incoming spreading code sequence with the locally generated code sequence $\widetilde{K}_c(\varepsilon_\tau)$ [JULIEN, 2010]:

$$\widetilde{K}_c(\varepsilon_\tau) = \int_{-\infty}^{+\infty} H(f)\, F_X(f) F_Y(f) e^{-2j\pi f x} df$$

(4-59)

Where:

- $F_X(f)$ is the Fourier transform of the incoming spreading code sequence
- $F_Y(f)$ is the Fourier transform of the locally generated spreading code sequence

The filtered autocorrelation function is sampled and stored. The autocorrelation value corresponding to the actual code delay estimation error ε_τ is then obtained by a linear interpolation of this function. Let's denote the sampled autocorrelation function $\widetilde{K}_{c,s}(i)$ given by:

$$\widetilde{K}_{c,s}(i) = (\varepsilon_{\tau_i})_{i\epsilon[0:N]} \rightarrow \widetilde{K}_c(\varepsilon_{\tau_i})$$

(4-60)

The linear interpolation of the sampled autocorrelation function is defined by

$$\widetilde{K}_{c,l}(\varepsilon_\tau) = \begin{cases} \widetilde{K}_{c,s}(i) + \dfrac{\widetilde{K}_{c,s}(i+1) - \widetilde{K}_{c,s}(i)}{\varepsilon_{\tau_{i+1}} - \varepsilon_{\tau_i}} \cdot (\varepsilon_\tau - \varepsilon_{\tau_{i+1}}), \varepsilon_{\tau_i} < \varepsilon_\tau < \varepsilon_{\tau_{i+1}} \\ \qquad\qquad\qquad 0, \qquad \varepsilon_\tau < \varepsilon_{\tau_0}\ ou\ \varepsilon_\tau > \varepsilon_{\tau_N} \end{cases}$$

(4-61)

We can make the assumption that the sampling step is small enough to consider that the true autocorrelation function can be approximated by the linear interpolation of the sampled autocorrelation function, and thus we obtain:

$$\widetilde{K}_c(\varepsilon_\tau) \approx \widetilde{K}_{c,l}(\varepsilon_\tau)$$

Therefore, we can compute different autocorrelation values, and in particular for Early, Prompt and Late correlation values K_E, K_P, K_L:

$$K_E = \widetilde{K}_c\left(\varepsilon_\tau + \frac{C_s}{2}\right) \cong \widetilde{K}_{c,l}\left(\varepsilon_\tau + \frac{C_s}{2}\right)$$

$$K_P = \widetilde{K}_c(\varepsilon_\tau) \cong \widetilde{K}_{c,l}(\varepsilon_\tau)$$

$$K_L = \widetilde{K}_c\left(\varepsilon_\tau - \frac{C_s}{2}\right) \cong \widetilde{K}_{c,l}\left(\varepsilon_\tau - \frac{C_s}{2}\right)$$

(4-62)

We obtained the autocorrelation values for the early, late and prompt correlators.

We can observe the impact of this filtering for different modulations in the following figures. As we can see, the shape of the autocorrelation function is distorted implying a less accurate synchronization between the received signal and the local replica.
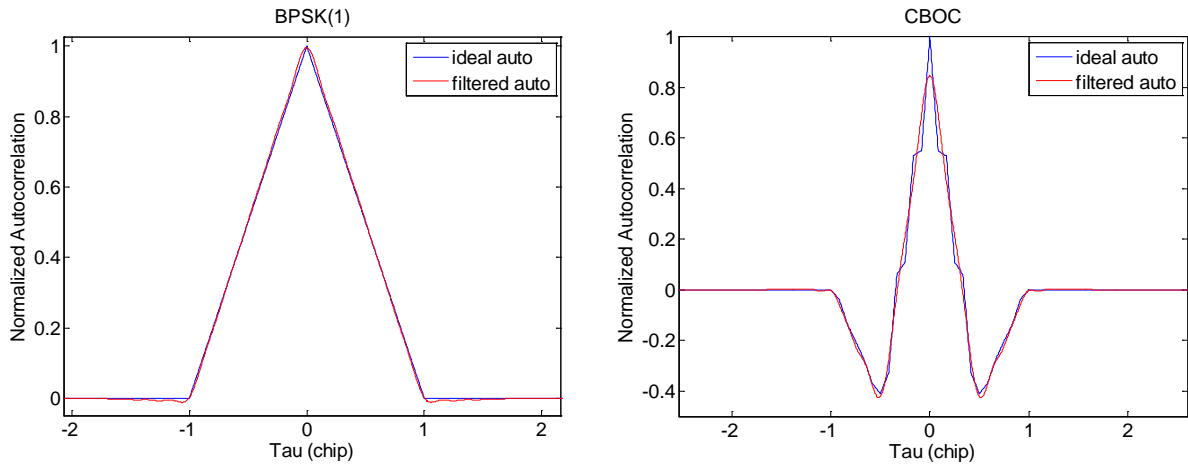
Figure 22 and 23: Impact of front-end filtering on BPSK and CBOC signals autocorrelation function

4.3.4.1.2   Correlators and discriminators

The expressions of the correlator outputs have been described several times. We can write here the final expressions of the modelled correlator outputs in presence of noise and n multipath using equations (4-24) and (4-62):

$$\begin{cases} I_E \cong a_0.\widetilde{K}_{c,l}\left(\varepsilon_\tau + \frac{C_s}{2}\right).\cos(\varepsilon_\theta) + \sum_{i=1}^{n} a^i.\widetilde{K}_{c,l}\left(\varepsilon_\tau + \frac{C_s}{2} + \delta\tau^i\right).\cos(\varepsilon_\theta + \delta\theta^i) + n_{I_E} \\[2mm] I_P \cong a_0.\widetilde{K}_{c,l}(\varepsilon_\tau).\cos(\varepsilon_\theta) + \sum_{i=1}^{n} a^i.\widetilde{K}_{c,l}\left(\varepsilon_\tau + \delta\tau^i\right).\cos(\varepsilon_\theta + \delta\theta^i) + n_{I_P} \\[2mm] I_L \cong a_0.\widetilde{K}_{c,l}\left(\varepsilon_\tau - \frac{C_s}{2}\right).\cos(\varepsilon_\theta) + \sum_{i=1}^{n} a^i.\widetilde{K}_{c,l}\left(\varepsilon_\tau - \frac{C_s}{2} + \delta\tau^i\right).\cos(\varepsilon_\theta + \delta\theta^i) + n_{I_L} \\[2mm] Q_E \cong a_0.\widetilde{K}_{c,l}\left(\varepsilon_\tau + \frac{C_s}{2}\right).\sin(\varepsilon_\theta) + \sum_{i=1}^{n} a^i.\widetilde{K}_{c,l}\left(\varepsilon_\tau + \frac{C_s}{2} + \delta\tau^i\right).\sin(\varepsilon_\theta + \delta\theta^i) + n_{I_E} \\[2mm] Q_P \cong a_0.\widetilde{K}_{c,l}(\varepsilon_\tau).\sin(\varepsilon_\theta) + \sum_{i=1}^{n} a^i.\widetilde{K}_{c,l}\left(\varepsilon_\tau + \delta\tau^i\right).\sin(\varepsilon_\theta + \delta\theta^i) + n_{I_P} \\[2mm] Q_L \cong a_0.\widetilde{K}_{c,l}\left(\varepsilon_\tau - \frac{C_s}{2}\right).\sin(\varepsilon_\theta) + \sum_{i=1}^{n} a^i.\widetilde{K}_{c,l}\left(\varepsilon_\tau - \frac{C_s}{2} + \delta\tau^i\right).\sin(\varepsilon_\theta + \delta\theta^i) + n_{I_L} \end{cases}$$

(4-63)

These correlator outputs are then summed over the integration time and sent to discriminators so as to estimate the code delay tracking error as well as carrier phase tracking error. Some of the discriminators have already been presented in Chapter 3. We remind here all the discriminators available in the simulator.

DLL discriminators:

- Dot-Product (DP): $D_{DP} = (I_E - I_L)I_P + (Q_E - Q_L)Q_P$
- Early Minus Late Power (EMLP): $D_{EMLP} = \left(I_E{}^2 + Q_E{}^2\right) - \left(I_L{}^2 + Q_L{}^2\right)$

PLL discriminators:

- Costas: $D_{Costas} = Q_P \times I_P$
- Atan: $D_{Atan} = atan\left(\frac{Q_P}{I_P}\right)$
- Atan2: $D_{Atan2} = atan2\left(\frac{Q_P}{I_P}\right)$

4.3.4.1.3   Loop filter and feedback computation

The outputs of the discriminators are filtered so as to build a command to the VCOs which control the locally generated replicas. We used the implementation of digital filters presented in [STEPHENS, 1995]. It is not described here.

The command is then used in the feedback computation to build a new estimate of the carrier phase and the code delay.

4.3.4.2   Multi-channel structure

The final architecture of the signal processing model is composed of two main functions:

- The tracking loops
- The supervisor

The tracking loops have been presented in the previous section. In the case of the complete receiver, several couples of tracking loops (DLL+ PLL) are run at the same time, each couple tracking a different signal depending on the configuration chosen.

However, it is necessary to distribute the different tasks to the different tracking loops each task corresponding to the tracking of a particular satellite and signal of this satellite. This is the role of the supervisor. It is in charge of monitoring:

- Rise or set of satellites: so as to activate/deactivate the tracking of these satellites
- Lock status of the tracking loops: to detect loss of locks and engage if necessary a new initialization
- Elevation of satellites: so as to activate/deactivate the introduction of a measurement into the position computation according to the mask angle

This module is not further described since it is simply a software algorithm.

4.3.5   Position computation and monitoring

We describe now the computation of the position solution estimate using the pseudorange measurements coming from the receiver signal processing module. Moreover, as we have seen in

chapter 2, it is necessary for civil aviation users to evaluate the quality of their position solution. We thus present the baseline integrity monitoring solution.

4.3.5.1 PVT

The PVT module is a classical least squares estimation of the position of the aircraft. We remind here the standard single constellation algorithm applicable to both GPS and GALILEO.

We define the state vector X as the user receiver position and receiver clock bias:

$$X = \begin{pmatrix} X_r \\ Y_r \\ Z_r \\ b_u \end{pmatrix}$$

Where:

- (X_r, Y_r, Z_r) are the Cartesian coordinates of the antenna phase center of the user receiver
- b_u is the user receiver clock bias defined by $b_u = c\Delta t_u$
- Δt_u is the user receiver clock error

The observations used to compute the position are the pseudorange measurements made by the receiver for each tracked signal. If we consider that the receiver is tracking at least N different satellite signals to be used to compute the position estimate, then we can obtain N non-linear equations relating the observations to the state vector. For example we have for satellite i:

$$P^i(k) = \sqrt{(X_s^i(k) - X_r(k))^2 + (Y_s^i(k) - Y_r(k))^2 + (Z_s^i(k) - Z_r(k))^2} + c\Delta t_u(k) + e^i$$

(4-64)

Where:

- (X_s^i, Y_s^i, Z_s^i) are the Cartesian coordinates of the antenna phase center of the satellite i
- e^i is the collection of noise, multipath, and iono, tropo, satellite clock and ephemeris residuals after correction. It can be assumed to be a zero mean Gaussian noise of standard deviation $\sigma_{UERE}{}^i$.

The Least Squares position solution is described in [MARTINEAU, 2008] for example. The measurement model can be expressed as:

$$Y(k) = h\big(X(k)\big) + E(k)$$

(4-65)

Where:

- $Y = \begin{bmatrix} P^1(k) \\ \vdots \\ P^N(k) \end{bmatrix}$ the measurement vector

- $E = \begin{bmatrix} e^1(k) \\ \vdots \\ e^N(k) \end{bmatrix}$ the measurement error

Since the measurements do not linearly depend on the true user position, an iterative least squares estimation technique is used. It implies the linearization of the measurement model around successive estimates of the receiver position.

Let's denote $\hat{X}_0(k)$ an initial estimate of $X(k)$. This initial estimate can be determined using past measurements or can be provided by other navigation means.

Assuming that $X(k) = \hat{X}_0(k) + \Delta X(k)$, the measurement model can be expressed as following:

$$Y(k) = h\left(\hat{X}_0(k) + \Delta X(k)\right) + E(k)$$

(4-66)

Thus we can linearize the model around $\hat{X}_0(k)$:

$$Y(k) \cong h\left(\hat{X}_0(k)\right) + \frac{\partial h}{\partial X}\left(\hat{X}_0(k)\right) \times \Delta X(k) + E(k)$$

(4-67)

The first order derivative that appears in this last equation is a $n \times 4$ matrix that can be expressed as:

$$H = \begin{bmatrix} \frac{\partial h^1}{\partial x}\left(\hat{X}_0(k)\right) & \frac{\partial h}{\partial y}\left(\hat{X}_0(k)\right) & \frac{\partial h}{\partial z}\left(\hat{X}_0(k)\right) & \frac{\partial h}{\partial b}\left(\hat{X}_0(k)\right) \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial h^N}{\partial x}\left(\hat{X}_0(k)\right) & \frac{\partial h^N}{\partial y}\left(\hat{X}_0(k)\right) & \frac{\partial h^N}{\partial z}\left(\hat{X}_0(k)\right) & \frac{\partial h^N}{\partial b}\left(\hat{X}_0(k)\right) \end{bmatrix}$$

(4-68)

These derivatives can be computed using equation (4-64) and we obtain:

$$\frac{\partial h^i}{\partial x}\left(\hat{X}_0(k)\right) = \frac{\hat{x}_0 - x^i(k)}{\sqrt{\left(\hat{x}_0 - x^i(k)\right)^2 + \left(\hat{y}_0 - y^i(k)\right)^2 + \left(\hat{z}_0 - z^i(k)\right)^2}}$$

$$\frac{\partial h^i}{\partial y}\left(\hat{X}_0(k)\right) = \frac{\hat{y}_0 - y^i(k)}{\sqrt{\left(\hat{x}_0 - x^i(k)\right)^2 + \left(\hat{y}_0 - y^i(k)\right)^2 + \left(\hat{z}_0 - z^i(k)\right)^2}}$$

$$\frac{\partial h^i}{\partial z}\left(\hat{X}_0(k)\right) = \frac{\hat{z}_0 - z^i(k)}{\sqrt{\left(\hat{x}_0 - x^i(k)\right)^2 + \left(\hat{y}_0 - y^i(k)\right)^2 + \left(\hat{z}_0 - z^i(k)\right)^2}}$$

$$\frac{\partial h^i}{\partial b}\left(\hat{X}_0(k)\right) = 1$$

(4-69)

Where $\hat{X}_0 = (\hat{x}_0 \quad \hat{y}_0 \quad \hat{z}_0)$

Therefore, the linearized model can be written:

$$Y(k) - h\left(\hat{X}_0(k)\right) = H \times \Delta X(k) + E(k)$$

(4-70)

Let's denote:

$$\Delta Y(k) = Y(k) - h\left(\hat{X}_0(k)\right)$$

(4-71)

Then we have:

$$\Delta Y(k) = H \times \Delta X(k) + E(k)$$

(4-72)

$\Delta Y(k)$ represents the deviation between the actual measurements and the predicted noiseless measurements that the receiver would have made if its position and clock delay were $\hat{X}_0(k)$.

The least square estimate of $\Delta X(k)$ is then given by:

$$\Delta \hat{X}(k) = [H^T H]^{-1} H^T \times \Delta Y(k)$$

(4-73)

This solution is the standard Least Square estimation. It can be improved if we have an a priori knowledge of the statistical distribution of the errors affecting the measurements. In this case, the Weighted Least Squares estimation can be used:

$$\Delta \hat{X}(k) = [H^T \Sigma^{-1} H]^{-1} H^T \Sigma^{-1} \times \Delta Y(k)$$

(4-74)

Where Σ is the measurement error covariance matrix $\Sigma = \text{cov}(E)$.

We denote A the projection matrix:

$$A = [H^T \Sigma^{-1} H]^{-1} H^T \Sigma^{-1}$$

(4-75)

Thus,

$$\Delta \hat{X}(k) = A \times \Delta Y(k)$$

(4-76)

$\Delta \hat{X}(k)$ is an estimate of $\Delta X(k)$ which is defined as the deviation between the initial estimate $\hat{X}_0(k)$ and $X(k)$. It is then possible to implement an iterative algorithm starting from an initial estimate $\hat{X}_0(k)$ and improving progressively this estimate through the comparison between the measurements

and the predicted measurements for each estimated position. The iterative algorithm can be implemented to stop if $\Delta\hat{X}(k)$ has a sufficiently small norm.

In our particular case, we have already described in the Pseudorange Measurement model section, the UERE budget for our measurements. This budget is the input to the measurement error covariance matrix allowing the use of the Weighted Least Squares algorithm.

It is also possible to express the positioning error as a function of the measurement error which is interesting for RAIM. If we use equation (4-74):

$$\Delta\hat{X} = [H^T\Sigma^{-1}H]^{-1}H^T\Sigma^{-1} \times \Delta Y$$

We can replace ΔY by its expression which is given in equation (4-72):

$$\Delta\hat{X} = [H^T\Sigma^{-1}H]^{-1}H^T\Sigma^{-1} \times (H \times \Delta X + E)$$

(4-77)

Then,

$$\Delta\hat{X} = \underbrace{[H^T\Sigma^{-1}H]^{-1}H^T\Sigma^{-1}H}_{I_N}\Delta X + [H^T\Sigma^{-1}H]^{-1}H^T\Sigma^{-1}E$$

(4-78)

With I_N the identity matrix.

Thus,

$$\Delta\hat{X} = \Delta X + [H^T\Sigma^{-1}H]^{-1}H^T\Sigma^{-1}E$$

(4-79)

And we obtain,

$$\Delta X - \Delta\hat{X} = -[H^T\Sigma^{-1}H]^{-1}H^T\Sigma^{-1}E$$

(4-80)

Where: $\Delta X - \Delta\hat{X} = (X - \hat{X}_0) - (\hat{X} - \hat{X}_0) = (X - \hat{X})$ is the positioning error.

The measurement residual represents the deviation between the measurements made and the predicted noiseless measurements that the receiver would have made if its position and clock delay were $\hat{X}$ and if there was no noise. It can be expressed such as:

$$\Delta Y = Y - h(\hat{X})$$

(4-81)

$$\Delta Y = h(X) + E - h(\hat{X})$$

(4-82)

$$\Delta Y = h(X_0 + \Delta X) + E - h(X_0 + \Delta\hat{X})$$

(4-83)

101

We can linearize around X_0 as done previously:

$$\Delta Y = H\Delta X - H\Delta\hat{X} + E$$

$$\Delta Y = H(\Delta X - \Delta\hat{X}) + E$$

(4-84)

We can use the result of equation (4-80):

$$\Delta Y = H(-[H^T\Sigma^{-1}H]^{-1}H^T\Sigma^{-1}E) + E$$

(4-85)

Finally,

$$\Delta Y = (I - H[H^T\Sigma^{-1}H]^{-1}H^T\Sigma^{-1})E$$

(4-86)

This final expression gives the relationship between the measurement residual and the measurement error.

4.3.5.2 *Integrity monitoring*

This section focuses on the standard LSR RAIM algorithm for single constellation position solution. Several other integrity means have been studied during this PhD, mainly SBAS, GALILEO GiC and GBAS. SBAS is already included for GPS in the simulator software. GiC and GBAS integrity algorithms have already been coded but are not yet included. However, it was necessary to define priorities and RAIM was identified as a promising solution to provide integrity for combined receivers. That's why RAIM only is described here. The other algorithms can be found in appendices.

AIM stands for Autonomous Integrity Monitoring and refers to situations where a receiver uses the redundancy of satellite measurements to determine whether a fault condition exists that would cause it to have an unacceptable probability to experience a position error outside a specified bound. RAIM is one type of AIM which uses exclusively satellite signals tracked by the receiver to operate. The Least Square Residual (LSR) RAIM is the most classical implementation.

Other standard classes of RAIM algorithms exist such as the Maximum Solution Separation (MSS) method [BROWN, 1988]. Some recently published techniques seems to be promising such as the Generalized Likelihood Ratio (GLR) test method [NIKIFOROV, 2005], the NIORAIM method [HWANG, 2006] or the Multiple Hypothesis RAIM algorithm [BLANCH, 2007]. These methods are not studied here since it has been decided to focus on a standard method and that can be considered as a reference method, using the WLSR RAIM described in [MARTINEAU, 2008] for LPV200. It must however be understood that our purpose here was not to demonstrate that this RAIM algorithm can support stringent phases of flight such as CAT-I but to obtain illustrative protection level values for RAIM used in CAT-I.

We first remind the definition of useful terms related to RAIM integrity monitoring.

4.3.5.2.1 Autonomous fault detection

- **Alert Limits and Protection Levels:**

Alert Limits have already been defined in section 2.4.4.

The **Horizontal Protection Level (HPL)** is the radius of a circle in the horizontal plane (the local plane tangent to the WGS-84 ellipsoid), with its centre being at the true position, that describes the region assured to contain the indicated horizontal position. It is a horizontal region where the missed alert and false alert requirement are met for the chosen set of satellites when autonomous fault detection is used [RTCA, 2006].

The **Vertical Protection Level (VPL)** is half the length of a segment on the vertical axis (perpendicular to the horizontal plane of WGS-84 ellipsoid), with its centre being at the true position, that describes the region assured to contain the indicated vertical position. It defines the vertical region where the missed alert and false alert requirement are met for the chosen set of satellites when autonomous fault detection is used [RTCA, 2006].

Protection Levels are functions of the satellites and user geometry and the expected error characteristics: they are not affected by actual measurements. Their value is predictable given reasonable assumptions regarding the expected error characteristics.

- **Fault Detection, Fault Exclusion**

Fault Detection and Exclusion (FDE) is a receiver processing scheme that autonomously provides integrity monitoring for the position solution, using redundant range measurements. The FDE consists of two distinct parts: fault detection and fault exclusion. The fault detection part detects the presence of an unacceptably large position error for a given mode of flight. Upon the detection, fault exclusion follows and excludes the source of the unacceptably large position error, thereby allowing navigation to return to normal performance without an interruption in service. The fault detection aspects of FDE are referred to as Receiver Autonomous Integrity Monitoring (RAIM). However, FDE also includes the capability to isolate and exclude failed ranging sources so that navigation can continue in the presence of the failure [RTCA, 2006].

Here is a description of the different event associated to FDE algorithms [RTCA, 2006].

Positioning Failure: If the equipment is aware of the navigation mode/alert limit, a positioning failure is defined to occur whenever the difference between the true position and the indicated position exceeds the applicable alert limit. If the equipment is not aware of the navigation/alert limit, a positioning failure is defined to occur whenever the difference between the true position and the indicated position exceeds the applicable protection level (either horizontal or vertical as applicable).

Failed Exclusion (exclusion not possible): A failed exclusion is defined to occur when a true positioning failure is detected and the detection condition is not eliminated within the time-to-alert (from the onset of the positioning failure). A failed exclusion would cause a navigation alert.

Wrong exclusion: A wrong exclusion is defined to occur when a detection occurs, and a positioning failure exists but is undetected after exclusion, resulting in a missed alert.

Missed Alert: Positioning failures that are not annunciated (as an alert) within the time-to-alert are defined to be missed alerts. Both missed detection and wrong exclusion conditions can cause missed alerts after the time-to-alert expires.

False Detection: A false detection is defined as the detection of a positioning failure when a positioning failure has not occurred.

False Alert: A false alert is defined as the indication of a positioning failure when a positioning failure has not occurred (a result of false detection). A false alert would cause a navigation alert.

4.3.5.2.2 LSR detection function

The LSR RAIM method is based on the comparison between a test statistic depending on the prediction error vector and a given threshold. As in the case of the position solution, if we have an a priori knowledge of the statistics of the measurement errors that is to say if we know the measurement error covariance matrix it is possible to use a Weighted LSR algorithm. We only present this solution which can be found in [MARTINEAU, 2008].

The measurement residual ΔY can be expressed as in (4-86):

$$\Delta Y = (I - H[H^T\Sigma^{-1}H]^{-1}H^T\Sigma^{-1})E$$

(4-87)

Where Σ is the measurement error covariance matrix $\Sigma = \mathrm{cov}(E) = \begin{bmatrix} \sigma^2_{UERE,1} & & \\ & \ddots & \\ & & \sigma^2_{UERE,N} \end{bmatrix}$

A scalar test statistics is defined using this observable:

$$WSSE = \Delta Y^T \Sigma^{-1} \Delta Y$$

(4-88)

$WSSE$ corresponds to the sum of the squares of the range residual errors normalized by the standard deviation of the measurement errors.

The weighted LSR RAIM test is defined by [WALTER, 1995]:

$$T = \sqrt{WSSE}$$

(4-89)

The detection threshold is obtained by considering the test statistic $WSSE$ in the fault free case. We assumed in the previous section that E is composed of zero mean Gaussian noise with variance $\sigma^2_{UERE,i}$. Therefore, $WSSE$ is chi-squared distributed with N-4 degrees of freedom:

$$WSSE \sim \chi^2_{N-4}$$

(4-90)

The probability of false alarm is used to determine the normalized detection threshold a such as:

$$P(WSSE > a) = P_{fa}$$

(4-91)

It implies:

$$P_{fa} = \int_a^\infty f_{\chi^2_{N-4}}(x)\, dx$$

(4-92)

We remind the expression of the density probability of the chi-squared distribution with N-4 degrees of freedom:

$$f_{\chi^2_{N-4}}(x) = \frac{e^{-x/2} x^{\frac{(N-4)}{2}-1}}{2^{(N-4)/2}\Gamma\left(\frac{N-4}{2}\right)}$$

(4-93)

Thus, a fault is detected if the chi-squared observable T is abnormally large above the assumed noise level.

In the case of weighted LSR RAIM, the threshold that is compared to our criteria is:

$$h = \sqrt{a}$$

(4-94)

The weighted LSR RAIM test δ is finally given by:

$$\delta = \begin{cases} H_0 \ if \ T \leq h \\ H_1 \ if \ T > h \end{cases}$$

(4-95)

4.3.5.2.3 Protection levels computation

The protection levels derive from the smallest bias the algorithm is able to detect with adequate false alarm rate and missed detection probability.

Let's consider that the measurement error E is noise and a bias b on one satellite j such as:

$$E(k) = \begin{bmatrix} n^1(k) \\ \vdots \\ n^j(k) \\ \vdots \\ n^N(k) \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ b_j \\ \vdots \\ 0 \end{bmatrix} with \ n^i \sim N(0, \sigma_i^2)$$

(4-96)

In this case, $WSSE$ is chi-squared distributed with $N-4$ degrees of freedom and non-centrality parameter λ such as $WSSE \sim \chi^2_{\lambda, N-4}$. This means that $WSSE$ can be written like this:

$$\exists X_i, SSE = X_1{}^2 + \cdots + X_{N-4}{}^2 \ iid, X_i \sim N(\mu_i, 1)$$

(4-97)

$$\lambda = \sum_{i=1}^{N-4} \mu_i{}^2$$

(4-98)

The non centrality parameter λ is computed in order to satisfy the P_{md} requirement such as:

$$P_{md} = \int_0^a f_{\chi^2_{\lambda, N-4}}(x) \, dx$$

<div align="right">(4-99)</div>

The obtained non-centrality parameter λ is the smallest bias that can be detected by the test. Using equation (4-88) $WSSE$ can be expressed as:

$$WSSE = \Delta Y^T \Sigma^{-1} \Delta Y = E^T (I - H[H^T \Sigma^{-1} H]^{-1} H^T \Sigma^{-1})^T \Sigma^{-1} (I - H[H^T \Sigma^{-1} H]^{-1} H^T \Sigma^{-1}) E$$

<div align="right">(4-100)</div>

Let's denote $B = H(H^t \Sigma^{-1} H)^{-1} H^t \Sigma^{-1}$. We can express $WSSE$ as:

$$WSSE = E^t (I - B)^t \Sigma^{-1} (I - B) E$$

<div align="right">(4-101)</div>

The relation between the smallest detectable bias on the pseudorange j and the test statistic $WSSE$ can be simplified as:

$$WSSE = \begin{bmatrix} n^1(k) & \cdots & n^j(k) + b_j & \cdots & n^N(k) \end{bmatrix} (I - B)^t \Sigma^{-1} (I - B) \begin{bmatrix} n^1(k) \\ \vdots \\ n^j(k) + b_j \\ \vdots \\ n^N(k) \end{bmatrix}$$

<div align="right">(4-102)</div>

WSSE is chi-squared distributed with $N - 4$ degrees of freedom and with non-centrality parameter λ:

$$\lambda = \begin{bmatrix} 0 & \cdots & b_j & \cdots & 0 \end{bmatrix} (I - B)^t \Sigma^{-1} (I - B) \begin{bmatrix} 0 \\ \vdots \\ b_j \\ \vdots \\ 0 \end{bmatrix}$$

<div align="right">(4-103)</div>

We denote $M = (I - B)^t \Sigma^{-1} (I - B)$ of size $N \times N$ and $M = [m_{ij}]$ for $i \in [1, N]$ and $j \in [1, N]$.

We can then write:

$$\lambda = b_j m_{jj} b_j = m_{jj} b_j{}^2$$

<div align="right">(4-104)</div>

It is then possible to determine the value of λ which verify $P[WSSE < T_h] = P_{MD}$ in the case of a measurement bias due to a satellite failure.

The smallest detectable measurement bias on satellite j can be then expressed as:

$$b_j = \sqrt{\frac{\lambda}{m_{jj}}}$$

<div align="right">(4-105)</div>

It is possible to relate the position error to the measurement error:

<div align="center">106</div>

$$X(k) - \hat{X}(k) = -A \times E(k)$$

(4-106)

Where:

$$A = [H^T \Sigma^{-1} H]^{-1} H^T \Sigma^{-1}$$

Therefore the impact of the bias b_j in the position domain is obtained by:

$$\Delta X = X(k) - \hat{X}(k) = -A \times \begin{bmatrix} 0 \\ \vdots \\ b_j \\ \vdots \\ 0 \end{bmatrix}$$

(4-107)

We can distinguish the impact of the bias in the horizontal and vertical domain using the North, East, Down (N, E, D) local coordinate frame linked to the user position:

$$\Delta X_H = \sqrt{\Delta X_N^2 + \Delta X_E^2} = \sqrt{A_{N,jj}^2 + A_{E,jj}^2} \times b_j$$

(4-108)

$$\Delta X_V = \left| A_{D,jj} \right| \times b_j$$

(4-109)

It is thus possible to determine a horizontal and vertical protection level for each satellite j as:

$$HPL_j = \sqrt{A_{N,jj}^2 + A_{E,jj}^2} \times b_j$$
$$VPL_j = \left| A_{V,jj} \right| \times b_j$$

The final protection levels are obtained by:

$$HPL = \max(HPL_j)$$
$$VPL = \max(VPL_j)$$

(4-110)

107

4.4 Synthesis

This Chapter aims at defining the different models used to represent the complete process of GNSS position computation, from the emission of the signals by GNSS satellites to the computation of the final position. It thus gathers:

- An analysis to determine which type of GNSS receiver simulator architecture was the most adapted to our study (position level, pseudorange level, correlator outputs simulator or full generation and processing of GNSS signals). The necessity to precisely represent the behaviour of the position error of the combined GPS-GALILEO and RAIM receiver led to the development of an IQ correlator outputs simulator. This type of simulator has the advantage to permit to represent GNSS receivers signal processing with a high degree of representativeness with a computational burden which is important but not excessive.
- The pseudorange errors models: these models are used to generate the errors affecting the pseudorange at the input of the signal processing of a GNSS receiver. These errors are the propagation delays (Ionospheric and Tropospheric), the Clock errors, the Thermal Noise, the multipath and the interferences.
- The RAIM pseudorange measurement error models: it gathers the models of the errors affecting the pseudorange measurements obtained as the outputs of the signal processing of a GNSS receiver including a RAIM monitoring.
- The GBAS pseudorange measurement error models: it gathers the models of the errors affecting the pseudorange measurements obtained as the outputs of the signal processing of a GNSS receiver augmented by GBAS. These pseudorange measurement error models are different from RAIM pseudorange measurement error models since GBAS provides differential corrections to improve the pseudorange measurements.

This chapter also aims at describing the architecture and the most important models of the civil aviation GNSS receiver simulator developed and used to derive the results presented in further chapters. This simulator allows simulating GPS and GALILEO constellations as well as the different signals presented in previous sections. A description of the following elements is thus given:

- The environmental models which consist in the simulation of the GNSS constellations and the aircraft trajectories.
- The receiver signal processing functioning
- The PVT module used to output the final position which consists in a classical Weighted Least Squares Residuals algorithm.
- The integrity module which is used to monitor the position solution. Only the RAIM monitoring has been used for this study.

The next chapter presents the GPS-GALILEO and RAIM combination model and the results obtained in simulation with the GNSS receiver simulator.

Chapter 5: GPS GALILEO combination model

5.1 Introduction

Nowadays, ABAS are commonly used onboard civil aviation aircraft to monitor the quality of the GPS L1 C/A signal, allowing using GNSS as a primary mean of navigation for phase of flight from oceanic down to NPA. However, the performance of the current ABAS position solutions does not allow extending their use to more stringent phases of flight and in particular it cannot guarantee sufficient levels of performance in terms of vertical guidance. However, ABAS solutions are very attractive since they do not need the support of external infrastructures to be operational which allows to benefit from the actual worldwide coverage of the GNSS constellations. Thus, the possibility to develop enhanced ABAS position solutions supporting more demanding phases of flight such as APV and CAT-I would be an important breakthrough for civil aviation users.

The introduction of new GNSS constellations such as GALILEO as well as new GNSS signals such as GPS L5 is an opportunity to develop new GNSS receivers including dual-frequency or even dual constellation capabilities. In this context, Civil Aviation community foresees ABAS as an interesting candidate to provide integrity monitoring to these receivers. It is thus an opportunity to combine the advantages of ABAS with the increased performance raised by new constellations, new satellites and new signals so as to reach more stringent SIS requirements.

However, it appears necessary to address the specific phenomena that will result from the combination of two different constellations in one positioning solution. The aim of this chapter is thus to present an analysis of the temporal behaviour of GPS and GALILEO combined position solution augmented by RAIM. A particular focus is put on the impact of constellation changes on the position error. Moreover a simple algorithm is proposed to limit this impact during the final part of GNSS approaches.

5.2 Combination model

5.2.1 Combined GPS/GALILEO position solution

In this study, the chosen approach was to combine GPS and GALILEO at the pseudorange level. Even if United States and European Union have agreed to provide the GPS to GALILEO Time Offset GGTO through each system's navigation signals, the determination of this offset is thought not to be part of a safety-critical chain [HAHN, 2005]. Thus, combined GPS-GALILEO receivers will have to solve the offset between the local receiver clock and GPS system time as well as the offset between the local receiver clock and GALILEO system time.

For the position solution it results in taking into account an additional bias in the state vector:

$$X = \begin{pmatrix} X_r \\ Y_r \\ Z_r \\ b_{GPS} \\ b_{GAL} \end{pmatrix}$$

<div align="right">(5-1)</div>

Where:

- (X_r, Y_r, Z_r) are the Cartesian coordinates of the antenna phase center of the user receiver
- b_{GPS} is the user receiver GPS clock bias defined by $b_{GPS} = c\Delta t_{GPS}$ and Δt_{GPS} is the time offset between the receiver clock and GPS system time
- b_{GAL} is the user receiver GALILEO clock bias defined by $b_{GAL} = c\Delta t_{GAL}$ and Δt_{GAL} is the time offset between the receiver clock and GALILEO system time

The impact on the observation matrix is the addition of a fifth column to account for the dependency of pseudorange measurements on b_{GAL} which is given by (using the same notations as presented in Chapter 4):

For GPS pseudorange measurement i:

$$\frac{\partial h^i}{\partial b_{GPS}}\left(\hat{X}_0(k)\right) = 1$$

$$\frac{\partial h^i}{\partial b_{GAL}}\left(\hat{X}_0(k)\right) = 0$$

(5-2)

For GALILEO pseudorange measurement j:

$$\frac{\partial h^j}{\partial b_{GPS}}\left(\hat{X}_0(k)\right) = 0$$

$$\frac{\partial h^j}{\partial b_{GAL}}\left(\hat{X}_0(k)\right) = 1$$

(5-3)

The final observation matrix, assuming n GPS pseudorange measurements and m GALILEO pseudorange measurements, can be written as:

$$H = \begin{bmatrix} \frac{\partial h^1}{\partial x}\left(\hat{X}_0(k)\right) & \frac{\partial h^1}{\partial y}\left(\hat{X}_0(k)\right) & \frac{\partial h^1}{\partial z}\left(\hat{X}_0(k)\right) & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial h^n}{\partial x}\left(\hat{X}_0(k)\right) & \frac{\partial h^n}{\partial y}\left(\hat{X}_0(k)\right) & \frac{\partial h^n}{\partial z}\left(\hat{X}_0(k)\right) & 1 & 0 \\ \frac{\partial h^{1+n}}{\partial x}\left(\hat{X}_0(k)\right) & \frac{\partial h^{1+n}}{\partial y}\left(\hat{X}_0(k)\right) & \frac{\partial h^{1+n}}{\partial z}\left(\hat{X}_0(k)\right) & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial h^{m+n}}{\partial x}\left(\hat{X}_0(k)\right) & \frac{\partial h^{m+n}}{\partial y}\left(\hat{X}_0(k)\right) & \frac{\partial h^{m+n}}{\partial z}\left(\hat{X}_0(k)\right) & 0 & 1 \end{bmatrix}$$

(5-4)

Of course, this fifth variable to be estimated is taken into account in RAIM equations mainly through the use of this adapted observation matrix. Moreover, for LSR RAIM, the sum of the squares of the pseudorange residuals are considered as chi-squared distributed with $N - 5$ degrees of freedom, instead of $N - 4$ (N being the total number of pseudorange measurements used for position computation with $N = n + m$).

5.2.2 Constellation freezing algorithm

To implement this algorithm we have assumed that the aircraft knows a sufficiently accurate estimate of its ETA (Estimated Time of Arrival) at the Final Approach Fix (FAF) of the simulated approach and at the touchdown. In fact, we know that most commercial aircraft are equipped with a Flight Management System which provides quite accurate predictions. The accuracy of these predictions

increases when getting closer to the point concerned by the predictions. Thus, this assumption can be considered as realistic. However, further studies are needed to evaluate the level of accuracy of these predictions needed to ensure the efficiency of the proposed algorithm. We also assume that the aircraft is able to estimate the time at which the approach can be considered as terminated. We denote t_{FAF} the ETA of the aircraft at the FAF and t_{TD} the ETA of the end of the approach.

The objective of this simple algorithm is to avoid constellation changes during the final part of the approaches which we have defined as the part of the approach starting at the FAF and terminating when the aircraft has successfully landed.

Since the aircraft can only compute imperfect estimates of t_{FAF} and t_{TD} and to increase robustness with our computation simplification we also defined two margins Δt_{FAF} and Δt_{TD} around this time interval. Thus, in reality, the aim of the algorithm is to avoid constellation changes on the time interval:

$$[t_{FAF} - \Delta t_{FAF}; t_{TD} + \Delta t_{TD}]$$

(5-5)

Additionally, we consider that aircraft is flying final approach when $t > t_{FAF}$.

The core of the algorithm consists in predicting set time of each satellite of the GPS or GALILEO constellation which is tracked by the receiver before t_{FAF}. The computed set times are only estimates which explains why we provisioned margins around the period of interest. The computation of the prediction is detailed hereafter.

It is based on a simple linear extrapolation of the satellites elevation during the end of the approach. We can compute the slope of the linear extrapolation of the satellite elevation as:

$$\alpha = \frac{\theta^i(k) - \theta^i(k-1)}{t(k) - t(k-1)}$$

(5-6)

Where:

- $\theta^i(k)$ is the current elevation angle of satellite i at epoch k.
- $t(k)$ is the current time

We want to know at which time the satellite will set, that is to say we want to know at which time the satellite elevation angle will be equal to the considered mask angle:

$$t_{set}^i = \frac{\theta_{mask} - \theta^i(k)}{\alpha} + t(k)$$

(5-7)

This prediction allows identifying which satellites may lead to a constellation change during the final approach when:

$$t_{set}^i \in [t_{FAF} - \Delta t_{FAF}; t_{TD} + \Delta t_{TD}]$$

(5-8)

Thus, it is possible to remove all these satellites from the position solution before entering the final approach phase so as to ensure that no constellation change will occur between the moment the aircraft is flying the FAF and the moment the aircraft has successfully landed. To do so, we also have to inhibit inclusion of rising satellites in the position solution after t_{FAF}.

In fact, this reasoning is based on the assumption that since we are using a combined GPS-GALILEO receiver, it is likely that even if we remove some of the available satellites from the position solution, there will be far enough visible satellites left to compute an accurate position. It remains interesting to compare the degradation of accuracy due to the reduction of the number of visible satellites induced by our algorithm with the improvement of accuracy due to the removal of position steps that would otherwise be induced by constellation changes.

5.3   Simulation assumptions

5.3.1   Simulated approaches

The horizontal and vertical approach profiles which were used during our simulations are shown in the following figures:
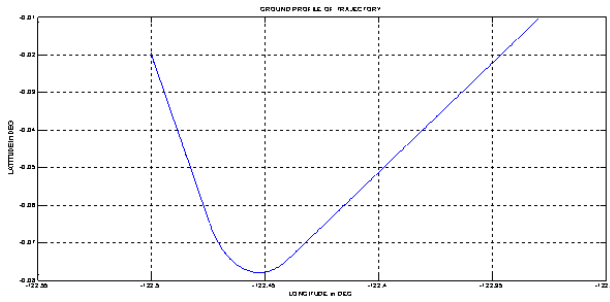
Figure 24: Horizontal profile of simulated approach   **Figure 25: Vertical profile of simulated approach**
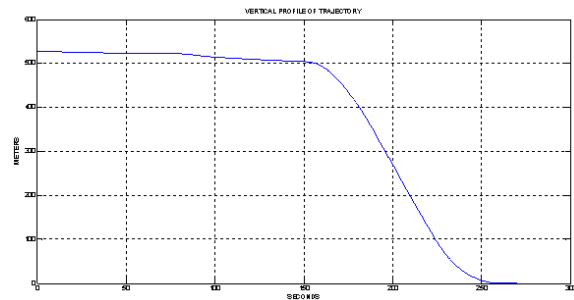
These profiles correspond to the case of a virtual approach at latitude 0° (meaning that it does not correspond to a real aircraft location). As we can see in these figures, the FAF is considered to be sequenced 160s after the initiation, at altitude 493 m (1618 ft) above the ground. Touchdown is considered to occur 270s after the initiation of the approach.

5.3.2   Simulation setup

The main objective of our simulations as explained previously was to statistically assess the characteristics of the NSE of a combined GPS-GALILEO receiver augmented by RAIM. In particular, we were interested in analyzing the impact of possible NSE steps caused by constellation changes.

Thus, to be as exhaustive as possible we tried to take into account all possible constellation configurations so as to be able to compute the probability of occurrence of such events as well as the amplitude of this phenomenon. To do so, we considered the ground track repeat period of both GNSS constellations concerned that is to say GPS and GALILEO.

The complete period of revolution of the ground tracks of the GPS satellites is 23 hours 56 minutes and 4 seconds. For GALILEO satellites the ground tracks repeat period can be assumed to be about three days. Thus, we simulated approaches over around three days and assumed that by doing so we would represent most of possible GPS and GALILEO constellation configurations. Of course, since our simulation capacity is limited we had to sample our simulations and that's why approaches were

simulated every $\Delta t_{app} = 10$ minutes over three days, thus corresponding to $n_{app} = 433$ different approaches, each approach having duration of $T_{app} = 270$ seconds.

This simulation plan was applied to $n_{air} = 16$ different airport locations, which are detailed in Table 17. Thus, the total exposure time T_{exp} is equal to:

$$T_{exp} = n_{app} \times T_{app} \times n_{air} = 1\,870\,560 \text{ s}$$

For each approach, the position solution algorithm provides position estimates of the aircraft at a rate of $f_{pos} = 5\,Hz$, that is we get a position estimate every 0.2 second, or equivalently $T_{app} \times f_{pos} = 1350$ position estimates over the complete 270 s of the approach.

These 433 approaches over the 16 airports were run with the nominal GPS/GALILEO combination algorithm, and with the adapted version of this algorithm presented in a previous section and which reduces the steps due to constellation changes.

Location	Latitude (°)	Longitude (°)	Height (m)
Anchorage	64.174361	-149.996361	46.33
Dallas	32.896828	-97.037996	185.01
New York City	40.639751	-73.778926	3.96
Los Angeles	33.942522	-118.407161	38.10
Miami	25.793250	-80.290556	2.44
Chicago	41.978143	-87.905870	204.83
Seattle	47.4498889	-122.311778	132.00
Johannesburg	26.148175	28.134939	20.00
Dakar	14.738436	-17.488747	20.00
Bruxelles	50.901702	4.483025	20.00
Hong Kong	22.316478	113.936553	20.00
Sydney	-33.933078	151.177550	20.00
Tokyo	35.769655	140.389686	20.00
Beijing	39.960555	116.256944	20.00
Lille	50,558700	3.086170	20.0
Lat 0 airport	0.000000	0.0	20.00

Table 17: Airport location for GPS-GALILEO and RAIM simulations

5.4 Simulation results

We are only interested here on the analysis of the behaviour of the Vertical Position Error (VPE) and behaviour of the Vertical Protection Levels (VPL). The VPE is equal to the geometrical projection of the code-carrier smoothed pseudorange measurement errors induced by thermal noise, multipath, residual ionosphere and troposphere, satellite and receiver clock errors and ephemeris errors. In absence of constellation change, we observed that the VPE is varying very slowly and behaves almost like a constant over the entire approach. This result is not surprising since the behaviour of the VPE is determined by its different error components:

- Rapidly varying errors such as noise, multipath, and interferences are filtered by the code-carrier smoothing filter which has a smoothing constant of 100 seconds. Thus, the impact of these errors on the smoothed pseudorange measurements cannot have time constants lower than 100 seconds.

- The other errors such as tropospheric errors, ionospheric errors and satellite clock errors have very long correlation time and are thus varying very slowly, with time constants much larger than the code-carrier smoothing time constant of 100 seconds.

In the following, we focus on the value of VPE and VPL steps observed when a constellation change occurs. To do that, we compute the difference between VPE and VPL just at the epoch when the constellation change occurs and the VPE and VPL at the previous epoch. This difference is thus dominated by the impact of the constellation change, the difference between two consecutive code-carrier smoothed measurement errors being negligible with $T_e = 0.2\ s$.

We have to remark here that the observed GPS-GALILEO RAIM availability is 100% over the entire simulations, i.e the VPL was never observed over the VAL applicable to CAT-I which is 35.0m (equivalent to former LPV200).

5.4.1   Nominal combination model results
We present in the following sub sections the results obtained for the nominal algorithm. They describe the behaviour of the NSE for a combined GPS-GALILEO receiver as presented previously in the case of civil aviation GNSS approaches.

5.4.1.1   Illustrations of temporal behaviours
The purpose of this section is to give examples of the impact of constellation changes on the horizontal position error (HPE) and vertical position error (VPE) as well as on the associated RAIM protection levels.

This sub section shows some illustrations of the impact of the constellation freezing algorithm compared to the reference situation when this algorithm is not used, together with some interpretations of these illustrations. The situations illustrated cannot be used to derive general performance results but help to understand the impact of the constellation freezing algorithm in comparison with the initial reference situation. Set1 of illustrations presents a situation where a satellite is lost before the FAF and one satellite after the FAF. Set2 of illustrations presents a situation where a satellite is included in the position solution after the FAF.

5.4.1.1.1   Set1 - first approach run among the 433, Beijing airport
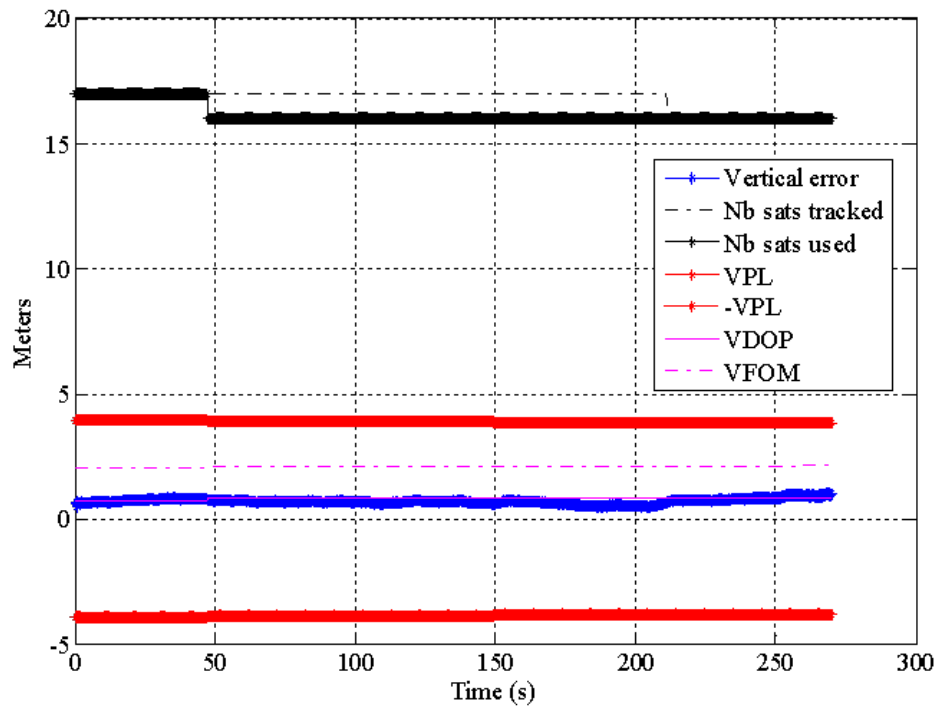
Figure 26: Set 1 - impact of constellation changes on vertical position error (VPE) in meters, VPL, VDOP and VFOM for nominal configuration

Several results are given in Figure 26:

- The Vertical Position Estimation error (VPE)
- The number of GPS/GALILEO satellites tracked by the receiver as well as the number of satellites effectively used in the position solution.
- The Vertical Protection Level (VPL)
- The Vertical Dilution of Precision (VDOP)
- The Vertical Figure Of Merit (VFOM)

We remind here that we have decided to distinguish the beginning of the approach and the final approach which are separated by the Final Approach Fix (FAF).

By observing the number of satellites used for position computation, we can see in Figure 26 that one satellite is lost before the a/c reaches the FAF. However, we can see that this satellite is still tracked by the receiver, thus this particular satellite must have crossed the relevant mask angle. It is no more considered as a reliable satellite.
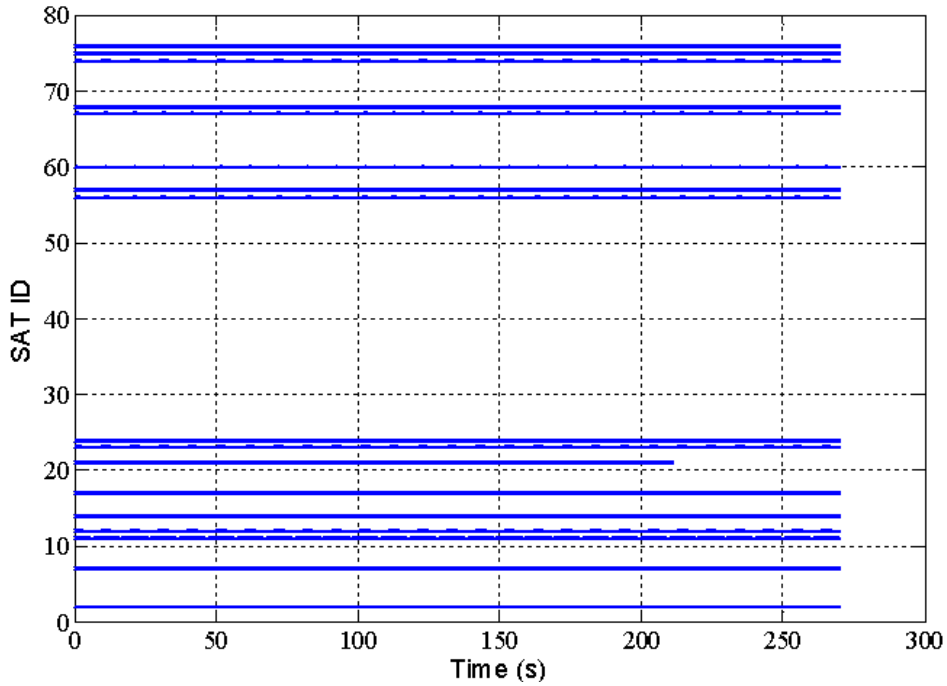
Figure 27: Set 1 - Ids of the tracked GPS and GALILEO satellites as a function of time
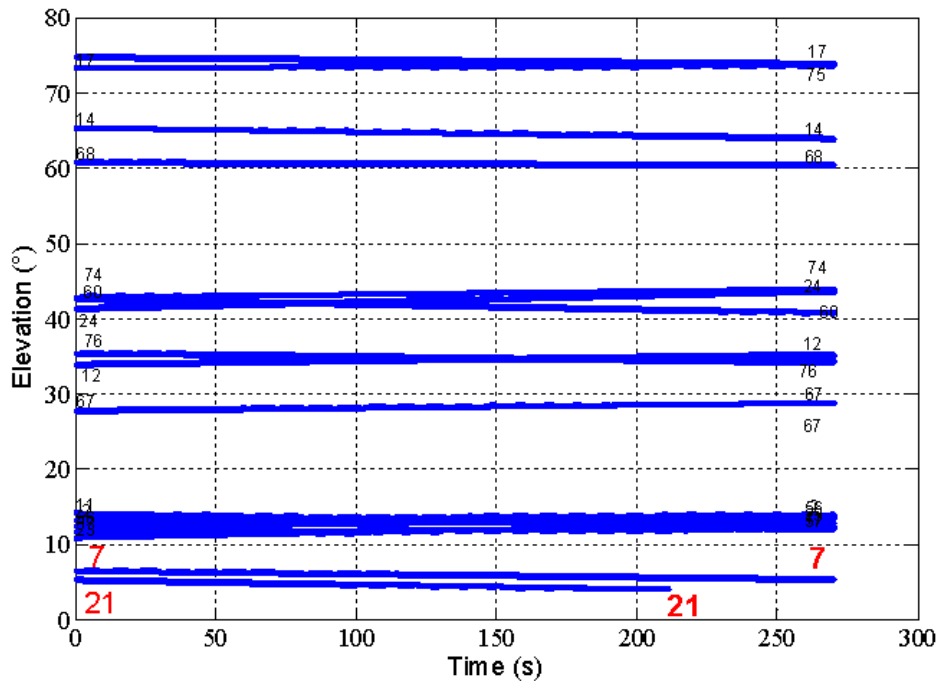
Figure 28: Set 1 - Elevation angle of GPS and GALILEO satellites as a function of time

The effect of this loss is easily observable, since it results in a sudden step increase of the HPE. The associated HPL vary accordingly and thus properly reflect the HPE behaviour. Another interesting parameter is the HDOP which increases too. In fact the DOP allows quantifying the quality of a constellation's geometry. In this particular case, the satellite removed from the position solution, which is GPS satellite number 21 according to Figure 26 and Figure 27, has a very low elevation close to the GPS mask angle as it can be seen in Figure 28. Thus, it had a favourable impact on the HDOP since it represented an important contribution to the horizontal repartition of the satellites.
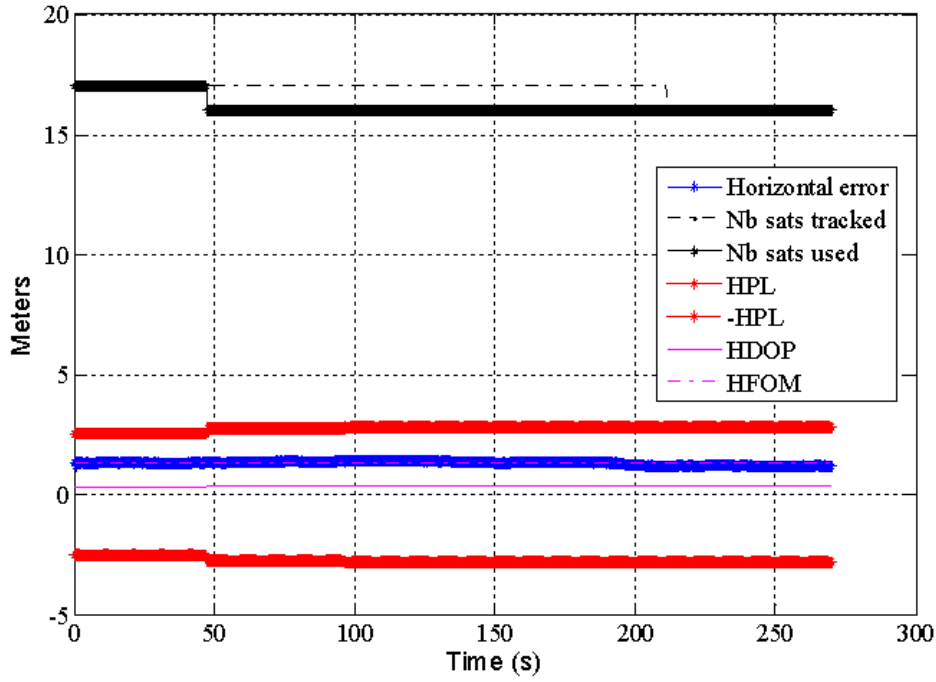
Figure 29: Set 1 - impact of constellation change on Horizontal Position Error (HPE) in meters, HPL, HDOP and HFOM for nominal configuration

However, we have to admit that the steps observed on the horizontal position are relatively small in this simulation since they are below $0,10\ m$.

An equivalent behaviour can be observed on the HPE in Figure 29.

5.4.1.1.2   Set 2 - First approach run among the 433, Tokyo airport

The simulation results obtained with the third set of data is different from the previous one since it introduces a different event. In this case, a satellite is rising instead of setting. Thus, the satellite is included in the position solution during the approach.

120
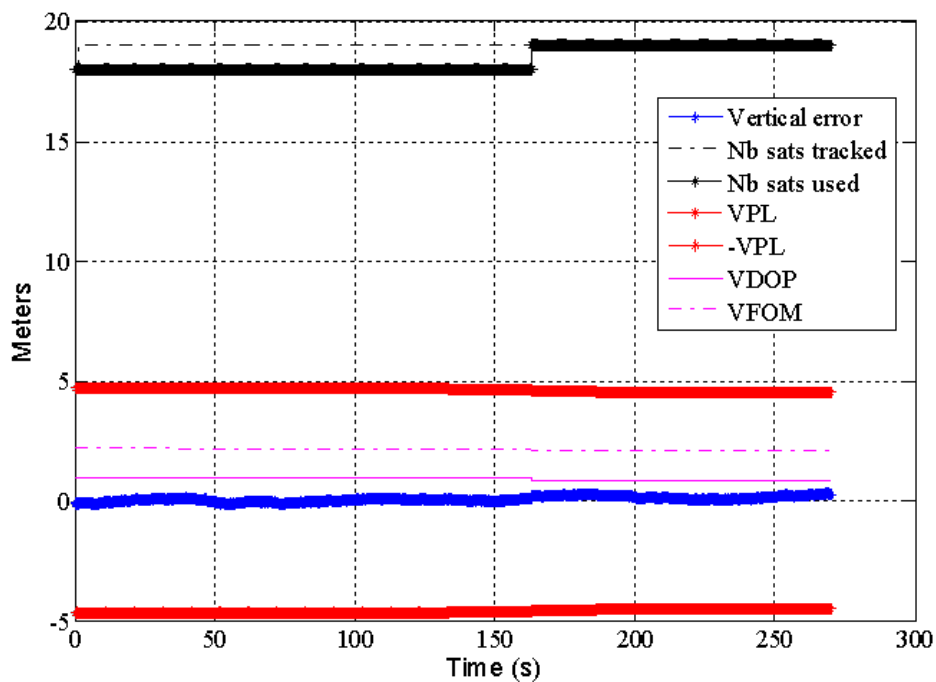
Figure 30: Set 2 - impact of constellation changes on Vertical Position Error (VPE) in meters, VPL, VDOP and VFOM for nominal configuration

In fact, we can observe in Figure 32 that GPS satellite number 23 is passing above the GPS mask angle after t_{FAF}. It results in the inclusion of this satellite in the position solution which is illustrated in Figure 30 by the number of satellites used represented by a continuous black line.

It is very interesting in this case to observe the behaviour of the position estimation error and DOP as well as the behaviour of the protection levels which are related to the RAIM integrity monitoring of the pseudorange measurements.
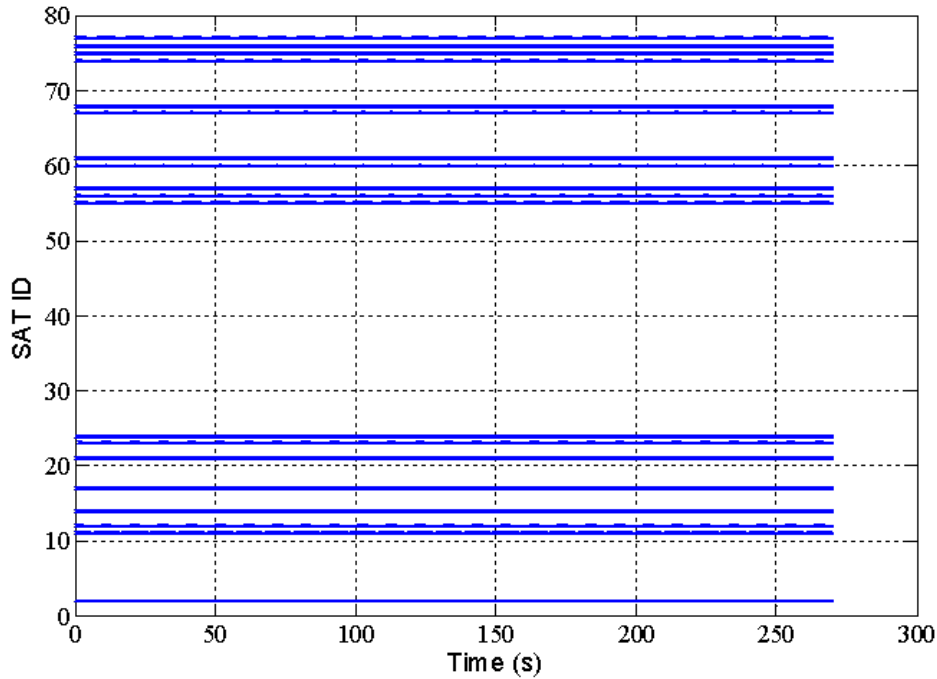
Figure 31: Set 2 - Ids of the tracked GPS and GALILEO satellites as a function of time
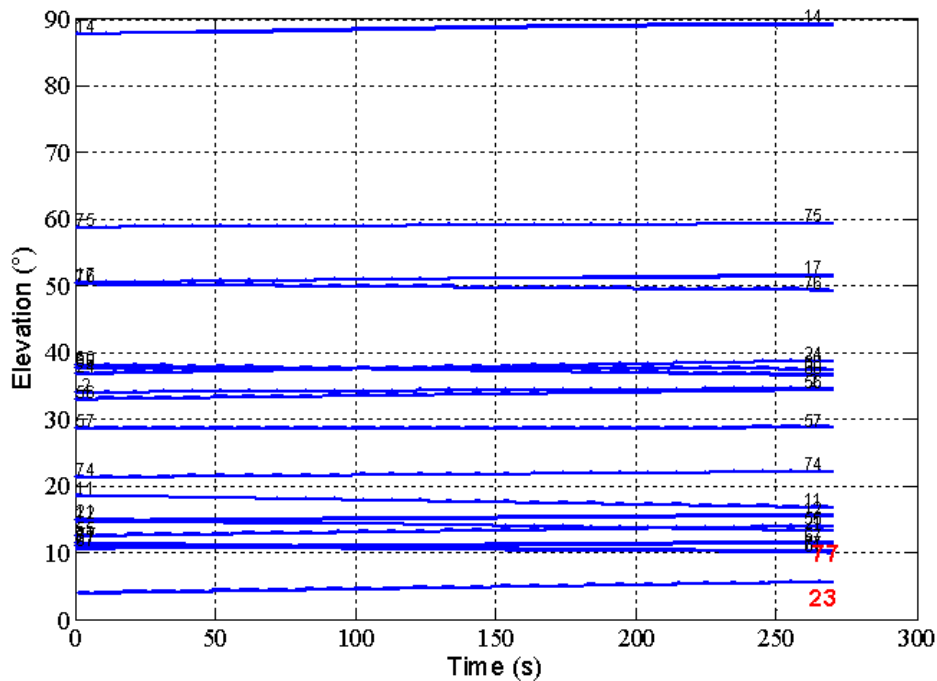
Figure 32: Set 2 - Elevation angle of GPS and GALILEO satellites as a function of time

In fact, we can see in Figure 30 that the inclusion of a healthy satellite leads to a decrease of the VPL as well as a slight decrease of the VDOP. The same effect can be observed in the horizontal plane in Figure 33: the inclusion of GPS satellite 23 leads to a decreasing step in the HPE as well as in the HDOP. The HPL does not seem to be really impacted. However, HPL were already very low since during all the approach the receiver was tracking a minimum of 19 satellites. To conclude, we can see that in this simulation the impact of the inclusion of a new satellite can be considered as negligible due to the high number of satellites already available thanks to the tracking of two different GNSS constellations.
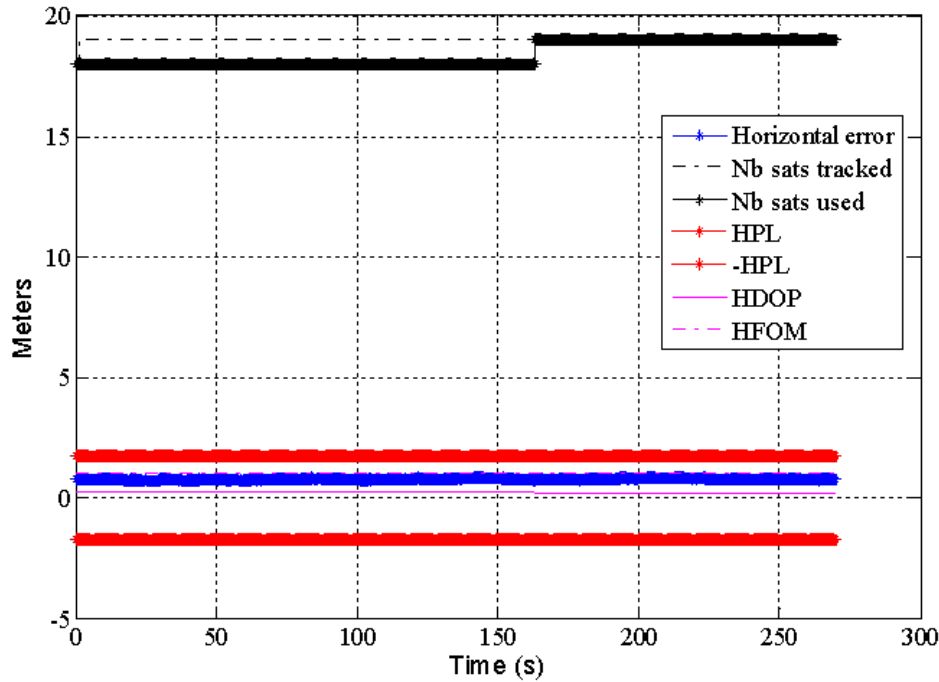
Figure 33: Set 2 - impact of constellation change on Horizontal Position Error (HPE) in meters, HPL, HDOP and HFOM for nominal configuration

5.4.1.2   *Statistical results*

Our approach for the statistical analysis of the simulations was to first evaluate the impact of constellation changes on the position estimation error during the whole approach and then to distinguish the constellation changes occurring during the beginning of the approach (before the FAF) and the ones occurring during the final approach (after the FAF).

5.4.1.2.1   **Results over the complete approach**

We have represented in Figure 34 the histogram of the estimated average value of the vertical position estimation error. Each average value used to build this histogram was derived using the samples of one simulation run, it was thus obtained by averaging $T_{app} \times f_{pos} = 1350$ samples. Thus, the histogram is finally composed of $n_{app} \times n_{air} = 6928$ mean values. As we can see, the obtained distribution has a mono-modal Gaussian shape. The minimum and maximum values of the estimated average vertical position errors over the approach are respectively -4.39 and $4.19\ meters$.
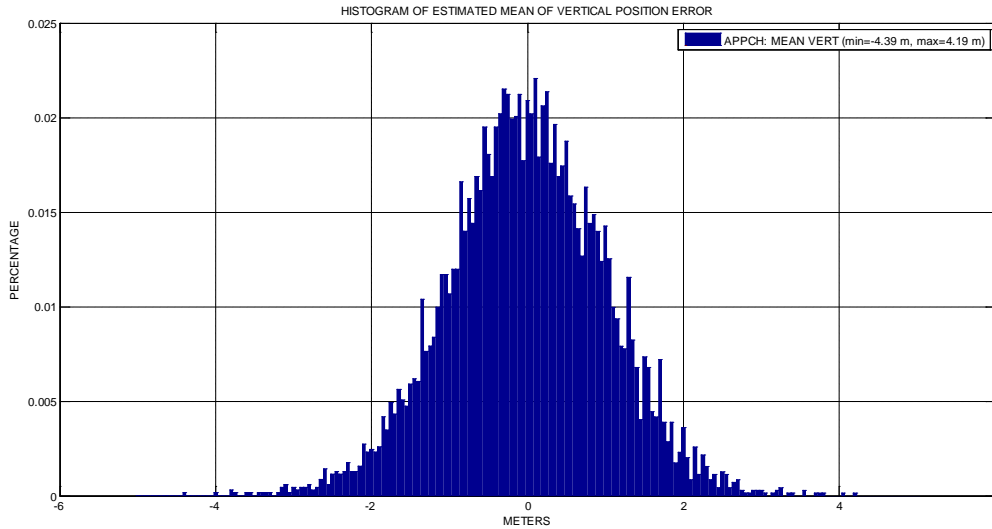
Figure 34: Histogram of the estimated mean of VPE over all simulation runs for the complete approaches in nominal configuration

On Figure 35, we can observe the histogram of the estimated standard deviation of the Vertical Position Error derived the same way as the estimated mean. We can see that the minimum and maximum values are respectively 0.02 and 1.21 *meters* which is very small with respect to classical GNSS receivers (ie. GPS L1 C/A receivers). In fact, we can see that the probability that the standard deviation of the VPE is higher than $0.4\ m$ is smaller than 5%.
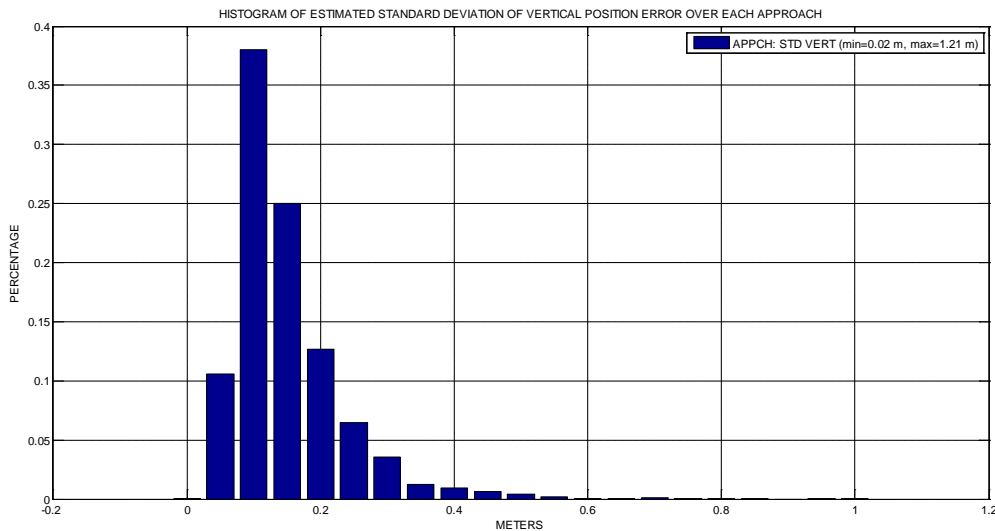
Figure 35: Histogram of estimated standard deviation of VPE over all simulation runs for the complete approaches in nominal configuration

The next figure presents the impact on the Vertical Position Error of the constellation changes. In particular, it represents the distribution of VPE steps induced by constellation changes over all simulation runs. It gathers the steps magnitude of the whole 2894 constellation changes observed during our simulations. We can see that the minimum and maximum steps that have been observed are respectively of -1.96 and 1.8 *meters*. These steps are quite big and it may be interesting for civil aviation users to avoid them as much as possible during the final part of approaches with vertical guidance. In fact, in the case of autoland these steps could lead to undesired reactions of the autopilot.
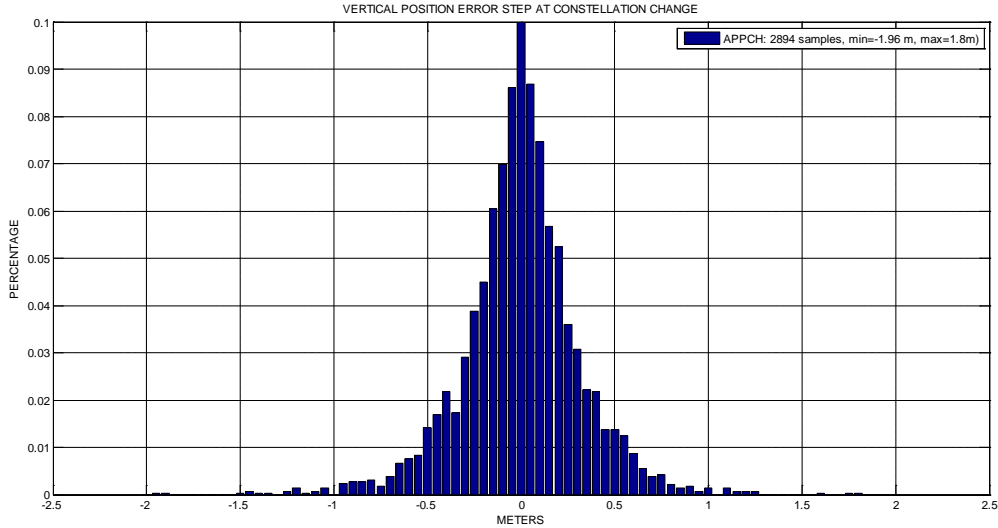
Figure 36: Histogram of VPE steps at constellation change over all simulation runs for the complete approaches in nominal configuration

The impact of constellation changes on the RAIM computed vertical protection levels has been represented in Figure 37. We can see that in some simulations, the steps on vertical protection levels have been higher than $3\ m$ and have even reached steps of 4 meters. During approaches with vertical guidance, we can imagine that these steps could lead to the vertical protection level exceeding the relevant vertical alert limit. It would thus result into the unavailability of the GPS+GALILEO and RAIM position solution and may lead to the abortion of the approach (Go Around). We can however notice, that during our simulations, VPL was never observed to exceed the VAL of 35.0 meters.
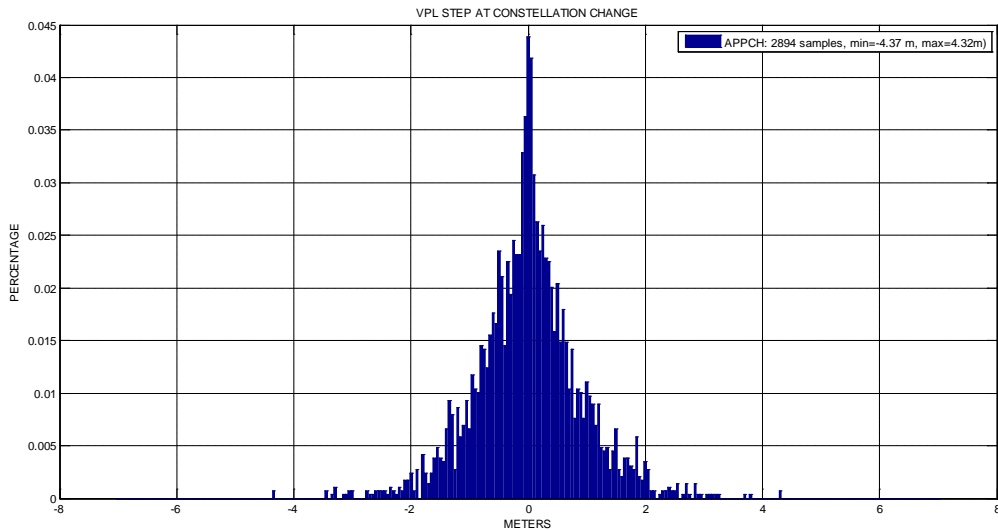
Figure 37: Histogram of VPL steps at constellation change over all simulation runs for the complete approaches in nominal configuration

5.4.1.2.2 Results before and after the FAF
In this section we tried to distinguish the behaviour of the vertical position error during the two different phases of the approach already identified previously:

- Before the FAF: as explained, the FAF is reached $T_{FAF} = 160\ s$ after the beginning of the simulation run.

- Final approach: this part takes place from the FAF to the end of the approach at $t = 270\ s$.

The aim here is to highlight the quantity of constellation changes occurring during the final approach since these are the most critical for autoland functions due to the proximity to the ground.
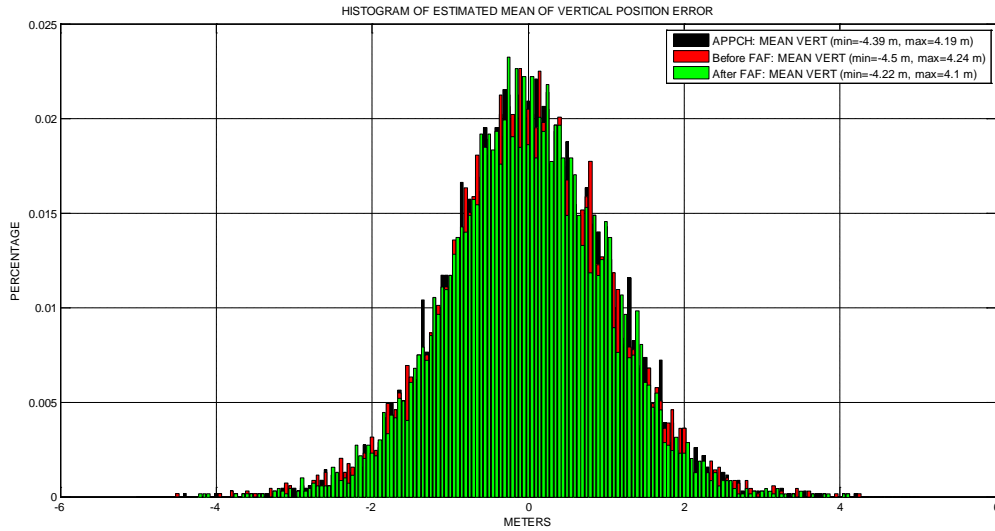
Figure 38: Comparison of the histograms of the estimated mean of VPE over all simulation runs before and after the FAF in nominal configuration

The same figures as previously have been represented. Figure 38 presents the two histograms obtained by collecting the estimated mean of the Vertical Position Error during all simulations and by distinguishing the samples before and after the FAF. The previous histogram over the complete approaches is reminded as a reference. Since FAF is passed by the aircraft at $t_{FAF} = 160\ s$ to compute each value of the estimated VPE we used $T_{FAF} \times f_{pos} = 160 \times 5 = 800$ samples. For the histogram corresponding to the events occurring after the FAF we used for each estimate for the mean VPE $(T_{APP} - T_{FAF}) \times f_{pos} = 110 \times 5 = 550$ samples. As previously we thus obtained $n_{app} \times n_{air} = 433 \times 16 = 6928$ estimated mean VPE to build each histogram.

	Minimum estimated mean VPE	Maximum estimated mean VPE
Complete approach	-4.39	4.19
Before FAF	-4.5	4.24
After FAF	-4.22	4.1

Table 18: Minimum and maximum estimated mean VPE observed in nominal configuration

This figure shows that the different histograms obtained have Gaussian shapes. The maximum and minimum mean values observed which are gathered in the previous table are nearly the same which is quite logical since the position of the a/c in the approach has a limited impact on the mean VPE. We can however see here that the maximum estimated mean values of the VPE are observed before the FAF with $min_{VPE} = -4.5\ m$ and $max_{VPE} = 4.24\ m$.

125
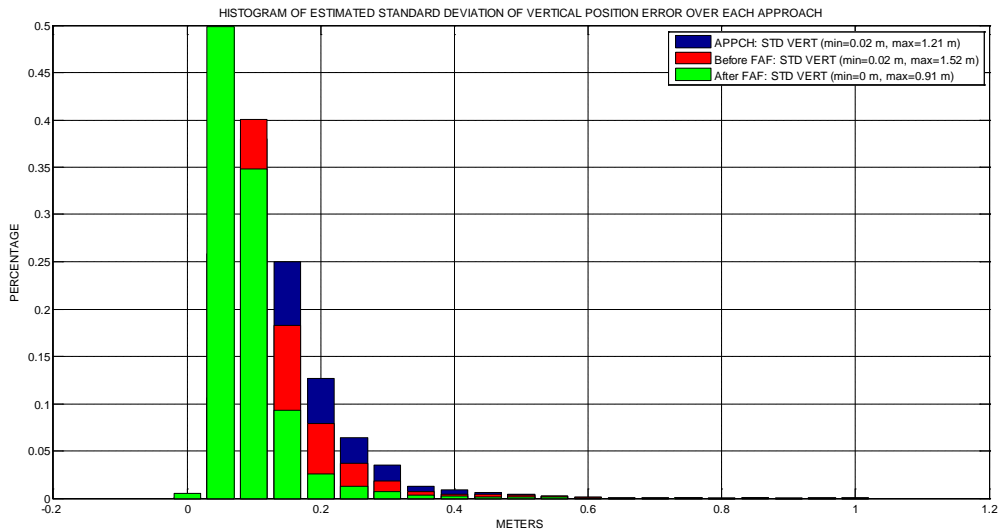
Figure 39: Comparison of the histograms of estimated standard deviation of VPE over all simulation runs before and after the FAF in nominal configuration

On the contrary, the results are slightly different when we take a look at the estimated standard deviation of the VPE represented in Figure 39. The minimum and maximum estimated standard deviations of VPE observed are gathered in the following table:

	Minimum estimated std VPE	Maximum estimated std VPE
Complete approach	0.02	1.21
Before FAF	0.02	1.52
After FAF	0.0	0.91

Table 19: Minimum and maximum estimated VPE standard deviation observed in nominal configuration

In fact, we have already seen that if we make no distinction between the first part of the approach and the final approach, the maximum observed standard deviation of VPE was about $1.21\ m$ and we also observed that the standard deviation was lower than $0.4\ m$ more than 95 % of the time. However, if we distinguish the standard deviation before and after the FAF, we can see that the obtained standard deviation histograms are different. On the one hand, before the FAF the observed standard deviation has a higher maximum of about $1.52\ m$. On the other hand, after the FAF the observed standard deviation has a maximum value of about $0.91\ m$ and the corresponding histogram is more concentrated in the lower VPE standard deviation magnitude. So, we can see that globally, the VPE standard deviation is lower during the Final approach than before the FAF.
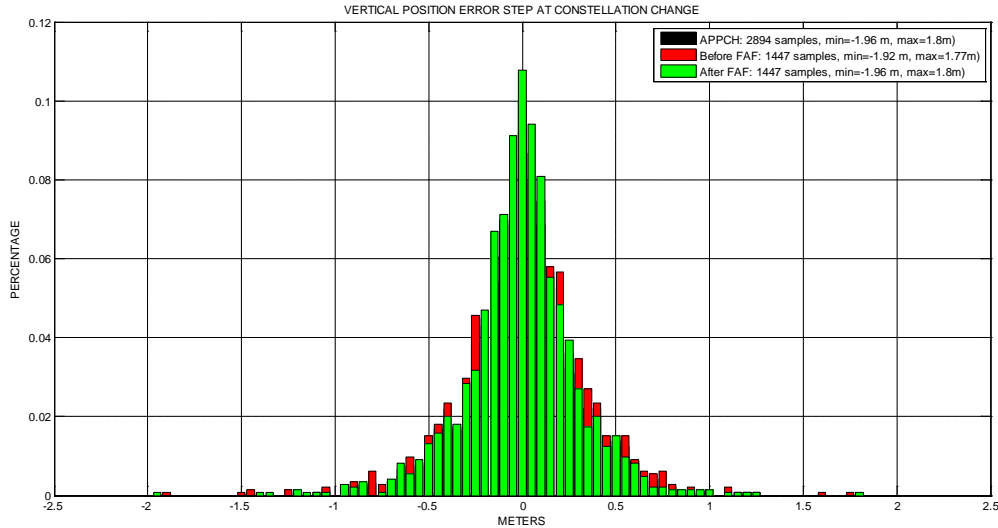
Figure 40: Comparison of the histograms of VPE steps at constellation change over all simulation runs before and after the FAF in nominal configuration

We then look at the distribution and the magnitude of the VPE steps due to a constellation change during the approach thanks to Figure 40. A total number of $n_{steps} = 2894$ constellation changes have been detected during our simulations resulting in VPE steps of minimum and maximum amplitude which are gathered in the following table:

	Minimum VPE steps	Maximum VPE steps
Complete approach	-1.96	1.8
Before FAF	-1.92	1.77
After FAF	-1.96	1.8

Table 20: Minimum and maximum estimated VPE steps observed in nominal configuration

The most interesting result is that we have observed the same number of VPE steps caused by constellation changes before and after the FAF. In fact, we observed 1447 steps before the FAF and 1447 steps after the FAF. IF we express the probability of occurrence of a step per approach we thus obtain:

$$P_{steps\_before\_FAF} = P_{steps\_after\_FAF} = \frac{1447}{n_{app} \times n_{air}} \approx 0.21$$

As we can see, the probability that a constellation change occurs during the approach is quite high.

In terms of magnitude, we can see that the distributions of the VPE steps magnitude have Gaussian shapes with the minimum and maximum observed values being around $-1.9\,m$ and $1.8\,m$ equivalently before and after the FAF.
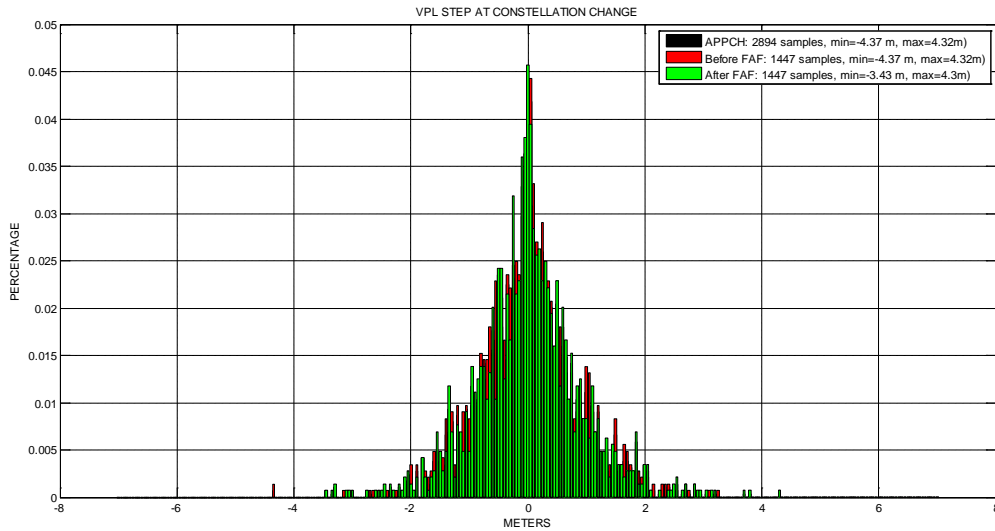
Figure 41: Histogram of VPL steps at constellation change over all simulation runs before and after the FAF in nominal configuration

Figure 41 represents the steps induced by the recorded constellation changes on the RAIM vertical protection levels. The different minimum and maximum values of each histogram are gathered in the following table:

	Minimum VPL steps	Maximum VPL steps
Complete approach	-4.37	4.32
Before FAF	-4.37	4.32
After FAF	-3.43	4.3

Table 21: Minimum and maximum estimated VPL steps observed in nominal configuration

We can see in this distribution that the magnitudes of the RAIM vertical protection levels are nearly the same before and after the FAF. We observe steps having an absolute magnitude higher than $4.0\ m$. The algorithm proposed previously is intended to prevent such jumps in the horizontal and vertical position error as well as in the associated RAIM protection levels.

5.4.2 Constellation freezing algorithm results

This algorithm has been described at the beginning of this chapter. The objective of this section is to illustrate the behaviour of this algorithm and the associated impact on the position error. First we describe the temporal behaviour of the receiver when using the constellation freezing algorithm. Two different behaviours are described.

5.4.2.1 *Illustrations of time behaviours*

5.4.2.1.1 **Set1 - first approach run among the 433, Beijing airport: anticipation of the loss of a satellite**
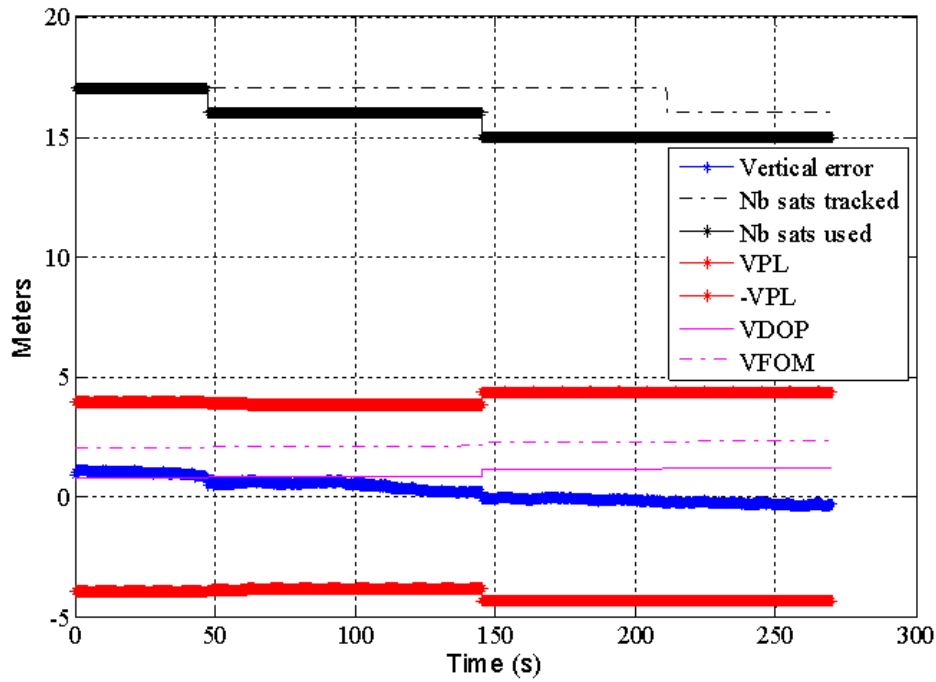
Figure 42: Set 1 – impact of constellation changes on vertical position error (VPE) in meters, VPL, VDOP and VFOM with constellation freezing algorithm

Several data are represented in Figure 42. What is interesting is to look at the number of tracked satellites $n_{tracked}$ (satellites which are tracked by tracking loops) and the number of used satellites n_{used} (satellites which are effectively used to compute the final position estimate). In fact, we have seen previously that in our baseline receiver simulator these data are different since we start tracking satellites when they are available even if their elevation is not higher than the applicable mask angle. Thus, a satellite which is visible from the receiver but which elevation is not above the mask angle is tracked but not used in the position solution.
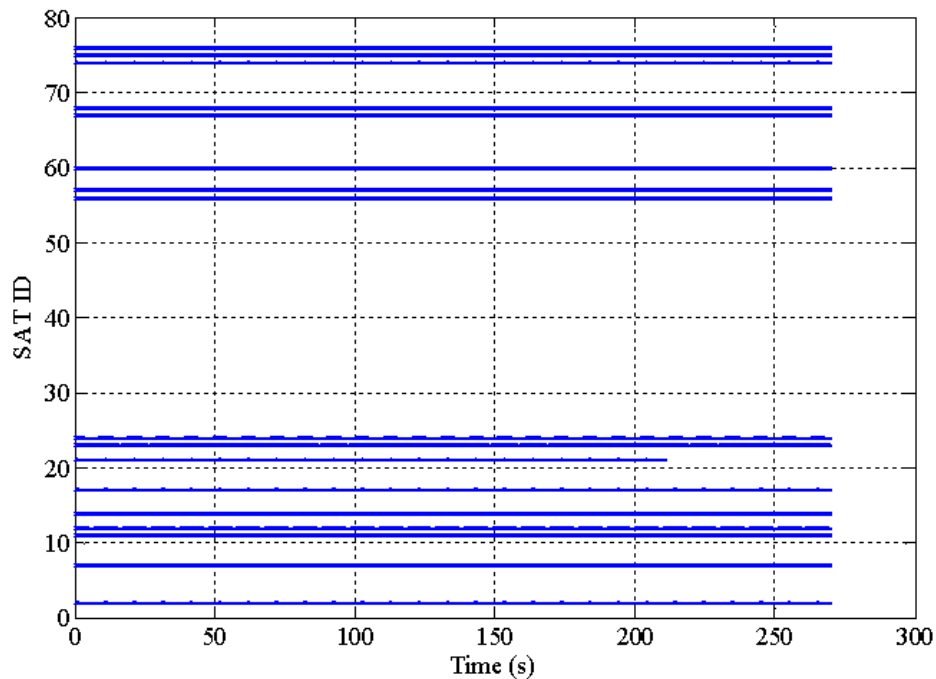
Figure 43: Set 1 - Ids of tracked GPS and GALILEO satellites as a function of time

129
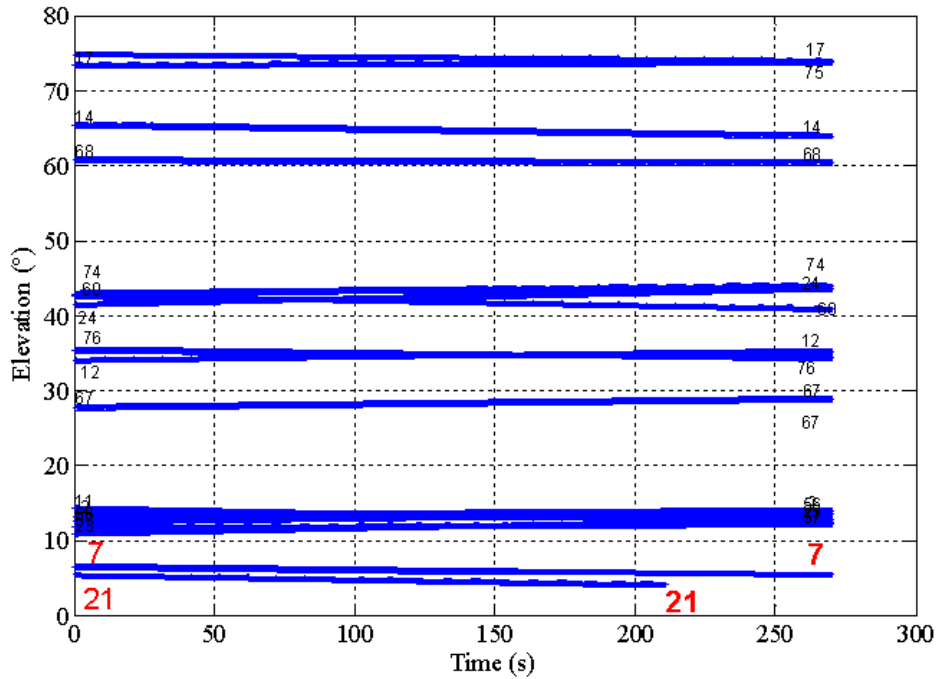
Figure 44: Set 1 - Elevation angle of GPS and GALILEO satellites as a function of time

In the case of the constellation freezing algorithm, satellites which are tracked and which elevation is above the applicable mask angle can be automatically removed if they are predicted to be lost after the FAF and before the end of the landing. This behaviour can be illustrated thanks to Figure 42, Figure 43 and Figure 44.

First, we can see that the GPS satellite number 21 is no more used in the position solution at around $t = 45\ s$ since it crosses the GPS mask angle of 5°. This results in a decrease of the vertical error as well as the RAIM VPL and in a small increase of the VDOP. This is a normal behaviour.

The algorithm impact is illustrated by satellite number 7. In fact, we can see in Figure 42 that the satellite is removed from the position at around $t = 145\ s$ from the beginning of the approach. This is due to the fact that the algorithm has predicted the crossing of the mask angle of this satellite which happens at about $t = 210\ s$ according to Figure 44. Thus, as we can see, the removal of this satellite instead of occurring after the FAF, is forced by the algorithm at $t = 145\ s$. It results in a step in the VPE and a small increase of the associated RAIM VPL. Thus, the algorithm allows stabilizing the position solution estimation error and the RAIM protection levels during the final approach phase which takes place after the FAF and until the end of the landing.
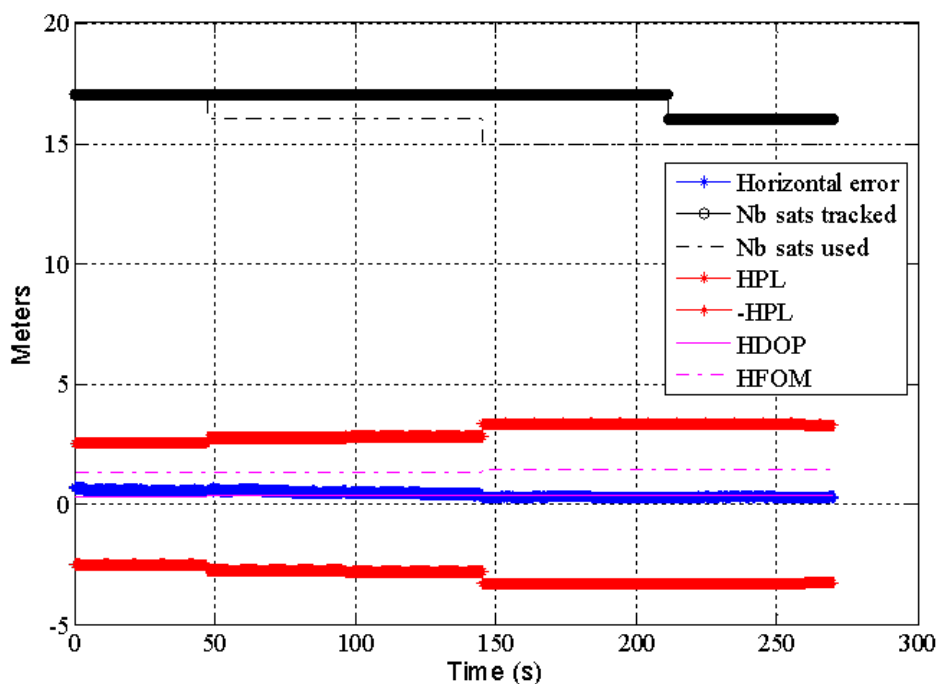
Figure 45: Set 1 – impact of constellation changes on Horizontal Position Error (HPE) in meters, HPL, HDOP and HFOM with constellation freezing algorithm

The same behaviour can be observed for the horizontal components. In fact, we can see that the prediction of the loss of satellite 7 from the position solution induces a step in the horizontal position error as well as an increase of the HPL before the FAF thanks to the constellation freezing algorithm.

5.4.2.1.2   Set 2 - First approach run among the 433, Tokyo airport: anticipation of the inclusion of a satellite

The second impact of the constellation freezing algorithm is presented in the following figures. We have seen in the previous section the case where a satellite is predicted to be lost during the Final Approach that is to say after the FAF. The algorithm role is to prevent this loss so as to avoid undesired position error steps due to constellation changes. However, these steps can also be provoked by the inclusion of a rising satellite in the position solution. Thus, the constellation freezing algorithm has to predict this type of constellation changes as well.

The results obtained during simulation Set 2, which are illustrated in Figure 46, show that two types of constellation changes occurred. If we refer to Figure 48 we can see that a GALILEO satellite is setting after the FAF and that a GPS satellite is rising after the FAF.
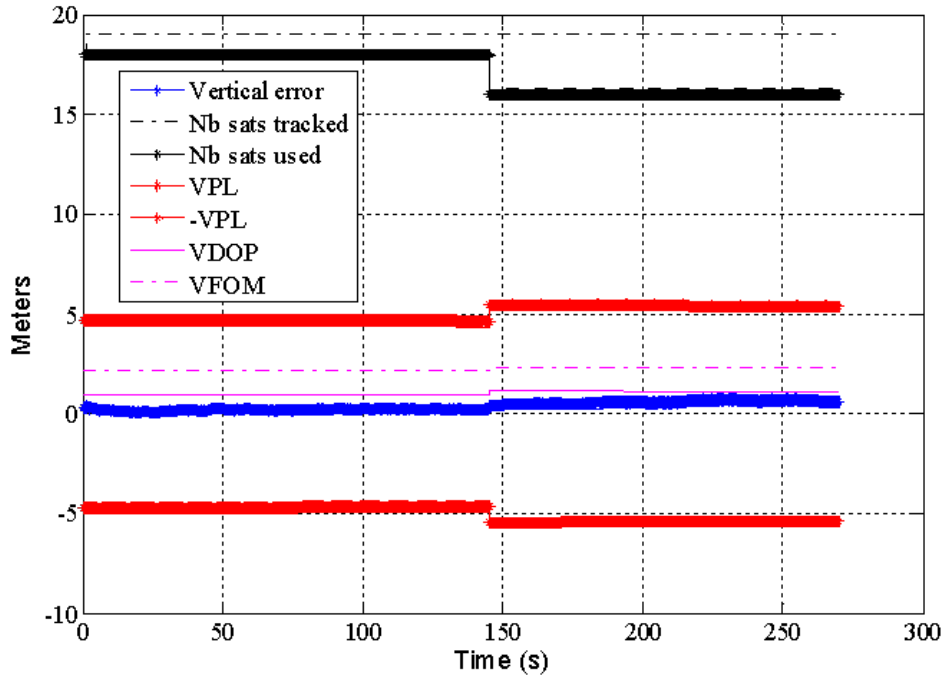
Figure 46: Set 2 – impact of constellation changes on Vertical Position Error (VPE) in meters, VPL, VDOP, VFOM with constellation freezing algorithm

If we look at $n_{tracked}$ and n_{used} on Figure 46 we can see that before the FAF, the GPS satellite is already tracked but not used in the position solution since it is below the applicable mask angle: $n_{tracked} = 19$ and $n_{used} = 18$. This simulation set has already been described previously when the constellation freezing algorithm is not used thanks to Figure 30. We saw at this moment that the satellite was passing the applicable mask angle at about 160 seconds and was thus immediately included in the position solution.
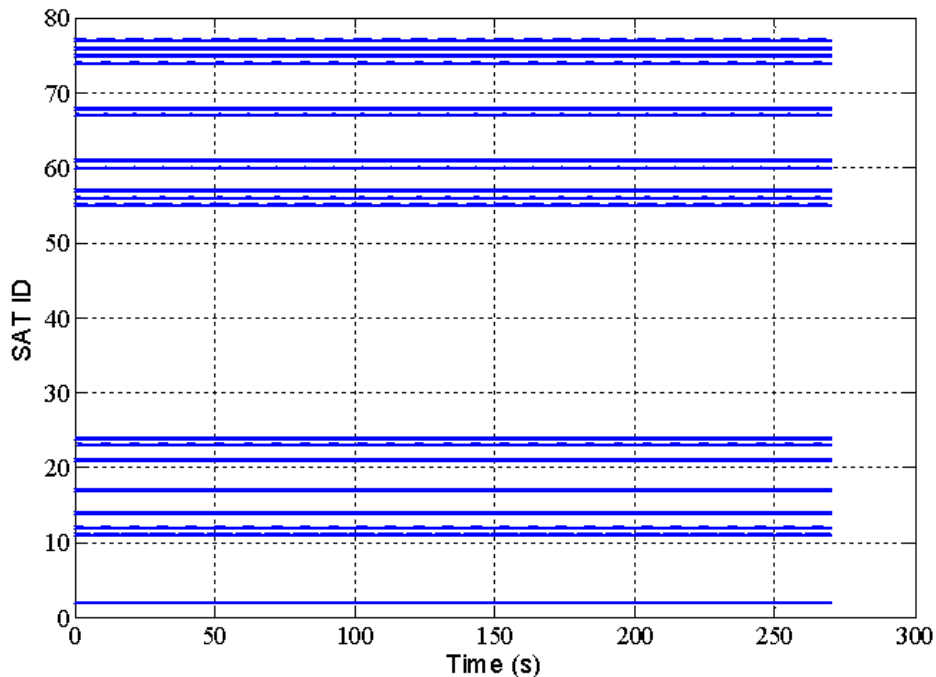
Figure 47: Ids of tracked GPS and GALILEO satellites as a function of time

We can see in Figure 46 that with the constellation freezing algorithm the rising satellite is not included in the position solution in order to avoid the associated vertical position error step.
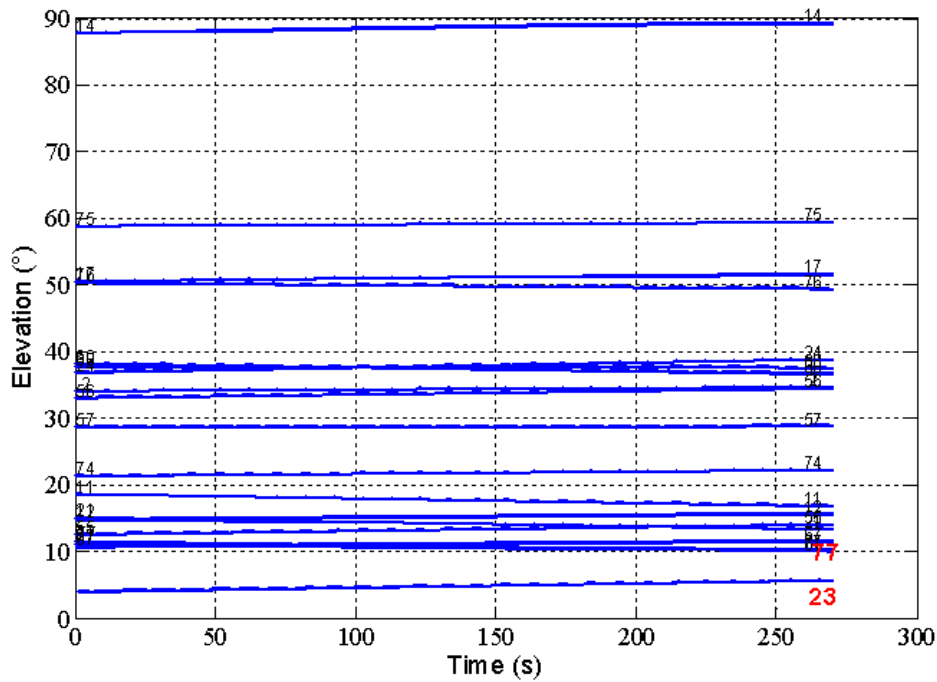
Figure 48: Elevation angle of GPS and GALILEO satellites as a function of time

Moreover we can see that a satellite is artificially removed from the position solution at the FAF since it is predicted as setting after the FAF and before the touchdown. This case is interesting since if we compare this result with Figure 48 we can see that in fact the satellite is setting just after the touchdown of the aircraft and thus it could have been used during the approach. Thus, in this case, our algorithm has erroneously removed the satellite from the position solution due to the margins implemented and the simplicity of our prediction model (linear extrapolation). However, we can consider that since there are still 16 satellites available, this error is not an issue.
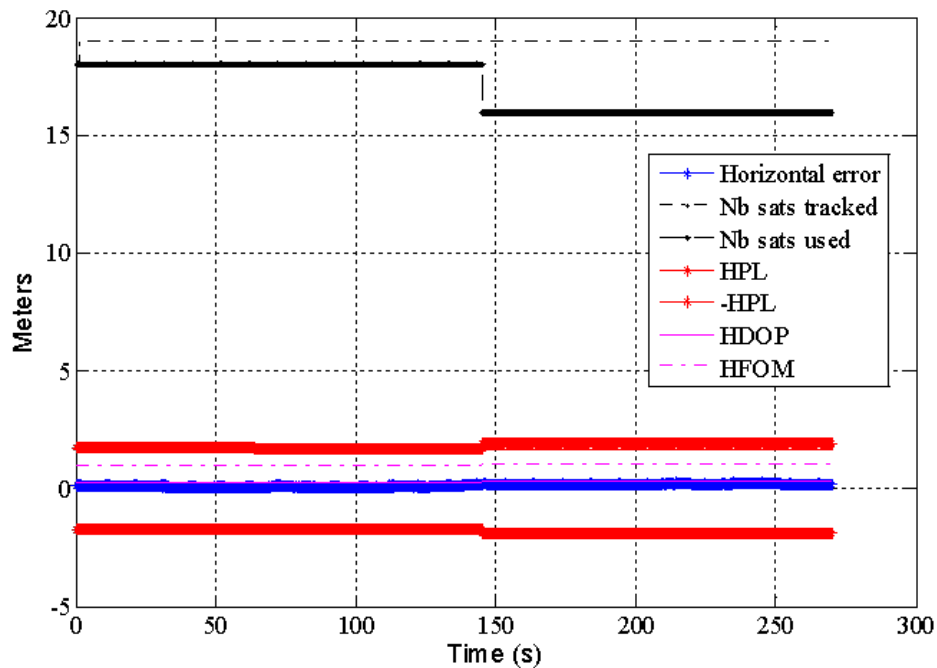
Figure 49: Set 2 – impact of constellation changes on Horizontal Position Error (HPE) in meters, HPL, HDOP and HFOM with constellation freezing algorithm

Figure 49 shows that the same type of behaviour as for the vertical components can be observed on the horizontal components.

5.4.2.2 Statistical results

5.4.2.2.1 Results over the complete approach

Figure 50 represents the histogram of the estimated mean of the vertical position error. As previously, each average value has been computed using the 1350 output samples produced during one simulated approach and the histogram is thus composed of 6928 mean values. The obtained distribution has a mono-modal Gaussian shape. As we can see, the observed value of the estimated average vertical position error over the complete approach ranges in [-3.89 m, 3.54 m]. In fact, we can see that it is slightly smaller than the values observed in the previous simulations, without the algorithm, in which we observed that the estimated average vertical position error was ranging in [-4.39 m, 4.19 m]. Please refer to Figure 34.
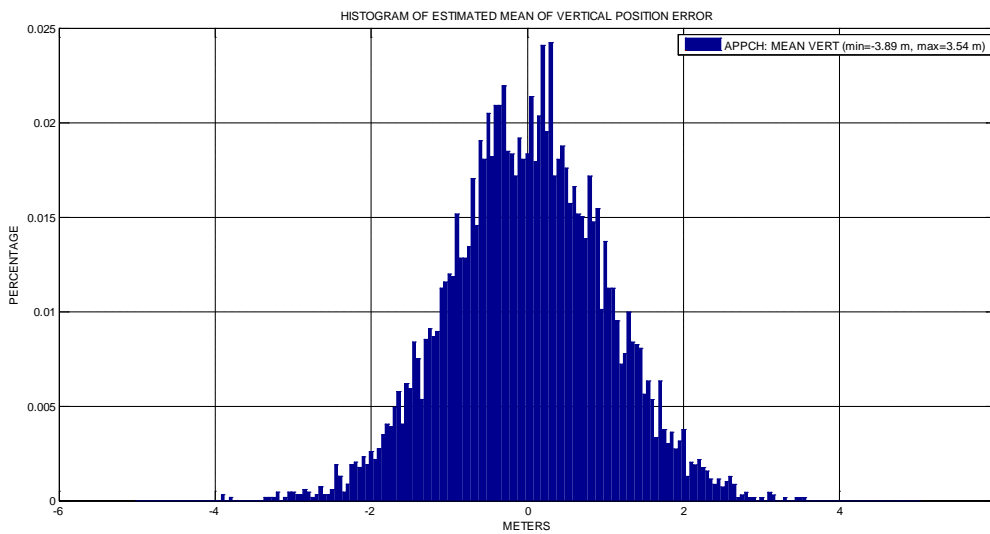
Figure 50: Histogram of estimated mean of VPE over all simulation runs for the complete approaches

In fact, it is possible that by using the proposed algorithm to fix the satellite constellation after the FAF, we have improved the receiver positioning accuracy. The algorithm induces two different behaviours:

- It can anticipate the loss of a satellite during final approach and force this loss to happen before sequencing the FAF. In this case, since the duration of the approaches are quite small we can imagine that the position error introduced due to the loss of a satellite is the same if it occurs naturally or if it is triggered by the algorithm. However, since the satellites are at low elevations, the associated errors such as ionosphere and troposphere may vary very rapidly, inducing significant residual errors differences before and after the FAF.
- It can anticipate the inclusion of a new satellite in the position solution and avoid it. In this case, we can imagine two different impacts. The inclusion of a new satellite could increase the accuracy of the position solution due to the enhancement of the DOP for example. Thus, by forbidding this inclusion we may degrade the position solution. However, in our case the mean position error seems lower. It could be explained by the fact that due to the high number of available satellites, it could be more beneficial to not include the new satellites due to the trade-off between the improvements of the constellation geometry with respect to the additional measurement errors introduced by the new satellite. The results presented in Figure

50, seems to indicate that the additional noise, multipath, ionosphere, troposphere uncertainties brought by the new satellite overcome the improvement of the constellation geometry.

However, deep understanding of the real reasons for these results to appear has not been reached yet and additional studies will be carried out.
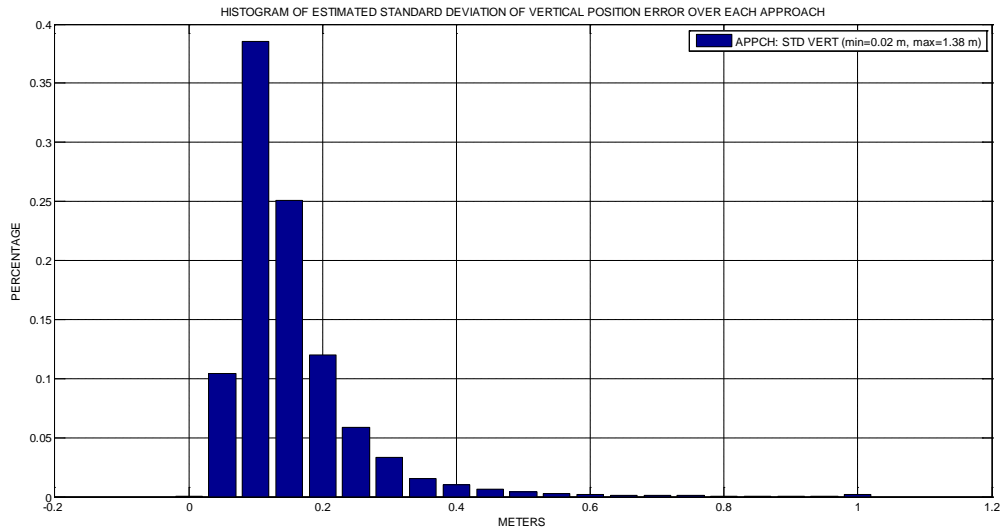
Figure 51: Histogram of estimated std of VPE over all simulation runs for the complete approaches

Figure 51 shows the histogram of the estimated standard deviation of the vertical position error. The estimated standard deviation of the vertical position error over the 1350 samples of the complete approach ranges in [0.02 m; 1.38 m]. This histogram is nearly the same as the equivalent one for the initial simulations ran without the proposed constellation freezing algorithm and presented in Figure 35. The standard deviation of the vertical position error thus does not seem to be impacted.

We can conclude here that the accuracy of the position solution does not seem to be degraded by the implementation of the constellation freezing algorithm because of the trade-off between the improvements of the average UERE through removal of low elevation satellites, while the DOPs remain essentially unchanged as it can be observed in the previous simulation set examples.
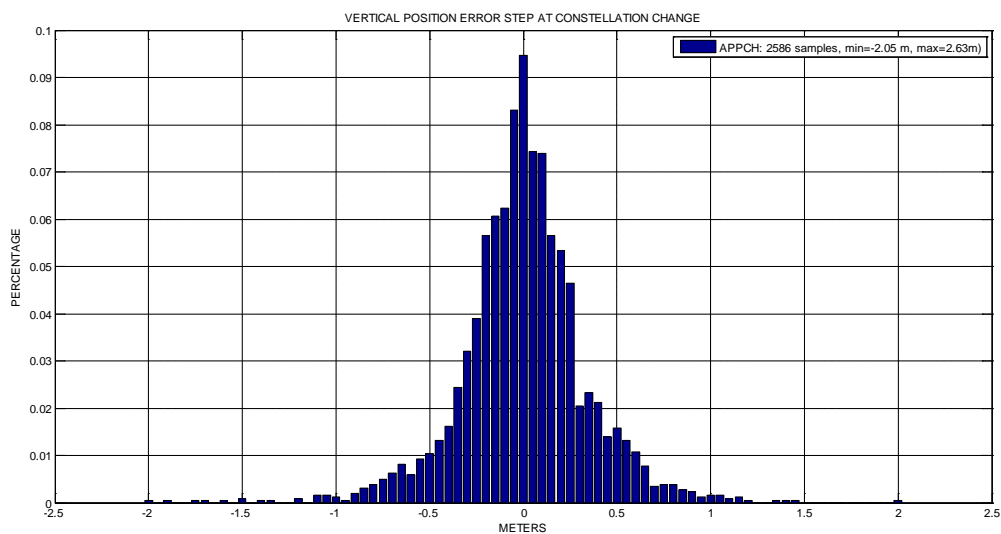
Figure 52: Histogram of VPE steps at constellation change over all simulation runs for the complete approaches

Figure 52 shows the histogram of the vertical position error (VPE) steps occurring when constellation changes occur, as observed during the 270s approaches triggered at 433 time samples over the 16 airports. We can see that 2586 constellation changes are observed. The vertical position error steps due to constellation changes range in [-2.05 m; 2.63 m]. This represents a small increase with respect to the results presented previously when the constellation freezing algorithm was not used since the maximum amplitude of vertical position error step was $1.96\ m$ as we can see in Figure 36. However, note that in Figure 52 the value of 2.63 m has only been observed once. This increase can be explained by the fact that in some simulations, we may force simultaneous removal of several satellites at the FAF.
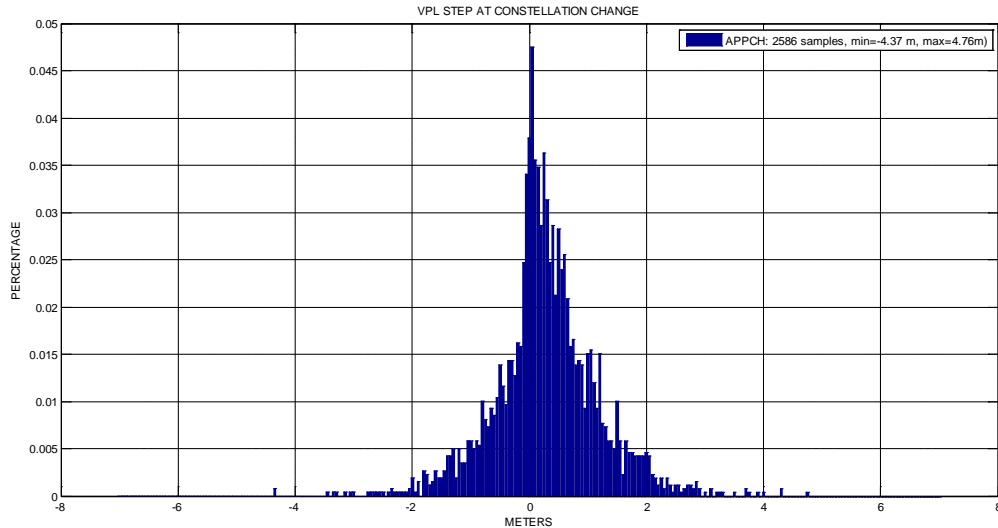
Figure 53: Histogram of VPL steps at constellation change over all simulation runs for the complete approaches

The same pattern can be observed in Figure 53 since the histogram of the observed steps of vertical protection levels induced by constellation changes illustrates that we observed steps ranging in $[-4.37; 4.76]\ m$ while the steps recorded in Figure 37 were ranging in $[-4.37; 4.32]\ m$. However, if we compare the results obtained with and without the constellation freezing algorithm we can conclude that the different histograms have globally the same shapes and that the quantities recorded have nearly the same behaviour. Thus, we can consider that the algorithm has a minor impact on the size of the steps in vertical protection levels and vertical position error and we can assume that it has degraded our position solution in a negligible way because these larger VPL and VPE values are isolated outlier samples.

We can notice that VPL was never observed to exceed the considered VAL or 35 meters.

The following section insists on the comparison of the position error before and after the FAF.

5.4.2.2.2   Results before and after the FAF
This section is equivalent to section 5.4.1.2.2 but this time we used the constellation freezing algorithm during our simulations. Thus, the intent here is to characterize the vertical position error as well as the vertical protection levels before and after the FAF and to illustrate the impact of the proposed algorithm.
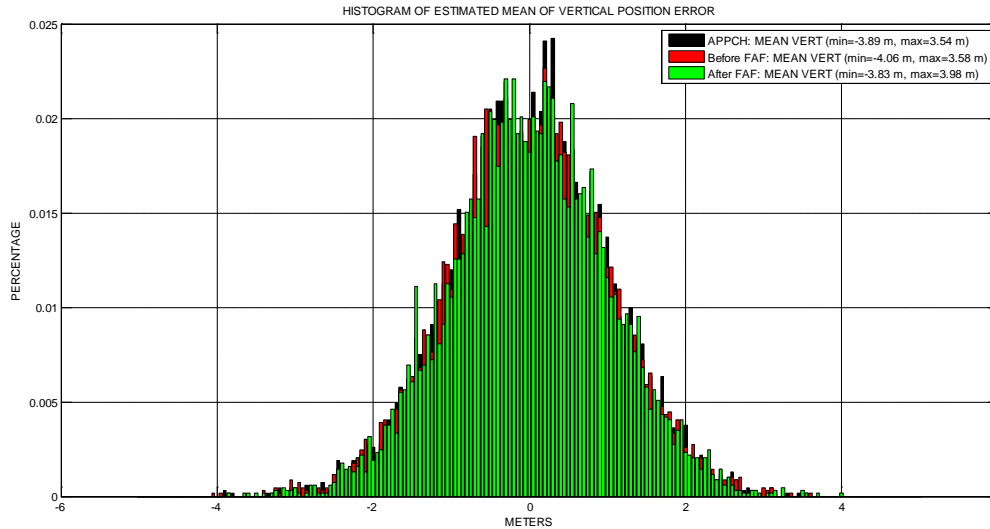
Figure 54: Comparison of the histograms of estimated mean of VPE over all simulation runs before and after the FAF with constellation freezing algorithm

Figure 54 allows comparing the histogram of the estimated average vertical position error:

- over the complete approach each mean value being computed using the $T_{APP} \times f_{pos} = 270 \times 5 = 1350$ samples of a complete approach
- before the FAF, each mean value being computed using $T_{FAF} \times f_{pos} = 160 \times 5 = 800$ samples
- after the FAF, each mean value being computed using $(T_{APP} - T_{FAF}) \times f_{pos} = 550$ samples

Since we did $n_{app} = 433$ simulations over $n_{air} = 16$ different airports each histogram is composed of 6928 values of estimated mean. As previously, each distribution has a mono-modal Gaussian shape. The different minimum and maximum values of each histogram are gathered in the following table:

	Minimum estimated mean VPE	Maximum estimated mean VPE
Complete approach	-3.89	3.54
Before FAF	-4.06	3.58
After FAF	-3.83	3.98

Table 22: Minimum and maximum estimated mean VPE observed with constellation freezing algorithm

If we compare these results with the results obtained for the nominal configuration of the receiver and represented in Figure 38 we can see that we globally obtained lower mean VPE over the complete approach. No further conclusion can be deduced from the overall shapes of these distributions. The interest here is to observe the potential impact of the proposed algorithm by comparing the mean VPE before and after the FAF. We can see that globally the different distributions have nearly the same Gaussian shapes with minimum and maximum which are really close to each other. Thus we can conclude that the algorithm has improved the mean vertical position solution error in a negligible way.
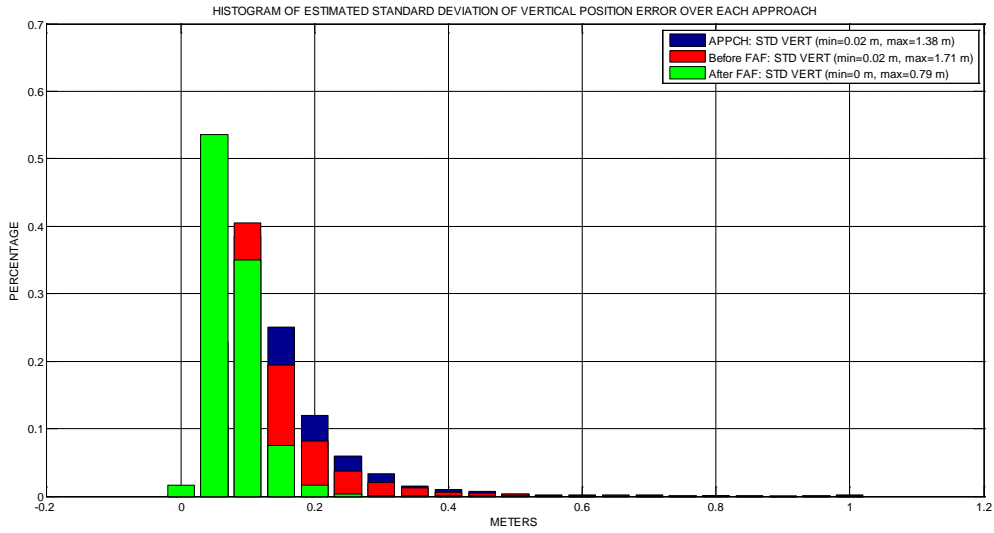
Figure 55: Comparison of histograms of estimated std of VPE over all simulation runs before and after the FAF with constellation freezing algorithm

We can observe the associated histograms of the estimated standard deviation of the VPE in Figure 55. The different minimum and maximum of these histograms are gathered in the following table:

	Minimum estimated std VPE	Maximum estimated std VPE
Complete approach	0.02	1.38
Before FAF	0.02	1.71
After FAF	0	0.79

Table 23: Minimum and maximum estimated mean VPE observed with constellation freezing algorithm

What we can see is that the implementation of the constellation freezing algorithm resulted in a reduction of the estimated standard deviation of the vertical position error after the FAF to a maximum value of $0.79\,m$ with our algorithm compared to a maximum value of $0.91\,m$ without. The counterpart of course is that the standard deviation of the VPE before the FAF has varied accordingly by increasing to a value of $1.71\,m$.

To conclude, when using the proposed algorithm, all steps resulting from a constellation change are forced to happen before the a/c sequences the FAF. No constellation change is occurring after the FAF and thus no VPE or VPL steps, resulting in the stabilization of the position error between the FAF and the touchdown.

5.5   Synthesis

This section presents the simulation results obtained with a GPS/GALILEO and RAIM simulator developed in the course of this study. The main objective of the simulations was to observe the impact of constellation changes on the position error at the output of the simulated receiver. In fact, constellation changes induce undesired steps in the NSE which could be potentially hazardous during precision approaches.

As we can see thanks to the obtained results, the position error is quite small, mainly due to the high redundancy in terms of satellites brought by the combination of two different GNSS constellations. We have to remark that even if we did not run complete availability simulations, the RAIM protection levels have never exceeded the relevant alert limits. Moreover, we can see that the fault-free combined GPS/GALILEO VPE is almost constant during an approach but can be as large as a few meters. The estimated standard deviation of the VPE can be a few 10 cm.

However, simulations have also proven that constellation changes still create unexpected position error steps.

To handle this phenomenon a simple algorithm has been developed to guarantee that there will not be error steps during the most critical part of the CAT I precision approaches which is considered to be defined between the FAF and the touchdown of the aircraft. To do so this algorithm makes a simple prediction of the constellation evolution by extrapolating the satellites position. It is thus capable to forecast which satellites will potentially rise or set during the final approach. The next step is to freeze the constellation at the FAF:

- Satellites which are predicted as setting during final approach are automatically removed at the FAF
- Satellites which are predicted to set during final approach are automatically inhibited and are thus not included

This algorithm thus results in no constellation changes, eliminating fault-free VPE and VPL steps during final approach. In addition, VPE variations after the FAF are highly reduced, as the estimated standard deviation of the VPE after the FAF is reduced to 0.79 meters as a maximum value while this standard deviation had a maximum value of 0.91 meters without the constellation freezing algorithm. The overall accuracy of the position solution is not affected by the proposed algorithm: we can observe that the average UERE is slightly improved while the DOPs are unchanged.

However, it has to be noticed that the maximum sigma before the FAF has consequently slightly increased. Moreover, the amplitude of the VPE steps due to constellation changes and observed before the FAF has slightly increased due to the forced simultaneous removal of several satellites at the FAF.

Chapter 6: GBAS autoland model

6.1   Introduction

GBAS is very attractive for civil aviation since it allows providing a navigation service with a very high level of performance in comparison with other augmentation systems.

GBAS is composed of a ground station able to compute differential pseudorange corrections and to monitor the quality of the Signal In Space (SIS). This station includes several receivers making pseudorange measurements used to elaborate pseudorange corrections and associated integrity data which are sent through a RF data link to the surrounding equipped aircrafts. Using this information, the user receiver is able to correct its own measurements but also to exclude some of them and to compute protection levels which are an evaluation of the confidence that the user can have in the final position solution.

Currently, GBAS has been certified for CAT I precision approaches, and it is anticipated that GBAS can provide further performance to meet CAT II/III requirements. This explains the interest of Civil Aviation, since GBAS may then become an alternative to classical ILS and MLS equipments which are currently the only mean to achieve CAT II/III precision approaches. Moreover, GBAS equipments are foreseen to be less expensive than ILS equipments in terms of installation, qualification, maintenance and needs for flights inspection. This possibility is actively investigated and ICAO and Industry standardization bodies are deriving new requirements for GBAS CAT II/III precision approaches.

The development of these new requirements follows a new concept using GBAS in an innovative fashion. Indeed, this new concept named "GAST D" results from a performance based approach taking credit of aircraft capabilities to allow the use of GBAS technology to reach CAT II/III minima, instead of putting all the constraints on the Signal in Space. In this new approach, there will be a transfer of responsibility from the ground station to the aircraft, unlike ILS. New requirements on the SIS and on the airborne side will impact the noise and the errors affecting the outputs of the fault-free on-board receiver which is the interface between GBAS SIS and the autopilot guidance laws.

In the future, civil aviation users desiring to take advantage of these new GBAS capabilities, it will be necessary to demonstrate their capability to land safely using GBAS CAT II/III and to obtain the certification of their solution. In particular, Autoland demonstrations for CAT II/III certification require numerous simulations to assess statistically the aircraft capability to autoland when the autopilot is receiving deviations from an ILS receiver for instance. It is thus necessary to identify precisely the GBAS GLS behaviour to perform autoland simulations, in line with applicable regulations for CAT II/III autoland operations [FAA, 1999] and [EASA, 2003].

It is therefore necessary to have a model representing GBAS L1 C/A NSE (Navigation Sensor Error) for autoland simulations purpose. It implies that this model will not be a high fidelity model, but a model adapted to demonstration of autoland capabilities. A model has been published in [MURPHY, 2005] which can model GBAS behaviour under nominal, limit and faulted conditions. However, validation methods used have not been described making it necessary to propose new material to provide a consolidated model. Moreover, this state of the art model was developed for CAT I simulations, and potentially for CAT II/III simulations, at a time when not all requirements were neither defined nor validated. Since then, CAT II/III requirements have been revised by ICAO and by Industry standards and therefore it is needed to update it to reflect latest developments [RTCA, 2008].

A proposal of modification of this model has been made to adapt it to GBAS CAT II/III in [MURPHY, 2009]. Before being incorporated in future GBAS CAT II/III regulations, it was proposed

to conduct an independent validation, and also to make an alternate proposal to update it if necessary so as to reflect recent evolutions of CAT II/III requirements (i.e. GAST-D).

We present here the results of our study that is to say our understanding of the GBAS NSE model initially proposed in [MURPHY, 2005]. Our results are compared to the legacy one and we propose a series of updates that appear to be relevant to us.

6.2   Requirements for the GBAS autoland model

The GBAS autoland model was intended to be proposed for standardization so as to be used to assess airworthiness of autoland in CAT II/III conditions using GBAS. For autoland certification, three different situations have to be considered [FAA, 1999]:

- Nominal: It corresponds to the variation of all parameters that may affect the position of the aircraft with respect to the touchdown box. In this case, the probability that the aircraft does not touch the ground in the touchdown box must not be higher than 1e-6.
- Limit Case: This situation must not be confused with a fault case. All parameters are varied in their nominal range distributions except one parameter which is set to its maximum value. Then, the probability that the aircraft does not land in the touchdown box must not exceed 1e-5.
- Malfunction Case: In this situation, the aircraft has to land in the touchdown box with a probability of one. The malfunction considered must be larger than a major impact. It means that it induces at the most a missed approach. Currently, this case is evaluated in simulation with a pilot.

Therefore, the model must be capable of:

- Reproducing the nominal GBAS NSE: the objective is to properly asses the nominal error and not a worst case error.
- Generating position steps: Steps are observed on the GBAS NSE due to constellation changes that is to say the inclusion or exclusion of a satellite from the position solution. This can occur when a satellite is crossing by the applicable mask angle but it can also occur when a satellite signal is lost because of outages for example. These steps can therefore be considered as part of the nominal GBAS NSE. Therefore, it must be properly represented by the model so as to evaluate the impact on the autopilot function.
- Representing fault mode conditions: It is necessary so as to demonstrate the hundred percents capability of the aircraft to land under these conditions.

This study will focus on the nominal GBAS NSE and on the steps since fault mode and limit case were out our scope.

6.3   State of the art GBAS NSE model

This section is intended to present our understanding of the model proposed for representing GBAS L1 C/A NSE in [MURPHY, 2005]. This model is illustrated in Figure 56.
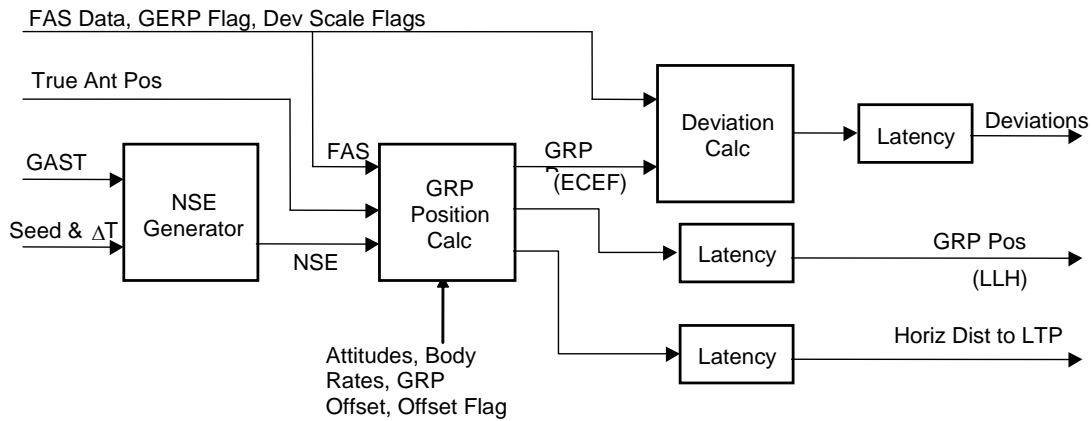
Figure 56: GBAS certification model [MURPHY, 2005]

What we are interested in is the NSE generator module. It is able to generate along track, cross track and vertical NSE. The NSE generator is detailed in the following figure:
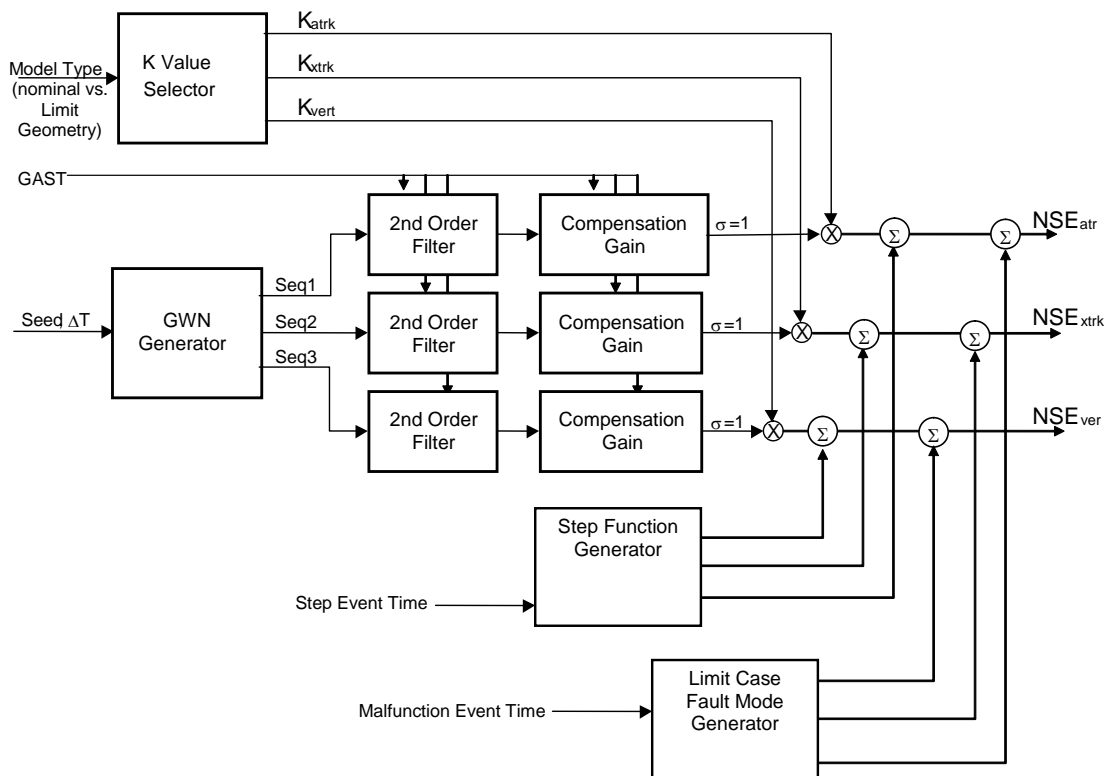
Figure 57: GBAS NSE generator [MURPHY, 2005]

To generate GBAS NSE along the three axes it produces three independent noise sequences $Seq1, Seq2 \ and \ Seq3$ with zero mean and unit variance that are then filtered by a second-order Butterworth filter and normalized. The filter outputs are then scaled by the NSE geometry scale factors $K_{atrk}, K_{xtrk}, K_{vert}$. These scale factors are drawn once at the beginning of the simulated approach and kept constant until the end. However, the duration of the simulations performed using

this model is not described in the reference model. This is what we can call the nominal NSE generation process.

As we can see in Figure 57, a step function generator allows generating steps and adding it to the nominal NSE.

Finally, Limit Case/Fault mode generator permits to generate ramps corresponding to various events. As previously explained, this module will not be detailed.

What is important to notice is that this model does not generate errors at the pseudorange level. Usually, to compute the position error, the process is to model and generate pseudorange measurement errors and then project them in the position domain. This method necessitates simulating the satellites constellation. On the contrary, the GBAS NSE model directly generates the position error. It is interesting because it lowers the number of operations necessary to run the model: Instead of simulating n satellite pseudorange measurement errors, it is only necessary to compute the 4 components of the position error. Knowing that n must be higher than 4 so as to be able to compute a GPS position and that in the case of GPS 8 satellites are visible in mean from a GPS receiver, it is easy to understand the gain obtained.

6.3.1   Nominal NSE generator
The nominal NSE generator is composed of different steps which are described here.

6.3.1.1   2nd order filter
We can see in Figure 57 that given a seed, three independent white Gaussian noise sequences $Seq1, Seq2 \ and \ Seq3$ with zero mean and unit variance are generated. Each noise sequence is filtered by a second-order Butterworth filter which has the following transfer function:

$$H(s) = \frac{\omega_n^2}{s^2 + \sqrt{2}\omega_n s + \omega_n^2}$$

(6-1)

With $\omega_n = 0.01 \ rad/sec$

Thus we can derive:

$$f_n = \frac{\omega_n}{2\pi} = 1.59 \times 10^{-3} \ Hz \text{ and } T_n = 628 \ s$$

The rationale for this filter is given in [MURPHY, 2009]:

Assuming that the error entering a GBAS receiver is white Gaussian, then the spectral content of the error on the smoothed pseudorange measurements is determined y two processes:

-   The code tracking loop (DLL) filtering: it is assumed first-order with a one sided bandwidth equal to 0.125 Hz. It is equivalent to a time constant of 2 seconds.
-   The carrier smoothing: in the case of GAST-C the carrier smoother is a first order filter with a time constant of 100 seconds.

It is explained that the frequency response of this cascade can be conservatively modelled using the above Butterworth filter. It is considered conservative since it places more power at low frequencies than the cascade does and that low frequency errors are more detrimental to autoland performance than

high frequency errors. This model has been validated by comparing its frequency response to the PSDs of smoothed pseudorange errors from several GPS receivers [MURPHY, 2009]

A compensation gain is used to scale the standard deviation of the output filter to 1. Therefore, we can see that the filter determines the temporal correlation of the GBAS position error and not the statistics of its amplitude. The compensation gain is obtained by modelling the standard deviation of the filter output when fed by a white noise. This standard deviation can be written:

$$\sigma_x = \sqrt{\frac{\omega_n}{2\sqrt{2}}} \sigma_u$$

(6-2)

Where:

- σ_u is the standard deviation of the input sequence
- σ_x is the standard deviation of the output sequence

Then, assuming the input sequence has a unit variance i.e. $\sigma_u{}^2 = 1$ we obtain the compensation gain G_x:

$$G_x = \sqrt{\frac{2\sqrt{2}}{\Delta T . \omega_n}}$$

(6-3)

Where ΔT is the time step size due to the discrete implementation of the filter.

6.3.1.2   NSE magnitudes
The NSE scale factors $K_{atrk}, K_{xtrk}, K_{vert}$ are used to control the standard deviation of NSE errors in the vertical, along-track and cross-track directions. They can thus be assimilated to the standard deviations of the NSE $\sigma_{atrk}, \sigma_{xtrk}, \sigma_{vert}$. We detail here the model used for these quantities.

6.3.1.2.1   Vertical NSE magnitude K_{vert}
The vertical accuracy variations over time due to changes in satellite geometry are modelled by K_{vert}. In the legacy model, it is obtained by sampling a value x from a uniform distribution between 0 and 1. Then, K_{vert} is obtained using x as an input to the following functions [MURPHY, 2009]:

- **GAST-C:**

$$K_{vert} = f(x) = 0.4 + 0.2x - \frac{0.004}{x - 1}$$

(6-4)

- **GAST-D:**

$$K_{vert} = f(x) = 0.73 + 0.36x - \frac{0.007}{x - 1}$$

(6-5)

The GAST-C function was derived from satellite geometry simulations conducted for different locations, time of day, satellite outages. These simulations were carried out only for LAAS coverage area and therefore it must be checked whether these results are applicable to other regions of the world such as Europe. Moreover, the simulations were not run over a grid of locations but for seven airports locations which were considered as representative.

The final K_{vert} is a standard deviation which is supposed to over bound the maximum values of standard deviation observed during simulations. This model is therefore supposed to be conservative.

The GAST-D function was simply deduced from the GAST-C function. In fact, one of the main evolutions introduced by GAST-D with respect to GAST-C is that the position will be computed with a different smoothing time constant of 30 seconds. Therefore, it is possible to assume that the NSE magnitude variations between the two concepts will be directly linked to the ratio between the two different time constants. This can be illustrated by using the expression presented in Chapter 4 to represent the impact of the code carrier smoothing on the pseudorange measurement errors [HEGARTY, 2006]:

$$\sigma_{\tilde{P}}^2 \approx \frac{\sigma_P^2}{2T_{smooth}}$$

(6-6)

Where:

- T_{smooth} is the time smoothing constant in seconds
- σ_P^2 is the raw code pseudorange measurement error variance
- $\sigma_{\tilde{P}}^2$ is the smoothed code pseudorange measurement error variance

If we consider the impact of two different time smoothing constant T_1, T_2 on the magnitude of pseudorange measurement errors we have:

$$\sigma_{\widetilde{P1}}^2 \approx \frac{\sigma_P^2}{2T_1}$$

$$\sigma_{\widetilde{P2}}^2 \approx \frac{\sigma_P^2}{2T_2}$$

It is then possible to compare the two resulting standard deviations by using the ratio:

$$\alpha = \frac{\sigma_{\widetilde{P1}}}{\sigma_{\widetilde{P2}}} = \sqrt{\frac{T_1}{T_2}}$$

Finally, to expected increase in the magnitude of pseudorange measurement errors introduced by the new smoothing constant of 30 seconds is given by:

$$\alpha_{GAST\_C/GAST\_D} = \frac{\sigma_{\tilde{P}_{GASTC}}}{\sigma_{\tilde{P}_{GASTD}}} = \sqrt{\frac{100}{30}} = 1.8319$$

(6-7)

6.3.1.2.2 Horizontal NSE magnitude K_{atrk}, K_{xtrk}

It is explained in [MURPHY, 2009], that during simulations the ratio between cross-track and vertical position error sigmas was monitored. To do so, they computed the covariance matrix of the position solution. Of course, since the coordinate frame in the horizontal plane is defined relatively to a given heading, it is necessary to make an additional assumption. For each simulation they considered the heading producing the largest cross-track error.

The idea behind this is to determine a unique ratio so as to deduce the horizontal components from the vertical one. The advantage is of course the simplification of the model.

It is explained that a curve fit to the distribution of the ratio $\sigma_{xtrk}/\sigma_{vert}$ as a function of σ_{vert} was computed. It is shown in Figure 58.
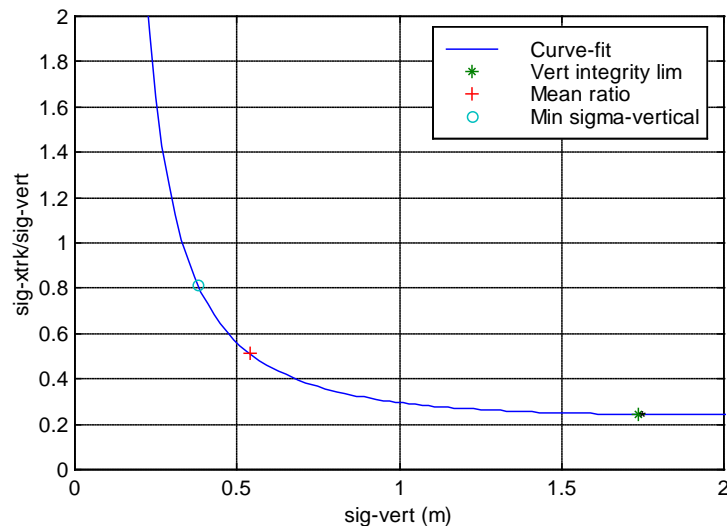
Figure 58: Curve fit to $\sigma_{xtrk}/\sigma_{vert}$ vs σ_{vert} Distribution [MURPHY, 2009]

Three different candidate ratios were selected:

- The ratio for which σ_{vert} would result in the vertical protection level to exceed the vertical alert limit
- The mean ratio
- The ratio corresponding to the smallest observed σ_{vert} value

The strategy to select the ratio was then to determine which one was conservative without being unrealistic. To do so they computed the probability that the real σ_{xtrk} is larger than the generated one for each candidate ratio. The results are gathered in the following table:

	Ratio	K_{xtrk} (m)	Probability
Vertical Integrity limit	0.245	0.425	1×10^{-2}
Mean ratio	0.515	0.894	2×10^{-4}
Minimum σ_{vert}	0.818	1.420	2×10^{-5}
HAL/VAL	4	6.942	4×10^{-8}

Table 24: K_{xtrk} candidates

The value corresponding to the smallest observed value of σ_{vert} was finally selected, since it was considered as conservative but not too much.

Therefore, the cross-track scale factor can be easily derived from the vertical scale factor. An additional assumption is to consider that cross-track and along-track errors have the same magnitude. The model used for horizontal scale factors is thus given by:

$$K_{xtrk} = K_{atrk} = 0.818 \times K_{vert}$$

(6-8)

As we can see this approach tends toward a very simple model. This may be explained by the context of autoland simulations which are very constraining in terms of computational burden since many different parameters have to be modelled and varied at the same time.

6.3.1.2.3 Vertical/Lateral errors correlation

As we explained previously the method used to model the GBAS NSE is to generate three independent noise sequences $Seq1, Seq2 \ and \ Seq3$ with zero mean and unit variance. These sequences are then filtered and scaled using the NSE scale factors $K_{atrk}, K_{xtrk}, K_{vert}$.

One potential drawback of the method used to generate vertical and lateral NSE is that the initial zero-mean Gaussian sequences $Seq1, Seq2 \ and \ Seq3$ are completely uncorrelated. In fact, the projection of pseudorange measurement errors on the position components implies in reality a correlation between each component of the position error.

Therefore, the authors of the state of the art model have investigated the necessity to implement an additional algorithm to produce correlated samples out of the three second order filters outputs. The algorithm used is not presented here since they finally concluded that it was not necessary.

They obtained the results gathered in Figure 59. As we can see, the correlation coefficients of two sequences were compared. The first one corresponds to the output sequences of the second order filter presented in section 6.3.1.1. The second one is the same sequence on which was additionally applied the correlation algorithm. This correlation algorithm is described later in this document in our numerical analysis of this model. As we can see, the two sequences are nearly the same.
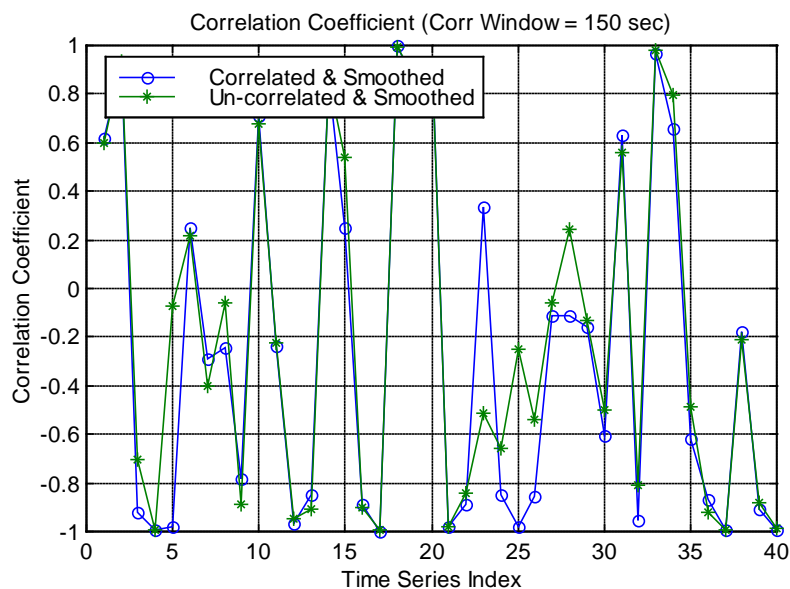
Figure 59: Study of the impact of the correlation algorithm [MURPHY, 2009]

149

To confirm this result, they computed the correlation coefficient distribution for each sequence. We can see in Figure 60 that the level of correlation is nearly the same for the two sequences. This may be explained by the fact that the second order filter already introduces enough correlation between successive samples by reproducing the 100 second smoothing.
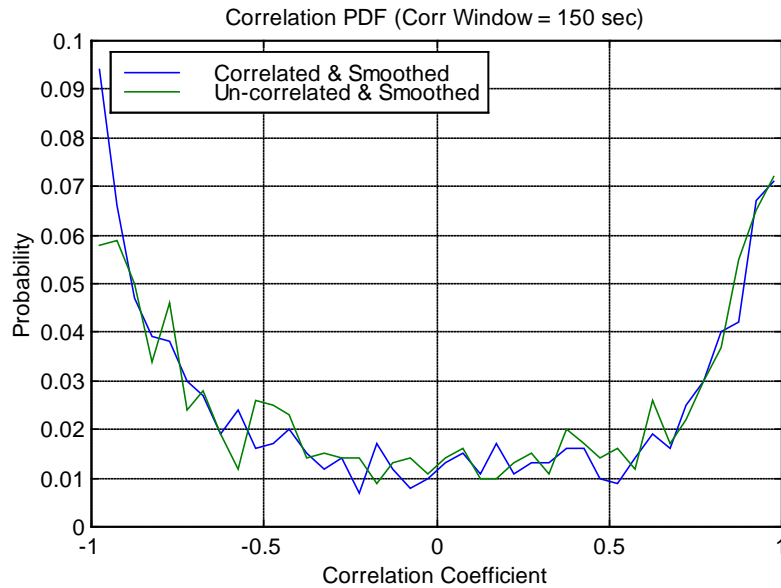
Figure 60: Correlation coefficient distribution comparison [MURPHY, 2009]

6.3.2 Step function generator

The step function generator module has been introduced in Figure 56. The detailed architecture can be seen in Figure 61.
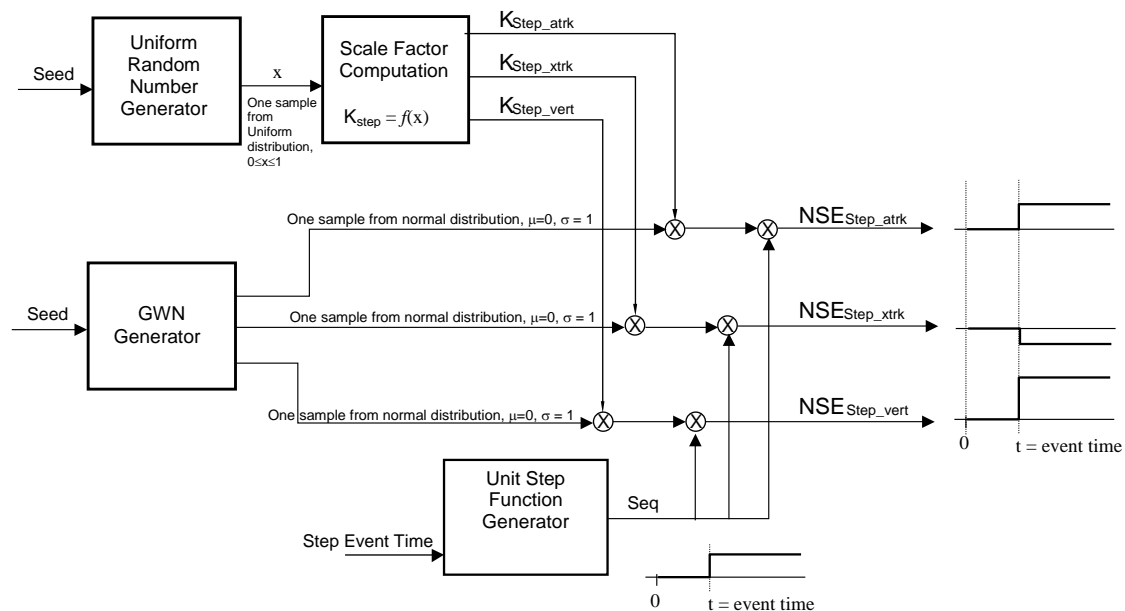
Figure 61: GBAS NSE step generator [MURPHY, 2009]

Step errors occur when individual satellites are removed from or included into the position solution. Therefore, a step can occur simply due to constellation evolution (rise of a satellite for example) or due to a Signal In Space (SIS) failure. The step function generator must therefore contribute to the

nominal NSE generator but also to the faulty model by representing the impact of the failures inducing the loss of a satellite.

The step function generator produces representative NSE step errors in the vertical, along-track and cross-track directions. First, three random samples are drawn from a zero mean and unit variance normal distribution. Then, these samples are multiplied by scale factors $K_{vert}^{step}, K_{xtrk}^{step}, K_{atrk}^{step}$ that are chosen to represent the statistical variation in the size of an error that would result from the variations in the relative geometry between the user and the satellites. Finally, the obtained constant factors are used to scale a unit step function time sequence. Therefore, the dimensioning parameters of the model are the scale factors statistics that it to say their distributions as well as the probability of occurrence of these events.

6.3.2.1 Step events
Two types of step events were considered:

- Single or Multiple Rise/Set event: this event is the inclusion or removal of one or several satellites from the position solution due to the nominal evolution of the geometry between the user and the constellation.
- Single or Dual Signal Loss event: this event is the removal of one or several satellites from the position solution due to GPS SIS aberrations.

For Rise/Set events, the probabilities of occurrence were computed over two different time windows of 15s and 150s by simulating the constellation geometry evolution over the GPS ground track repeat period and over a user location grid covering the Northwest quadrant of the world as indicated in [MURPHY, 2009]. Each result for Rise/Set events and presented in Table 25 corresponds to the locations which experienced the largest probabilities of occurrence of rise/set events, thus each probability may correspond to a different location.

For single and dual Signal Loss events, it is not possible to predict the value of a probability of occurrence. The probabilities presented in Table 25 in the case of single and dual signal loss events are empirical probabilities provided in [MURPHY, 2009].

The probabilities of single or dual satellite signal loss were derived using the assumption of a maximum of 3 satellite failures per year [GPS SPS, 2008]. We present here the computation published in [MURPHY, 2009]. We can compute the probability of occurrence of a major satellite failure:

$$P_{major} = \frac{n_{failure\ per\ year}}{365\ days \times 24\ hours} = \frac{3}{365 \times 24} = 3.42 \times 10^{-4}\ failures\ per\ hour$$

In [MURPHY, 2009], they considered the conservative assumption of 12 satellites in view from the receiver and computing the probability of a single signal loss during the exposure intervals of interest which are $150\ s$ and $15\ s$:

$$P_{single\_loss\_150} = P_{major} \times \frac{12}{24} \times \frac{150}{3600} = 7 \times 10^{-6}$$

$$P_{single\_loss\_15} = P_{major} \times \frac{12}{24} \times \frac{15}{3600} = 7 \times 10^{-7}$$

The probabilities of experiencing two independent failures are given by:

$$P_{dual\_loss\_150} = \left(P_{major} \times \frac{12}{24}\right) \times \left(P_{major} \times \frac{11}{24}\right) \times \frac{150}{3600} = 1.1 \times 10^{-9}$$

$$P_{dual\_loss\_15} = \left(P_{major} \times \frac{12}{24}\right) \times \left(P_{major} \times \frac{11}{24}\right) \times \frac{15}{3600} = 1.1 \times 10^{-10}$$

These results are gathered in the following table.

Configuration Change	Time Interval	
	150 s	**15 s**
Single Rise Set	0.167	0.0167
Multiple Rise Set	0.0222	0.00222
Single SV signal loss	7×10^{-6}	7×10^{-7}
Dual SV signal loss	1.12×10^{-9}	1.12×10^{-10}

Table 25: Probability of step events during exposure intervals [MURPHY, 2009]

6.3.2.2 Vertical steps magnitude K_{vert}^{step}

It is explained in [MURPHY, 2009] that the step error magnitude due to the loss of a particular satellite n is a Gaussian random variable which variance can be computed as:

$$\sigma_{step\_vert}^2 = \sigma_{vert}^2(k_0) - \sigma_{vert}^2(k_0 + 1)$$

(6-9)

$$\sigma_{step\_vert}^2 = (a_{vert}^n . \sigma_n)^2 + \sum_{i=1}^{n-1} \left(\left(a_{vert}^i(k_0) - a_{vert}^i(k_0 + 1)\right). \sigma_i\right)^2$$

(6-10)

Where:

- k_0 is the epoch just before the loss of a satellite signal
- σ_i is the pseudorange measurement error standard deviation for satellite i
- σ_{vert}^2 is the variance of the vertical GBAS NSE
- $\sigma_{step\_vert}^2$ is the variance of the amplitude of the resulting step
- a_{vert}^i represents the coefficients of the vertical part of the projection matrix A presented in Chapter 4.

Using these equations during simulations, they were able to build histograms of $\sigma_{step\_vert}^2$ and deduce an observed distribution. The following functions were derived so as to reproduce this observed distribution in the GBAS NSE generator. The function for GAST-C and the proposed function for GAST-D are reminded here [MURPHY, 2009]:

GAST-C:

$$K_{vert}^{step} = \sigma_{step\_vert} = 0.22 + 0.22x - \frac{0.0132}{x - 1.01}$$

(6-11)

GAST-D:

$$K_{vert}^{step} = \sigma_{step\_vert} = 0.4 + 0.4x - \frac{0.0242}{x - 1.01}$$

(6-12)

Where x is selected from a uniform distribution between 0 and 1.

6.3.2.3   *Horizontal steps magnitudes K_{xtrk}^{step}, K_{atrk}^{step}*

The relation used between horizontal and vertical nominal NSE has been also used for steps NSE. Therefore, horizontal steps magnitudes can easily be deduced from vertical steps magnitude using the same ratio:

$$K_{xtrk}^{step} = K_{atrk}^{step} = 0.818 \times K_{vert}^{step}$$

(6-13)

6.3.3   Pseudorange measurement error models

We briefly discuss here the pseudorange measurement models used in [MURPHY, 2009] so as to obtain the scale factors function presented previously. It is clear that the position error is closely related to the characteristics of the pseudorange measurement errors. Thus it is critical to use adequate models.

In the simulation run to develop the GBAS NSE generator according to [MURPHY, 2009] the pseudorange measurement error models were computed on the basis of the requirements published in DO-245 [RTCA, 1998]. These models are dependent on the Ground Accuracy Designator of the GBAS ground station and the Airborne Accuracy Designator (AAD) of the airborne GBAS capable receiver. The associated expressions are taken from [MURPHY, 2009]:

$$\sigma_i^2 = \sigma_{pr\_gnd,i}^2 + \sigma_{pr\_air,i}^2 + \sigma_{mp,i}^2$$

(6-14)

$$\sigma_{pr\_gnd,i}^2 = \underbrace{\frac{\left(0.16 + 1.07e^{-\frac{\theta_i}{15.5}}\right)^2}{2} + 0.08^2}_{GAD\ B2} + \underbrace{\left(\frac{0.03}{\sin(\theta_i)}\right)^2}_{Residual\ Iono\ and\ Tropo}$$

(6-15)

$$\sigma_{pr\_air,i}^2 = \underbrace{\left(0.11 + 0.13e^{-\frac{\theta_i}{4}}\right)^2}_{AAD\ B}$$

(6-16)

$$\sigma_{mp,i}^2 = \left(0.2e^{-\frac{\theta_i}{75}}\right)^2$$

(6-17)

These models have been updated in a more recent issue of DO-245 which is DO-245A [RTCA, 2004]. In fact, weaknesses have been identified since then in the models presented in this section. In particular, the multipath model was proven to fairly underestimate the multipath impact on the measurements. That's why during simulations we used the updated expressions of the GBAS GPS L1

C/A pseudorange measurement error models which can be found in [RTCA, 2004]. These models are reminded in Chapter 4.

6.4 Mathematical Analysis of the legacy model

6.4.1 Nominal NSE determination

6.4.1.1 NSE Time behaviour

The evolution of the position errors is driven by the 2nd order Butterworth filter. It has a time constant of $T_n = 628s$ which is very large in comparison with the time series that has to be generated in the GBAS autoland simulations. It implies that it has to be properly initialized which is not very easy.

We present here the analysis we carried out to justify this model. However, this task was not very easy and we present here the elements that we obtained. However, these results have not been validated and in particular not cross-checked in a real situation with real receivers.

The legacy model [MURPHY, 2005] assumes that all error components can be assimilated to white noise filtered by the equivalent transfer function of the tracking loops and by the first order code-carrier smoothing filter. The analysis that follows discusses whether this assumption can be made or not.

The GPS L1 C/A signal entering the tracking loops can be written as:

$$r(t) = A.d(t - \tau).c(t - \tau).\cos(2\pi f_0 t + \theta) + m(t) + J(t) + n(t)$$

(6-18)

Where:

- A is the amplitude of the received signal
- d is the navigation message materialization
- c is the PRN code sequence materialization
- m is the multipaths signal
- J is the interference signal
- f_0 the carrier frequency
- θ the carrier phase
- $n(t)$ the noise affecting the measurements

We assume in the following that the effect of interference can be assimilated as an additional white noise contribution. So, we consider $J(t) = 0$ in the following.

The true group propagation delay is $\tau(k)$ reflecting the geometrical distance, the satellite-receiver clock offset, and the ionospheric and tropospheric delays.

We could thus define an equivalent true pseudorange as:

$$P(k) = c.\tau(k) = \rho(k) + c.\Delta t^i(k) - c.\Delta t^u(k) + I(k) + T(k)$$

(6-19)

Where:

- ρ is the true geometrical distance

- Δt^i is the satellite clock error
- Δt^u is the user clock error
- I is the ionospheric error
- T is the tropospheric error

We can also define $\hat{P}(k)$ the measured pseudorange as provided by the DLL.

The pseudorange measurement error can be written as:

$$\varepsilon_P(k) = \hat{P}(k) - P(k)$$

(6-20)

$$\hat{P}(k) = \varepsilon_P(k) + P(k)$$

(6-21)

If we denote $H(z)$ the equivalent closed loop transfer function of the equivalent linear model of the DLL then the pseudorange measurement error can be written:

$$\varepsilon_P(z) = [1 - H(z)]P(z) + H(z)N_P(z)$$

(6-22)

Where:

- $P(z)$ is the z transform of the true pseudorange value $P(k)$
- $N_P(z)$ is the z transform of the equivalent noise affecting the pseudorange measurements
- $\varepsilon_P(z)$ is the z transform of the pseudorange measurement error $\varepsilon_P(k)$

The estimated pseudorange is given as:

$$\hat{P}(k) = \rho(k) + \left[c.\Delta t^i(k) - c.\Delta t^u(k)\right] + \hat{I}(k) + \hat{T}(k) + \hat{m}(k) + \hat{b}(k)$$

(6-23)

Where:

- $\hat{I}(k)$ is the impact of the ionospheric delay on the pseudorange measurement
- $\hat{T}(k)$ is the impact of the tropospheric delay on the pseudorange measurement
- $\hat{m}(k)$ is the impact of the multipaths on the pseudorange measurement
- $\hat{b}(k)$ is the impact of the noise on the pseudorange measurement

We assume here that the satellite clock variations are ideally tracked. The next process executed in the receiver is the code carrier smoothing. We consider the steady-state carrier smoothing filter response. The smoothed pseudorange can be written as:

$$P_L(k) = \alpha.\hat{P}(k) + (1 - \alpha)[P_L(k - 1) + \varphi(k) - \varphi(k - 1)]$$

(6-24)

Where:

- $\alpha = T_s/T_{sm}$

155

- T_{sm} the code carrier smoothing time constant
- T_s is the pseudorange measurement output interval
- φ is the carrier phase measurement in meters

Then the different errors impact the smoothed pseudorange:

$$P_L(k) = \rho(k) + \left[c.\Delta t^i(k) - c.\Delta t^u(k)\right] + \hat{I}_L(k) + \hat{T}_L(k) + \hat{m}_L(k) + \hat{b}_L(k)$$

(6-25)

We can define the error affecting the smoothed pseudorange as:

$$\varepsilon_L = P_L(k) - P(k)$$

With :

$$P(k) = \rho(k) + c.\Delta t^i(k) - c.\Delta t^u(k) + I(k) + T(k)$$

(6-26)

Then we have:

$$\varepsilon_L(k) = \left[\hat{I}_L(k) - I(k)\right] + \left[\hat{T}_L(k) - T(k)\right] + \hat{m}_L(k) + \hat{b}_L(k)$$

(6-27)

Finally, the smoothed pseudoranges are corrected using the GBAS PRC:

$$P_L{}^C(k) = P_L(k) + PRC(k) + T_c(k) - c.\Delta t^i{}_c(k)$$

(6-28)

Where:

- T_c is the tropospheric correction provided by GBAS
- $\Delta t^i{}_c$ is the satellite clock correction computed from broadcasted data

Then we obtain:

$$P_L{}^C(k) = \rho(k) - c.\Delta t^u(k) + \varepsilon_I(k) + \varepsilon_T(k) + \varepsilon_{\Delta t^i}(k) + \hat{m}_L(k) + \hat{b}_L(k) + m_G(k) + b_G(k)$$

(6-29)

With:

- $\varepsilon_I(k)$ is the residual ionospheric error
- $\varepsilon_T(k)$ is the residual tropospheric error
- m_G is the multipaths error experienced by the ground station receivers
- b_G is the noise error experienced by the ground station receivers

The final error on the smoothed pseudorange after application of all corrections is:

$$\varepsilon_L{}^C = P_L{}^C(k) - \left[\rho(k) - c.\Delta t^u(k)\right]$$

$$\varepsilon_L{}^C = \varepsilon_I(k) + \varepsilon_T(k) + \varepsilon_{\Delta t^i}(k) + \widehat{m}_L(k) + \widehat{b}_L(k) + m_G(k) + b_G(k)$$

(6-30)

Finally, these errors $\varepsilon_L{}^C$ on each pseudorange measurement are projected on the three positing axes using the pseudoinverse matrix by the weighted least squares algorithm. The final errors must then be properly described and to do so we need to identify the statistics of these errors i.e. the sigmas as well as the time evolution of these errors.

We consider that the only error for which the evolution at each step of the receiver has the lowest level of uncertainty is the thermal noise $\widehat{b}_L$ and b_G. Indeed, we can model the thermal noise superimposed to the received signal as white noise at antenna output, so we can model the impact of noise on the pseudorange measurement using the equivalent DLL filter, and finally we can model the impact of the noise on the smoothed pseudoranges using the filter model.

However, for ionospheric ε_I, tropospheric ε_T errors, for multipath $\widehat{m}_L$ and m_G we have to identify an evolution law to be able to represent the final impact.

It can be assumed that I and T vary normally with a time constant much longer than the DLL equivalent filter time constant and smoothing filter time constant. Thus, $\widehat{I}_L$ and $\widehat{T}_L$ have very slow fluctuations compared to the DLL and smoothing filter time constant. However, ε_I and ε_T depend on the iono and tropo correction contained in the PRC, so the final evolution of ε_I and ε_T is not known to us. Maybe we can assume that ε_I and ε_T also behave like very slow quantities because the iono and tropo corrections contained in the PRC may be assumed to have very slow variations. So, they may be modeled as the output of a filter with a very large time constant. However, the exact model of the time evolution of these terms is not known.

The user clock bias $c.\Delta t^u$ is ignored because it has not effect on the projection of the pseudorange errors on the positioning axes.

The satellite clock correction error $\varepsilon_{\Delta t^i}$ may be assumed as a very slowly varying error. It may then be assumed as the output of a filter with a very large time constant too. However, the exact model of the time evolution of this term is not known.

The user and ground station multipaths errors $\widehat{m}_L$ and m_G are more difficult to model. Both errors result from the code-carrier smoothing of the raw multipath error affecting respectively the user and ground measurements. So they cannot have frequency components larger than the smoothing filter cut-off frequency which means that their time constant is larger than the time constant of the smoothing filter. Similarly the raw multipaths errors cannot have frequency components larger than the equivalent DLL filter bandwidth. However, it may not be correct to assume that the errors resulting from this dual filtering process are filtered white noise. Indeed these resulting errors may only have a few frequency components below the cut-off frequencies. It may be optimistic to model these errors as filtered white noise because these errors may rather have a few frequency components below the combined equivalent DLL and smoothing filter cut-off frequency.

However, we must not forget that these errors are the pseudorange errors and that they should be properly projected to obtain the position errors. This process is described in a further section.

To conclude we think that it may be interesting to further study this model so as to completely assess the impact of the receiver signal processing on the position errors. However, since the filter model presented in [MURPHY, 2005] has been validated using real data it can be considered as valid.

6.4.1.2   Scale factors determination

We present here the mathematical developments that we made to analyze the nominal GBAS NSE generator. We start with the natural relationship between measurement errors and position errors already presented in Chapter 4:

$$dX = \hat{X}(k) - X(k) = A \times E(k)$$

(6-31)

With:

- $X(k)$ the true aircraft position at epoch k

- $\hat{X}(k)$ the estimated aircraft position at epoch k.

- $E(k) = \begin{bmatrix} e^1(k) \\ \vdots \\ e^n(k) \end{bmatrix}$ the aggregate of the user pseudorange measurement errors including all user possible corrections, plus the GBAS Pseudorange Correction (PRC). It includes residual ionosphere error, residual troposphere error, airborne multipath, air noise, ground multipaths, and ground noise.

- $A = \left[H^t \Sigma^{-1} H\right]^{-1} H^t \Sigma^{-1}$ the weighted least squares position estimation projection matrix.

- H the observation matrix expressed in the local reference frame oriented toward the ideal aircraft trajectory (atrk, xtrk, vert).

- $\Sigma = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n^2 \end{bmatrix}$ where σ_i^2 is the variance of the residual pseudorange measurement error for satellite i. These variances contain all the information on pseudorange errors and that's why it is important to use adequate values.

Thus, the errors in the position domain are a linear combination of the pseudorange measurement errors. We denote:

$$A(k) = \begin{bmatrix} a_{atrk}^1(k) & \cdots & a_{atrk}^n(k) \\ a_{xtrk}^1(k) & \cdots & a_{xtrk}^n(k) \\ a_{vert}^1(k) & \cdots & a_{vert}^n(k) \\ a_b^1(k) & \cdots & a_b^n(k) \end{bmatrix}$$

(6-32)

We can derive the expressions of the position errors. We express here the vertical position error as:

$$dX_{vert} = a_{vert}^1 . e^1(k) + a_{vert}^2 . e^2(k) + \cdots + a_{vert}^n . e^n(k)$$

(6-33)

If we assume that the errors contained in $E(k)$ can be modeled as noise processes independent for each satellite then we can represent the position error as a single noise process with the following variance:

$$\sigma_{vert}^2 = (a_{vert}^1.\sigma_1)^2 + (a_{vert}^2.\sigma_2)^2 + \cdots + (a_{vert}^n.\sigma_n)^2$$

(6-34)

This can be written:

$$\sigma_{vert}^2 = \sum_{i=1}^{n}\left(a_{vert}^i.\sigma_i\right)^2$$

(6-35)

This leads to the following expression for the vertical position error:

$$dX_{vert} = \sigma_{vert}.du_{vert}$$

(6-36)

With du_{vert} the unit variance equivalent process to the vertical position error thus resulting from the characteristics of the receiver measurement errors and GBAS corrections.

This result shows that it is possible to generate directly the vertical positioning error instead of generating every pseudorange measurement errors so as to project it in the position domain. The only problem however lies with the stationarity of the characteristics of σ_{vert}^2. Indeed, the number n of satellites varies with time and the characteristics a_{vert}^i and σ_i also vary with time. One method to reflect this change is to compute statistics of the possible values of σ_{vert}^2, and run different simulations when a new value of σ_{vert}^2 is drawn each time. This however does not reflect the possible change of a_{vert}^i and σ_i during each single simulation.

The same expressions can be developed for horizontal NSE:

$$\sigma_{atrk}^2 = \sum_{i=1}^{n}\left(a_{atrk}^i.\sigma_i\right)^2$$

$$\sigma_{xtrk}^2 = \sum_{i=1}^{n}\left(a_{xtrk}^i.\sigma_i\right)^2$$

(6-37)

And

$$dX_{atrk} = \sigma_{atrk}.du_{atrk}$$

$$dX_{xtrk} = \sigma_{xtrk}.du_{xtrk}$$

(6-38)

Finally, we obtain:

$$dX = \Sigma_{\text{pos}}.\, dU$$

<div align="right">

(6-39)

</div>

Were:

- $\quad \Sigma_{\text{pos}} = \begin{pmatrix} \sigma_{vert} & 0 & 0 \\ 0 & \sigma_{atrk} & 0 \\ 0 & 0 & \sigma_{xtrk} \end{pmatrix}$

- $\quad dU = \begin{pmatrix} du_{vert} \\ du_{atrk} \\ du_{xtrk} \end{pmatrix}$

The processes $du_{vert}, du_{atrk}, du_{xtrk}$ are correlated in time individually and with each other.

This statistical model for the GBAS position error conforms to the model proposed in [MURPHY, 2005]. Then, our model and the state of the art model will be based on the same statistical base. We can deduce from our developments that the K factors used in the GBAS NSE generator correspond exactly to the standard deviations we have highlighted:

$$K_{vert} = \sigma_{vert} \quad K_{xtrk} = \sigma_{xtrk} \quad K_{atrk} = \sigma_{atrk}$$

<div align="right">

(6-40)

</div>

In fact, weighted least squares theory tells us that to compute these standard deviations we can use a more direct method. We know that the covariance matrix C of the position estimation error is given by:

$$C = (H^T \Sigma^{-1} H)^{-1}$$

<div align="right">

(6-41)

</div>

Where:

- H the observation matrix as defined in Chapter 4.
- Σ is the measurement error covariance matrix with $\Sigma = \text{cov}(E)$

Therefore, we have simply:

$$\begin{cases} \sigma_{atrk}{}^2 = C(1,1) \\ \sigma_{xtrk}{}^2 = C(2,2) \\ \sigma_{vert}{}^2 = C(3,3) \end{cases}$$

<div align="right">

(6-42)

</div>

Where $C(i,j)$ is the coefficient of matrix C in row i and column j.

We have to remark here that the scale factors are logically highly dependent on the pseudorange measurement error models. In our study we used the pseudorange measurement models presented in Chapter 4 which are up to date in comparison with the models used in the GBAS NSE generator and which are reminded in section 6.3.3.

6.4.2 Steps NSE determination

We present here the mathematical developments that we made to analyze the state of the art step function generator architecture. Satellite geometry simulations were run as in the nominal case to determine the distribution of the magnitude of the steps due to step events. To compute the magnitude of the error due to step events we will make the conservative assumption that the loss of a satellite induces a larger step than the inclusion of a new one as in [MURPHY, 2009].

In fact, it is necessary to define a step function since as we said previously, in the state of the art model, for the nominal NSE generator the K scale factors are only drawn once at the beginning of the simulation and kept constant. Thus, evolutions of the statistics of the GBAS NSE introduced by constellation changes are not represented by the nominal NSE generator.

If we assume no constellation change between epoch $k-1$ and k, the evolution of the error is given by:

$$dX^{nom} = dX(k) - dX(k-1)$$

(6-43)

Let's assume that the nth satellite out of n is lost between epoch $k-1$ and k, then we have:

$$dX^P = dX^{Loss}(k) - dX(k-1)$$

(6-44)

Where:

$dX^{Loss}(k) = A^{Loss}(k) \times E^{Loss}(k)$ is the position error considering the new constellation.

- $A^{Loss}(k)$ is the projection matrix for the new constellation
- $E^{Loss}(k)$ is the aggregate of the GBAS pseudorange measurement errors for the new constellation

Then, the additional error induced by the loss of a satellite is given by:

$$dX^{step} = dX^P - dX^{nom} = dX^{Loss}(k) - dX(k)$$

(6-45)

Using equations (6-43) and (6-44) we can write:

$$dX^{step} = A^{Loss}(k) \times E^{Loss}(k) - A(k) \times E(k)$$

(6-46)

We denote:

- $A_{1,n-1}$ the submatrix of A of dimension (3,n-1)
- A_n the sub-matrix of A of dimension (3,1)
- E_n the Nth component of E such that $E = \begin{bmatrix} E^{Loss} \\ E_n \end{bmatrix}$

We can write:

$$E^{step} = A^{Loss} \times E^{Loss} - \begin{bmatrix} A_{1,n-1} & A_N \end{bmatrix} . \begin{bmatrix} E^{Loss} \\ E_n \end{bmatrix}$$

<div align="right">(6-47)</div>

Which can be expressed as following:

$$E^{step} = \left(A^{Loss} - A_{1,n-1}\right)E^{Loss} - A_n E_n$$

<div align="right">(6-48)</div>

Finally, we can easily derive the covariance of E^{step} as:

$$\text{cov}(E^{step}) = \begin{bmatrix} A^{Loss} - A_{1,n-1} \end{bmatrix} . \Sigma^{loss} . \begin{bmatrix} A^{Loss} - A_{1,n-1} \end{bmatrix}^T + \begin{bmatrix} A_n A_n{}^T \end{bmatrix} \sigma_n^2$$

<div align="right">(6-49)</div>

With $\Sigma^{loss} = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_{n-1}^2 \end{bmatrix}$

We can derive the expression associated to the vertical step for example:

$$\left(\sigma_{vert}^{step}\right)^2 = \sum_{i=1}^{N-1} \left[(a_{vert,i}^{Loss} - a_{vert,i})^2 . \sigma_i^2(k) \right] - a_{vert,n}{}^2 . \sigma_n^2(k)$$

<div align="right">(6-50)</div>

This expression is compliant with the one presented for the state of the art GBAS NSE generator.

6.5 Numerical analysis of the legacy model

Now that the mathematical expressions necessary for the computation of the parameters of the model have been determined we present the results obtained using these during simulations. In this section, we only focus on GAST-C related simulations since our first objective is to evaluate the validity of the legacy GBAS NSE model which was initially developed for CAT-I GBAS autoland simulations.

6.5.1 Determination of nominal GBAS NSE statistics through simulations

6.5.1.1 Simulation assumptions

6.5.1.1.1 Constellation assumptions

We have to make a number of assumptions so as to assess the GBAS NSE characteristics. In fact, to determine the scale factors previously presented which are K_{vert}, K_{xtrk}, K_{atrk} it is necessary to simulate the GPS constellation and to collect sufficient data so as to derive distributions. To do so, the idea is to compute the covariance matrix associated to the weighted least squares estimation of position in several locations and for several constellations configurations.

We thus considered the standard 24 satellites GPS constellation and we took into account every possible constellation state with up to two different satellites removed from service. In fact, we could have considered more out of service satellites but we limited this parameter so as to reduce the computational burden. We can see in [RTCA, 2004] that for operational service availability simulations, only two out of service satellites are considered which legitimate our assumption.

Considering different constellation states made it necessary to consider the constellation state probabilities which are presented in Table 26 so as to combine the obtained results in a unique histogram. Two different sets of probabilities are presented in this table:

- The first one was used for the development of the legacy GBAS NSE model. This set of probabilities can be found in the first issue of LAAS MASPS [RTCA, 1998]. These probabilities have proven to be very conservative with regard to the observed states of the GPS constellation and have since then been updated so as to better reflect the reality.
- The second set corresponds to the updated probabilities which have been published in DO-245A by RTCA [RTCA, 2004].

It is important to understand that these standard probabilities are used to assess GPS availability considering the standard 24 satellite constellations as explained in [RTCA, 2004]. In fact, more than 24 satellites can be in service at any one time and the actual performance of the GPS constellation has to proven to be higher than this model.

As a first assumption, we have decided to compare the results obtained using each set of probabilities so as to assess the impact on the GBAS NSE statistics. However, DO-245A probabilities are considered as the reference in the rest of our work.

As we considered only two maximum satellite failures at the beginning of our simulations we have to modify the probabilities presented in Table 26 to obtain a probability of 1 when combining our different constellation configurations. We decided to overestimate the probability of 22 operational satellites since it appeared to be the more conservative choice to us:

$$P(22\ op.\,sat) = 1 - P(24\ op.) - P(23\ op.) = 0.02$$

(6-51)

Number of Operational Satellites N	Probability of N operational satellites used in [MURPHY, 2009]	Probability of N operational satellites DO-245A [RTCA, 2004]
24	0.72	0.95
23	0.17	0.03
22	0.064	0.012
21	0.026	0.0048
20	0.02	$0.03 \times 0.34^{(23-N)}$

Table 26: Constellation state probabilities

Now that we have determined the constellation state probabilities it is necessary to determine how it will be used. In fact, when considering n out of service satellites at the initiation of a simulation, it is necessary to assess and take into account every possible constellation configurations that is to say all possible combinations of (N-n) satellites.. For example, with a 24 satellite constellation, assuming one satellite failure will lead to 23 different constellation configurations resulting in 23 different computations of σ_{vert}. The results obtained for each constellation state have then to be combined so as to obtain a single result. Let's denote X_{vert} the random variable of the positioning error in the vertical direction. Then we have:

$$\sigma_{vert}{}^2 = E\left[X_{vert}{}^2\right] - E[X_{vert}]^2$$

(6-52)

In the case of nominal GBAS NSE, we assume that measurement errors are not biased which leads to $E[X_{vert}] = 0$. Thus we obtain:

$$\sigma_{vert}{}^2 = E\left[X_{vert}{}^2\right]$$

The sigma value obtained is the standard deviation of a random variable depending of different other random variables: noise, troposphere and ionosphere error, multipaths for example as well as projection matrix related variables such as user position and constellation state (number of satellites, health, and position of each satellite...). We can denote:

- N_F the random variable representing the number of satellite failures
- I_d the random variable representing the ids of the faulted satellites

With these assumptions we can develop the previous expression using conditional expectations:

$$\sigma_{vert}{}^2 = E\left[X_{vert}{}^2\right] = E_{N_F}\left[E\left[X_{vert}{}^2 | N_F = n\right]\right]$$

(6-53)

We can apply the law of total expectations which gives:

$$\sigma_{vert}{}^2 = E_{N_F}\left[E\left[X_{vert}{}^2 | N_F = n\right]\right] = \sum_{n \in N_F} E\left[X_{vert}{}^2 | N_F = n\right] P[N_F = n]$$

(6-54)

The probabilities $P[N_F = n]$ correspond to the constellation state probabilities discussed previously and presented in Table 26.

We can further develop this expression by introducing the second variable:

$$E\left[X_{vert}{}^2 | N_F = n\right] = E_{I_d}\left[E\left[X_{vert}{}^2 | I_d =, N_F = n\right]\right]$$

(6-55)

If we apply again the law of total expectations we obtain:

$$E\left[X_{vert}{}^2 | N_F = n\right] = \sum_{i \in I} E\left[X_{vert}{}^2 | I_d = i, N_F = n\right] P[I_d = i | N_F = n]$$

(6-56)

We can develop this expression in with our assumptions: we assumed that only two satellites failures can occur at a given epoch. Thus the variable N_F can take three different values which are 0, 1and 2:

$$N_F = \{0,1,2\}$$

The values of the variable I depend on the number of failed satellites. If there is no failed satellite, this variable is not relevant and if there is one failed satellite then it represents the ids of the combinations of one satellite out of the complete constellation of N satellites.

Thus, if we apply the last equation we have two possibilities

- If we assume no satellite failure $N_F = 0$ and I={Ø}

$$E\left[X_{vert}{}^2|N_F = 0\right]$$

This value has not been developed further.

- If we assume one satellite failure then: I={1, ... ,24}

$$E\left[X_{vert}{}^2|N_F = 1\right] = \sum_{i=1}^{N} E\left[X_{vert}{}^2|I_d = i, N_F = 1\right]P[I_d = i|N_F = 1]$$

(6-57)

In this study we assumed that each satellite of the considered constellation had the same probability to fail. Thus we assumed:

$$P[I_d = i|N_F = 1] = C_N^1 = \frac{1}{N}$$

(6-58)

Thus, we obtain:

$$E\left[X_{vert}{}^2|N_F = 1\right] = \frac{1}{N}\sum_{i=1}^{N} E\left[X_{vert}{}^2|I_d = i, N_F = 1\right]$$

(6-59)

- If we assume two satellite failures then I is the group of all possible combinations of two satellites out of N. There are C_N^2 possible combinations which are considered equiprobable thus we have:

$$P[I_d = \{i,j\}|N_F = 1] = C_N^2 = \frac{1}{N(N-1)}$$

(6-60)

$$E\left[X_{vert}{}^2|N_F = 2\right] = \frac{1}{N(N-1)}\sum_{i=1}^{N} E\left[X_{vert}{}^2|I_d = \{i,j\}, N_F = 2\right]$$

(6-61)

Finally, we can use equation (6-54) to compute the final $\sigma_{vert}{}^2$:

$$\sigma_{vert}{}^2 = \sum_{n=0}^{2} E\left[X_{vert}{}^2|N_F = n\right]P[N_F = n]$$

(6-62)

Thus,

$$\sigma_{vert}{}^2 = E\left[X_{vert}{}^2|N_F = 0\right]P[N_F = 0] + E\left[X_{vert}{}^2|N_F = 1\right]P[N_F = 1]$$
$$+ E\left[X_{vert}{}^2|N_F = 2\right]P[N_F = 2]$$

(6-63)

And finally,

$$\sigma_{vert}{}^2 = E\left[X_{vert}{}^2 | N_F = 0\right]P[N_F = 0] + \frac{1}{N}P[N_F = 1]\sum_{i=1}^{N}E\left[X_{vert}{}^2 | I_d = i, N_F = 1\right]$$

$$+ \frac{1}{N(N-1)}P[N_F = 2]\sum_{\{i,j\}\epsilon I_d}E\left[X_{vert}{}^2 | I_d = \{i,j\}, N_F = 2\right]$$

(6-64)

6.5.1.1.2 Pseudorange measurement error model assumptions

We have already presented the models used to derive the legacy GBAS NSE model. As previously stated GBAS pseudorange measurement error models have since then been refined and that's why we use them in our simulations. These models have been presented in chapter 4 of this report. It is important to keep in mind that some of these pseudorange measurement error models ($\sigma_{pr\_gnd}^2$, $\sigma_{pr\_air}^2$, and σ_{mp}^2) are known as over bounding terms, and therefore by using them we may slightly overestimate the real GBAS pseudorange measurement errors. It would have been more appropriate for deriving the nominal NSE behaviour to use nominal models. However, there is no easy way to obtain these nominal models since we are not aware of the existence of such models.

We can remind here the expression of the standard deviation of the pseudorange measurement error of GBAS L1 C/A receivers [RTCA, 2004]:

$$\sigma_i^2 = \sigma_{pr\_gnd,i}^2 + \sigma_{pr\_air,i}^2 + \sigma_{mp,i}^2 + \sigma_{tropo,i}^2 + \sigma_{iono,i}^2$$

(6-65)

Where:

- $\sigma_{pr\_gnd,i}^2$ is the total (post correction) fault-free noise term provided by the ground function (via the VDB) for satellite i

- $\sigma_{pr\_air,i}^2$ is the standard deviation of the aircraft contribution to the corrected pseudorange error for the ith satellite. It includes the receiver and airframe multipaths contribution.

- $\sigma_{tropo,i}^2$ is the standard deviation of the residual tropospheric error for satellite i.

- $\sigma_{iono,i}^2$ is the standard deviation of the residual ionospheric delay (due to spatial decorrelation) uncertainty for satellite i.

As we have seen, the quality of the GBAS SIS and GBAS airborne equipment can be categorized in terms of Ground Accuracy Designator (GAD) and Airborne Accuracy Designator (AAD).

In the case of nominal GBAS NSE simulations we used the same assumptions as in the legacy model that is to say GAD B3 and AAD A [MURPHY, 2005]. Readers may refer to Chapter 3 for detailed expressions.

6.5.1.1.3 Other assumptions

Computations were made each second over the ground track repeating period of the GPS constellation which is 23 hours 56 minutes and 4 seconds and were repeated for different locations of the GPS receiver. For the locations of the receiver we used the same airports as in [MURPHY, 2005] but we also added new airports so as to extend the results to world coverage and not only for North America. We also added what we called high latitudes airports so to illustrate the evolution of the GBAS nominal NSE for latitudes above 70°. The different locations used are given in Table 27. As we will

see in further sections, high latitudes results are not used to derive the model. All other airports results are combined to obtain the final results using a simple mean.

We considered a standard mask angle of 5° for satellites visibility determination.

We computed the GBAS NSE scale factors according to the expressions presented in section 6.4.1. However, an additional mechanism was used. In reality, GBAS allows monitoring GPS signals. The user receiver is able to compute protection levels according to data transmitted by the ground. In our simulations we decided to reject the values that would lead to the non availability of GBAS service that is to say values that would make the Protection Levels exceed the associated Alert Limits. More precisely we only considered a vertical constraint since this is the most stringent requirement in the case of CAT-I and CAT-II/III approaches. We considered a vertical alert limit of 10 meters. It must be noticed that in the new GBAS CAT III concept currently developed by ICAO, the VAL used by the aircraft may be lower than 10 meters according to the geometry screening process and associated thresholds, leading to exclude more values of $\sigma_{vert}{}^2$.

Location	Latitude (°)	Longitude (°)	Height (m)
Anchorage	64.174361	-149.996361	46.33
Dallas	32.896828	-97.037996	185.01
New York City	40.639751	-73.778926	3.96
Los Angeles	33.942522	-118.407161	38.10
Miami	25.793250	-80.290556	2.44
Chicago	41.978143	-87.905870	204.83
Seattle	47.4498889	-122.311778	132.00
Johannesburg	26.148175	28.134939	20.00
Dakar	14.738436	-17.488747	20.00
Bruxelles	50.901702	4.483025	20.00
Hong Kong	22.316478	113.936553	20.00
Sydney	-33.933078	151.177550	20.00
Tokyo	35.769655	140.389686	20.00
Beijing	39.960555	116.256944	20.00
Lat 0 airport	0.000000	0.0	20.00
Lat 70 airport	70.0	0.0	20.00
Lat 75 airport	75.0	0.0	20.00
Lat 80 airport	80.0	0.0	20.00
Lat 85 airport	85.0	0.0	20.00

Table 27: Airport locations

6.5.1.2 *Vertical GBAS NSE magnitude*
On the basis of our simulations we computed several histograms and CDF (Cumulative Density Functions) so as to analyze the results. The goal was to assess the validity of the legacy model and if necessary to propose updates so as to reflect current evolutions of the standards and systems. In the following sections we focus on airport locations below 70° of latitude. The high latitude locations are treated separately since the associated behaviour is quite different.

6.5.1.2.1 **Comparison with legacy model**
So as to compare our results with the state of the art model, we initially used the same constellation probabilities as in [MURPHY, 2005], that is to say the first column of Table 26.
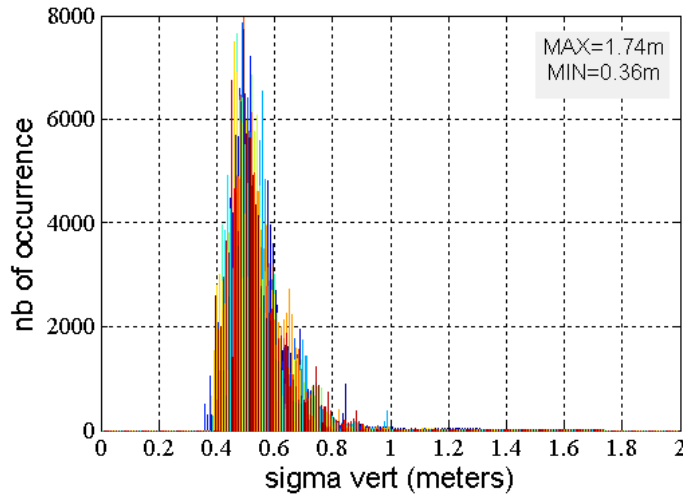
Figure 62: observed σ_{vert} histogram superimposed for all airports for GAST-C and GAD B3 using same probabilities as in [MURPHY, 2009]

The obtained results are presented in Figure 62 and Figure 63. We compare the obtained cumulative density functions obtained for each simulated location to the function used in the legacy model to generate the K_{vert} scale factor. The CDF are presented with inversed axes so as to ease the comparison with the legacy model. The equivalent figure for this legacy model is shown in Figure 64.
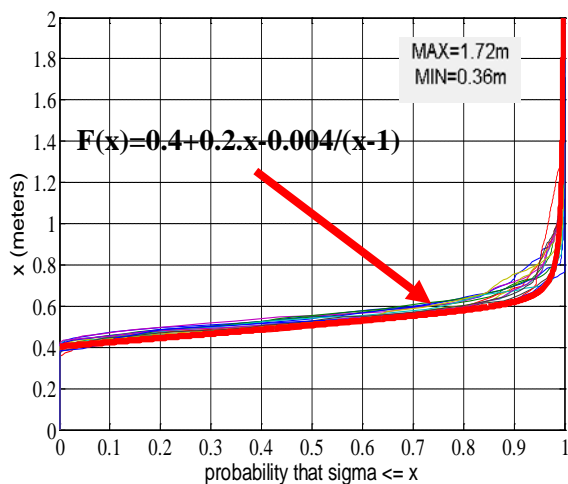
Figure 63: Comparison between observed σ_{vert} CDF for all simulated airports for GAST-C and GAD B3 and Kvert generation function from [MURPHY, 2005]
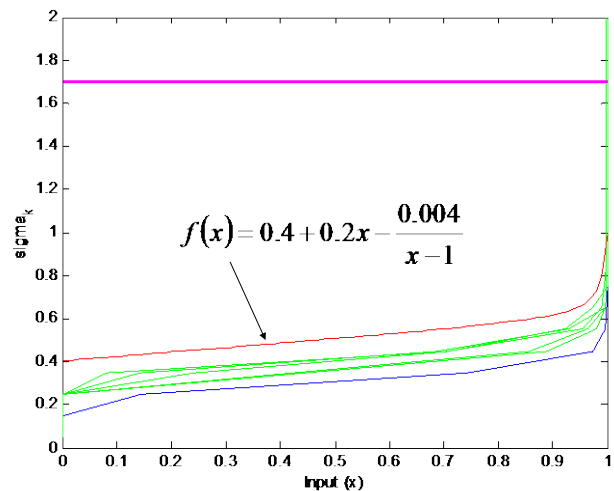
Figure 64: Kvert generation function from [MURPHY, 2005]

As we can see here, we obtained slightly higher results since our results exceed the red function. This function f was supposed to be larger than the results obtained with the combination of a GAD B3 ground station and an AAD A airborne receiver [MURPHY, 2009]. This difference can be explained by the difference in the pseudorange measurement error models used: first, we use here a more appropriate model for multipath which we know is larger than the initial one and second, the DO-245A models are bounds to the observed pseudorange measurement errors. We can assume therefore that they are conservative. However, on the contrary the functions proposed in the legacy model may be too optimistic due to inappropriate error models.

6.5.1.2.2 Impact of constellation state probabilities

The next step was to compare our results when using the two different sets of constellation state probabilities gathered in Table 26. This can be done by comparing the CDF of Figure 63 and Figure 66 or the histograms of Figure 62 and Figure 65. Only small differences can be noticed. We can remark that high values of σ_{vert} are more rarely observed in the case of DO-245A probabilities. This is easily understandable since in this case we increase the contribution of full constellation operational results which logically results in lower errors than with degraded constellation states. To conclude, the difference brought by the new set of probabilities is not obvious. However, since it is more realistic, these probabilities are used as reference for the rest of our simulations.
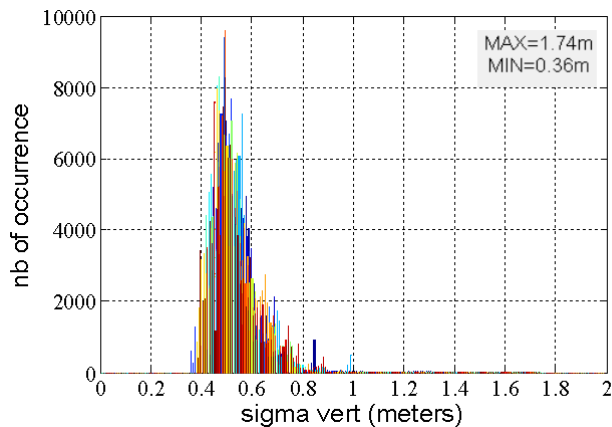
Figure 65: observed σ_{vert} histograms superimposed for all airports for GAST-C and GAD B3 using DO-245A probabilities
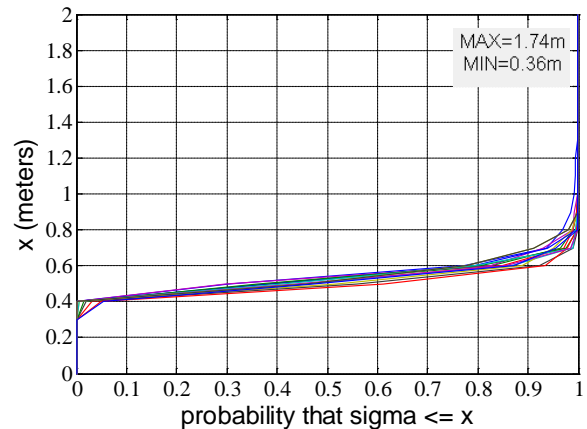
Figure 66: observed σ_{vert} CDF for all airports for GAST-C and GAD B3 using DO-245A probabilities

6.5.1.2.3 High latitude airports

During our simulations we noticed that for high latitudes, results are quite different from what we see for other locations. Thus we decided to dedicate a section to these latitudes so as to illustrate the evolution of the vertical NSE magnitude with respect to the latitude. The associated observed histograms and CDF are represented in Figure 67 and Figure 68. We observe that the accuracy is degrading when getting closer to the pole due to bad geometries. In these particular cases, the initial model can no longer be applied and thus it may be interesting for specific airports to investigate this problem. This study was not in the initial scope of our study that's why we do not give further details here but this could be studied in future studies.
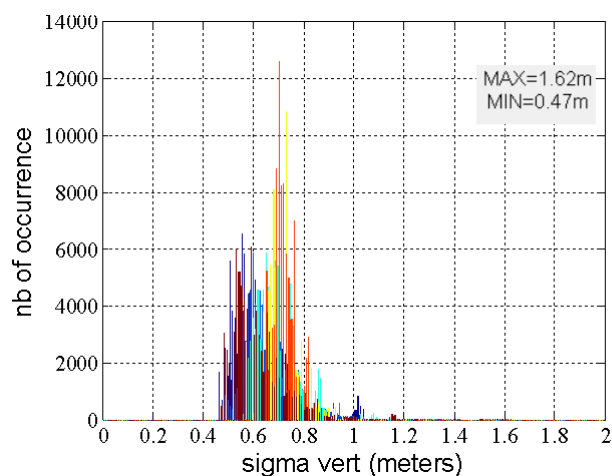
Figure 67: observed σ_{vert} histograms superimposed for high latitudes airports for GAST-C and GAD B3
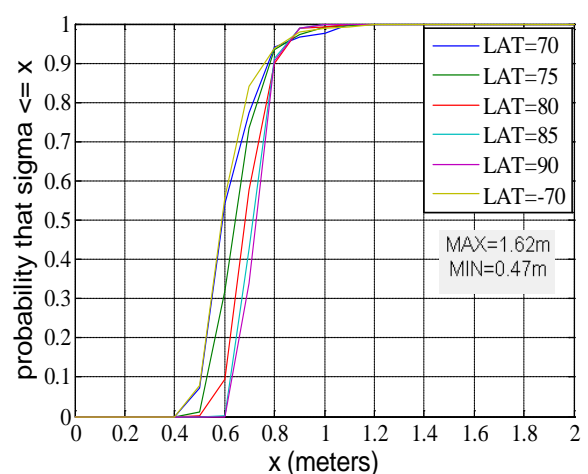
Figure 68: observed σ_{vert} CDF superimposed for high latitudes airports for GAST-C and GAD B3

6.5.1.2.4   Conclusions

We finally derived the distribution of $\boldsymbol{K_{vert}}$ which corresponds to the combination of the results obtained for every considered airport. It is shown in Figure 69. The final comparison between the simulated results and the function used in the legacy model can be seen in Figure 70. This last figure confirms that our results are slightly higher than the function f used in the legacy model. We conclude by proposing to generate K_{vert} scale factor by directly drawing it from our observed distributions so as to take advantage of the updated models used during our study.
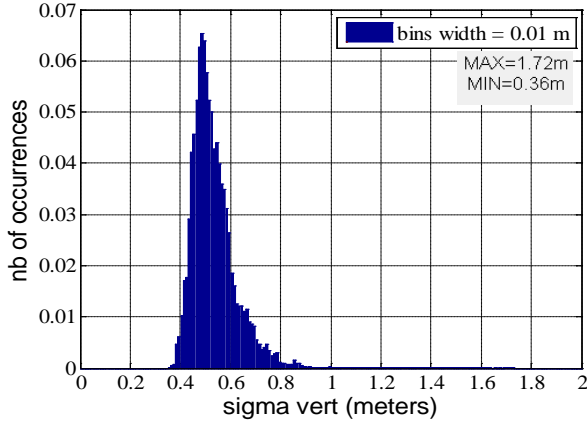
Figure 69: observed σ_{vert} histogram s for GAST-C and GAD B3
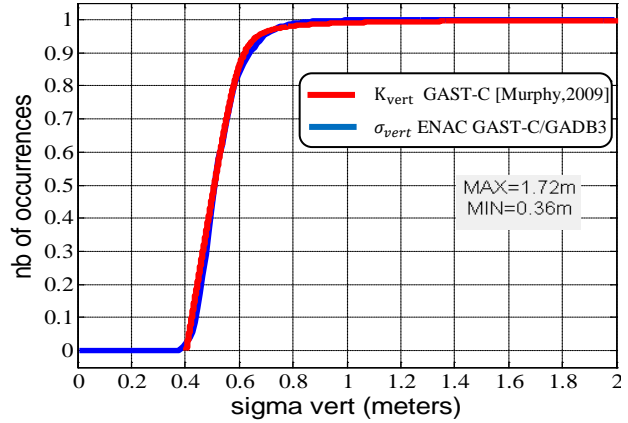
Figure 70: Comparison between observed σ_{vert} CDF for GAST-C and GAD B3 and Kvert generation function from [MURPHY, 2005]

6.5.1.3   *Horizontal GBAS NSE magnitude*

The horizontal GBAS NSE magnitude is computed under the exact same assumptions as previously. However, it is necessary to make an additional assumption since the horizontal components along track and cross track are related to the orientation of the local reference frame.

6.5.1.3.1   Worst horizontal sigma

Our approach is to compute what we have called the worst horizontal sigma that is to say that at each epoch of our simulations we determined the orientation that would produce the largest NSE sigma along the cross track axis. The idea is that in the final model we would randomly pick an angular deviation, draw the worst horizontal sigma σ_{whor} in its distribution, and project it using the random angle.

The position error in the horizontal plane (North-East components) is a random variable. If we consider the orientation producing the largest NSE sigma along the cross track axis, the coordinates of the position affected by the position error can be described by two Gaussian random variables which we denote here x_{min} and x_{whor} and which are independent in the a/c coordinate frame. The a/c coordinate frame results from the rotation of the local coordinate frame North-East-Down of an angle θ corresponding to the current heading of the a/c. We can thus express the coordinates of the position error in the local coordinate frame as:

$$x_{north} = [x_{min} \times \cos(\theta) - x_{whor} \times \sin(\theta)]$$

$$x_{east} = [x_{min} \times \sin(\theta) + x_{whor} \times \cos(\theta)]$$

(6-66)

Thus, what we propose is to generate the errors coordinate x_{min} and x_{whor}. Since it would be far too much conservative to always consider the worst error case we can consider that the a/c is currently

flying with a heading different from the heading producing the largest NSE sigma along its cross-track axis. The idea is here to randomly draw an angular deviation representing the rotation angle between the worst horizontal sigma coordinate frame and the current a/c coordinate frame φ.

Finally the position error could be obtained by:

$$x_{atrk} = [x_{min} \times \cos(\varphi) - x_{whor} \times \sin(\varphi)]$$

$$x_{xtrk} = [x_{min} \times \sin(\varphi) + x_{whor} \times \cos(\varphi)]$$

(6-67)

Our aim is thus to determine the major and minor axes of the ellipsoid defined by the variables x_{atrk} and x_{xtrk} depending on φ and which correspond to x_{whor} and x_{min}. This can be done by diagonalizing the covariance matrix of the position error C which expression has been reminded in equation (6-41). For clarity, we will focus on the horizontal part of the matrix C that we call C_{hor}.

$$C_{hor} = \begin{bmatrix} C_x(1,1) & C_x(1,2) \\ C_x(2,1) & C_x(2,2) \end{bmatrix}$$

(6-68)

We then computed the eigen vectors and eigen values of this matrix:

$$\text{Eigenvectors: } V_1 = \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix} \text{ and } V_2 = \begin{bmatrix} -\sin(\theta) \\ \cos(\theta) \end{bmatrix}$$

(6-69)

$$R = [V_1 \quad V_2]$$

(6-70)

$$\text{Eigenvalues: } \lambda_{maj}, \lambda_{min}$$

With θ the rotation angle between the north, east, down reference frame (which was used as the basis coordinate frame) and the obtained reference frame. The eigenvalues exactly correspond to minor and major axes of the ellipse and using the eigenvectors we were able to determine the orientation producing these maximum errors.

6.5.1.3.2 Horizontal GBAS NSE magnitude results

The results obtained for the horizontal GBAS NSE magnitudes are presented in Figure 71 and Figure 72. These figures do not raise many comments. What we can say is that the horizontal errors are really lower than the vertical one. Moreover, σ_{min} is always lower than 0.3 meters which is not really significant. We can see here that in terms of accuracy the constraint is really driven by the vertical component of the error.
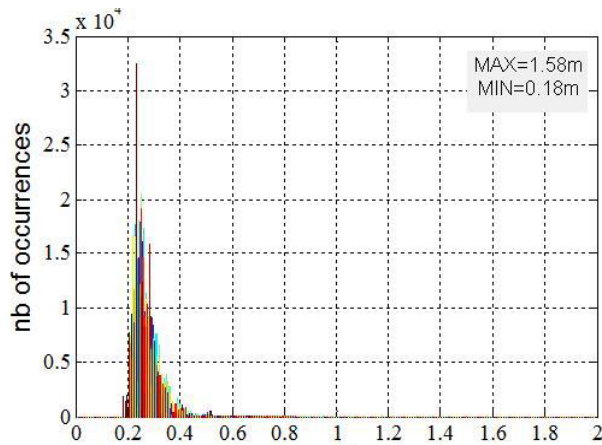
Figure 71: σ_{whor} **CDF superimposed for all airports for GAST-C and GAD B3**
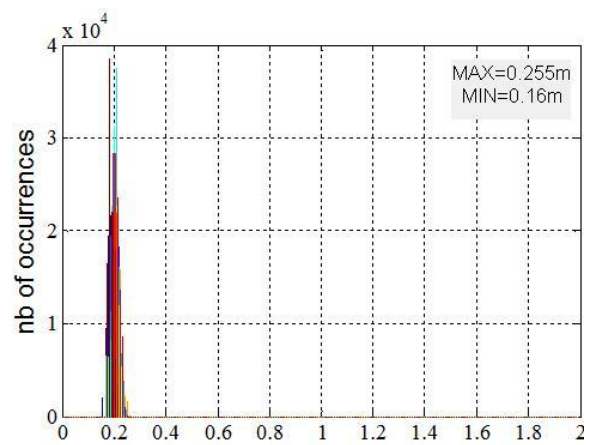
Figure 72: σ_{min} **CDF superimposed for all airports for GAST-C and GAD B3**

6.5.1.3.3 Ratio between σ_{whor} and σ_{vert}

As we explained previously in the legacy model the horizontal scale factors are directly derived from the vertical scale factor using a constant ratio of 0.818. We computed in our simulations the ratio α between σ_{vert} and σ_{whor}. Results are gathered in Figure 73 and Figure 74. The ratio chosen in the legacy model is represented in red.
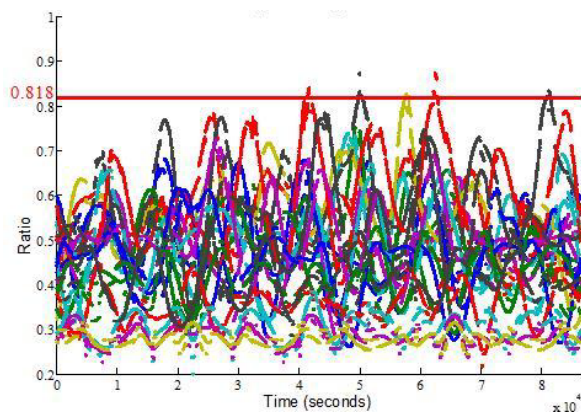
Figure 73: evolution of $\alpha = \sigma_{whor}/\sigma_{vert}$ **superimposed for all airports for 24 hours simulation**
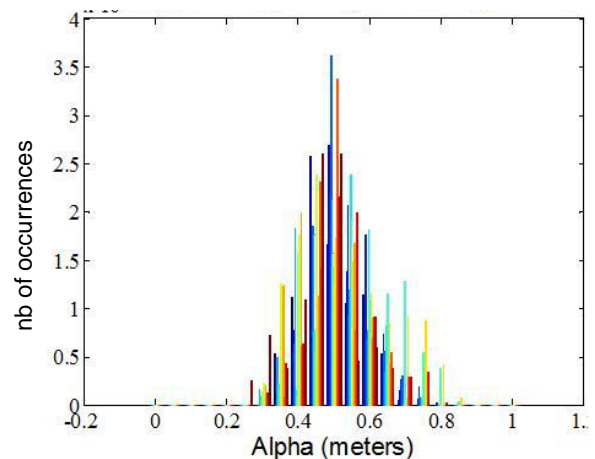
Figure 74: histogram of $\alpha = \sigma_{whor}/\sigma_{vert}$ **superimposed for all airports**

According to the results we obtained in simulation the probability that the ratio α exceeds the threshold 0.818 is equal to about 4.10^{-4}. It has to be compared with the probability of 2.10^{-5} reminded in Table 24 and which was published for the legacy model in [MURPHY, 2009]

This difference can be explained by two facts. First, we added new locations to our simulations which may have introduced new geometries. Secondly we used different pseudorange measurement error models. However, the ratio proposed in [MURPHY, 2005] can still be considered as representative since it is used in the frame of nominal NSE generation. Still, we propose a different mechanism to generate the horizontal NSE magnitudes in the conclusions which implies to generate independently the horizontal and vertical scale factors.

6.5.1.3.4 Correlation between horizontal and vertical NSE

We already explained that tests have been run to check if it was necessary to implement an additional correlation algorithm to account for satellites geometry correlation with time. The study published in

[MURPHY, 2009] concluded that it was not necessary but since we obtained slightly different results we want to check if this hypothesis is still valid in our case.

The vertical and horizontal position estimation errors at one specific epoch originate from the pseudorange measurement errors at the same epoch since we assume here that the position estimation technique is the snapshot LSR algorithm. Thus, a geometrical correlation is introduced between vertical and horizontal position errors at the same epoch and our model does not reproduce this feature.

To check the necessity to reproduce this instantaneous geometrical correlation, we first compute the correlation coefficient between vertical and horizontal errors. In fact we already did it since we computed the covariance matrix C which gives the cross-correlation between vertical and worst horizontal errors. Thus, the correlation coefficient is given by:

$$\rho_{vert/xtrk} = \frac{\sigma_{vert/xtrk}}{\sigma_{vert} \cdot \sigma_{xtrk}}$$

(6-71)

We obtained the distribution presented in Figure 75.
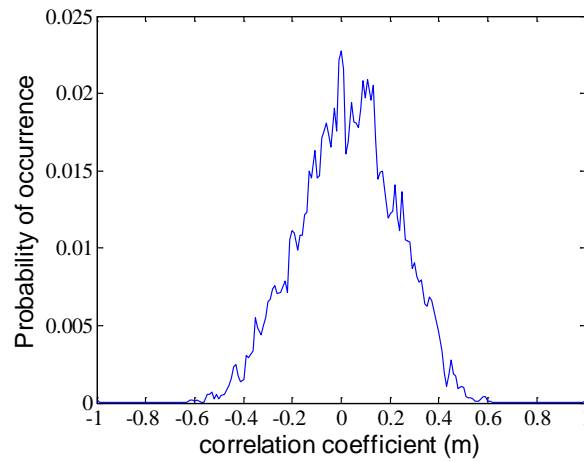
Figure 75: Correlation coefficient between vertical and worst horizontal errors

We then made the following experiment. We repeated 2000 times the generation of two sequences of 360 seconds of white noise samples. Each sequence was then filtered by the 2nd order filter introduced in section 6.3.1.1. We then computed the correlation coefficient between the two filtered sequences. This was the first part of the test. The second one was to apply a correlation algorithm so as to introduce additional correlation driven by the distribution presented in Figure 75. The correlation algorithm is given by:

$$\begin{bmatrix} N_1 \\ N_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \rho & \sqrt{1-\rho^2} \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix}$$

(6-72)

Where:

- S_1 and S_2 are the filtered sequences
- N_1 and N_2 are the filtered and correlated sequences

173

- ρ is the desired correlation coefficient

The results obtained can be seen in Figure 76 and 22. What we can observe is that the correlation algorithm has no major impact on the correlation between the two sequences. Obviously, the 2nd order filter introduces a correlation which is not altered by the correlation algorithm. Thus, we agree with the conclusions presented in [MURPHY, 2009]: according to our results it appears that it is not necessary to introduce an additional correlation algorithm in the nominal NSE generator to account for the natural geometric correlation existing between the different GBAS NSE error components.
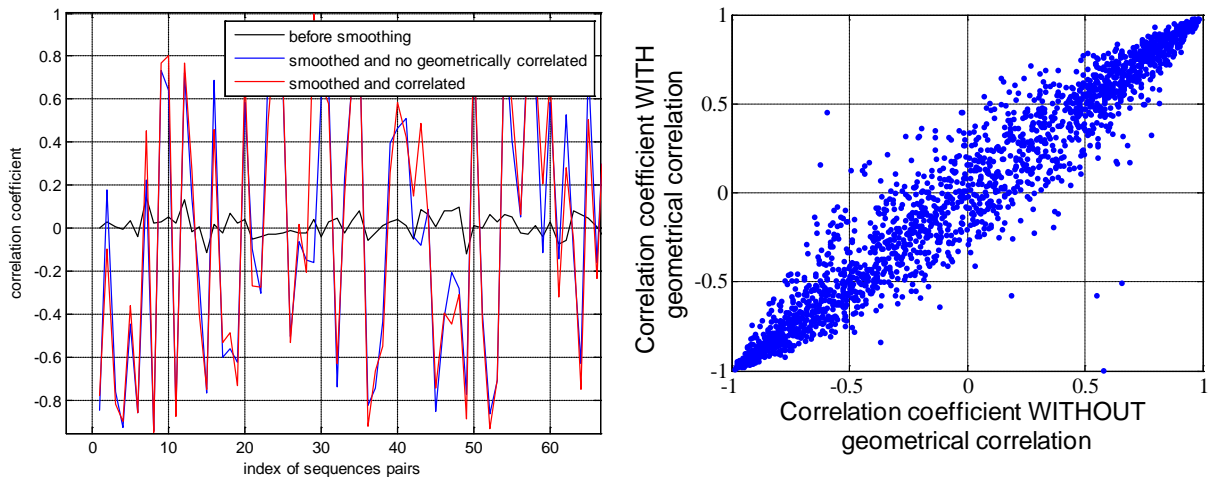
Figure 76 and 77: Comparison of the correlation coefficient of two sequences before and after geometrical correlation

6.5.1.3.5 High latitude simulations

In the case of high latitude simulations we observed the results shown in Figure 78 and Figure 79 for the σ_{whor}. It seems that the distribution of σ_{whor} is not really impacted by the latitude variations. This may be due to still good geometries of the constellation for the horizontal positioning.
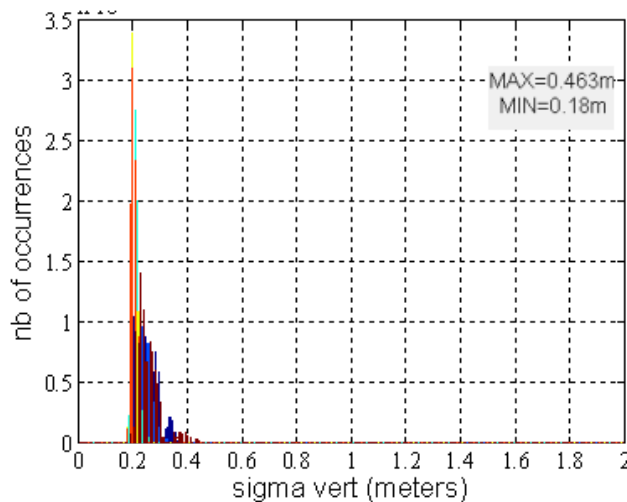
Figure 78: σ_{whor} histogram for high latitudes superimposed for GAST-C and GAS B3
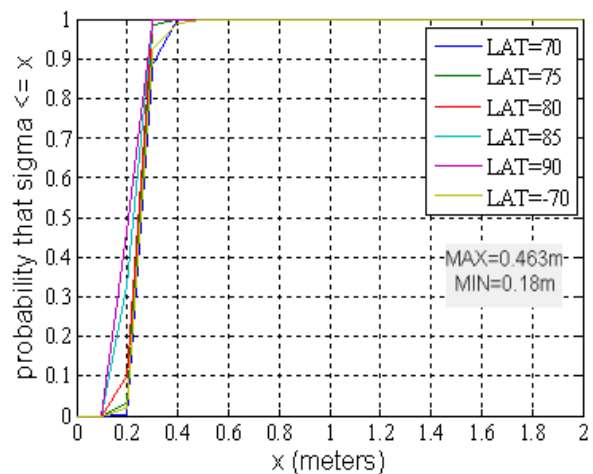
Figure 79: σ_{whor} CDF for high latitudes superimposed for GAST-C and GAS B3

6.5.1.3.6 Conclusions

We finally derived the distribution and CDF of the worst horizontal sigma of the GBAS NSE observed during our simulations which are represented in Figure 80 and Figure 81. These figures represent the results concerning horizontal NSE generation and obtained when combining the samples of all the

simulations concerning GAST-C and ground station GAD B3. Figure 81 highlights the fact that we found a curve producing σ_{whor} values which are smaller than the curve proposed in [MURPHY, 2005] for the generation of K_{xtrk} scale factors.
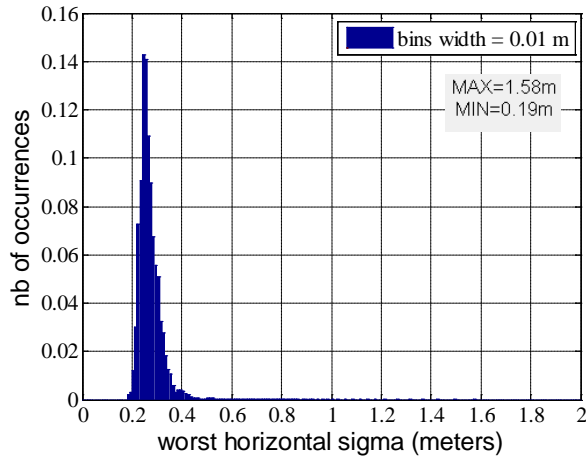
Figure 80: Observed σ_{whor} histogram for GAST-C and GAD-B3
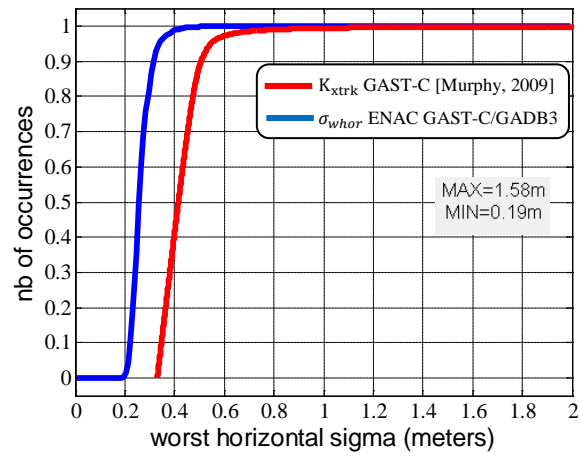
Figure 81: Comparison between observed σ_{whor} CDF for GAST-C and GAD B3 and K_{xtrk} generation function from [MURPHY, 2005]

As seen previously, in the legacy model a worst case constant scale factor has been used to derive the horizontal scale factors K_{xtrk} and K_{atrk} from the vertical scale factor K_{vert}. We think that this approach is very conservative for a model which is supposed to represent the nominal behaviour of the GBAS NSE. This is confirmed when we compare our results with the function corresponding to the generation of horizontal scale factors in Figure 81.

Different methods could be used. The first one would be to use a different constant ratio between horizontal and vertical components. However, we assume as a first proposal that it may be interesting to better represent the diversity of possible ratios between the vertical and horizontal components and thus to not consider a constant ratio. To do so, we propose to directly draw a scale factor called K_{whor} from the observed distribution of σ_{whor} and then to project this scale factor on the along track and cross track axes depending on a random angular deviation. Another advantage of this method is that instead of always generating the worst horizontal sigma, we project the worst case on an axis and thus we take only a portion of it.

We believe that - even if it is conservative –it is a good assumption to consider that the cross track components and along track components are the same and thus generate a circular error instead of ellipsoidal. In particular it allows reducing the computational burden. Therefore, cross track and along track components are considered to be equal to the maximum between the two possible projections of the worst horizontal sigma. We can remind the previously presented equations:

$$\sigma_{atrk} = [\sigma_{min} \times \cos(\varphi) - \sigma_{whor} \times \sin(\varphi)]$$

$$\sigma_{xtrk} = [\sigma_{min} \times \sin(\varphi) + \sigma_{whor} \times \cos(\varphi)]$$

If we consider that $\sigma_{min} = \sigma_{whor}$ we obtain:

$$\sigma_{atrk} = [\cos(\varphi) - \sin(\varphi)] \times \sigma_{whor}$$

$$\sigma_{xtrk} = [\cos(\varphi) + \sin(\varphi)] \times \sigma_{whor}$$

(6-73)

The drawback of this method is that it implicitly means that in the final model, the vertical and horizontal NSE scale factors will be independent since we draw their values independently into two different distributions. In reality, we have shown previously thanks to equation (6-35), that the NSE scale factors correspond to the standard deviation of the vertical and horizontal GBAS NSE and that they are both a combination of the standard deviation of the pseudorange measurement error standard deviations weighted by geometric factors which implies that they are in reality correlated.

Moreover, we may observe too often large ratios α between K_{xtrk} and K_{vert} for example. To evaluate this risk we computed the probability that the ratio α exceed the worst case ratio of 0.818 using our method. This was evaluated to be around 10^{-2} which is not satisfactory as we have seen previously. Therefore, we propose an additional limiting mechanism which would limit the ratio α to the worst case value of 0.818.

The method proposed here has been discussed with an autoland simulations specialist and answers to the needs expressed at the time of the study. Depending on the applications, this model may be simplified or a different method may be of course chosen.

Typically, if the objective was to study the worst case GBAS NSE we would certainly have made alternate assumptions.

6.5.2 Determination of GBAS NSE steps statistics through simulations

6.5.2.1 Simulation assumptions

The same assumptions as in the nominal case were used to assess the impact of constellation changes on the GBAS NSE. We considered the same exposure intervals as in [MURPHY, 2009] which are 15 seconds and 150 seconds. The additional hypotheses which are necessary consist in defining the nature of the step events and their associated probabilities of occurrence. These have already been presented in [MURPHY, 2009].

Nominal steps:

- Single rise or set: This event is conservatively estimated by removing the lowest visible satellite from the position solution. This assumption is very conservative since the satellite removed may often be far from crossing the mask angle. However, if we considered only satellites which would actually set, we would not dispose of enough data to build a pertinent histogram.
- Dual rise or set: This event is conservatively estimated by removing the two lowest visible satellites. For the same reasons, it is a very conservative assumption.

Fault steps:

- Single signal loss: This event is conservatively estimated by removing from the position solution the satellite inducing the largest σ_{vert_step}. Therefore our results can be considered as the worst case.
- Dual Signal Loss: This event is estimated by removing the satellite found for the previous event, and by removing the satellite producing the largest variance of step magnitude from the resulting subset of satellites.

To combine the rise/set results on one hand and the signal loss results on the other hand we had to determine the probabilities of occurrence of the corresponding events.

In the case of Signal Loss events we used the same probabilities as in [MURPHY, 2009] which were computed using the assumption of three satellites failures per year [GPS SPS, 2008]. Please refer to section 6.3.2.1 for detailed computation.

In the case of rise/set events we decided to check the correctness of the probabilities presented in Table 25. So, we considered a grid of location points which is defined in [RTCA, 2006] for SBAS availability tests. It covers latitude from 0 to 90 degrees with a sampling step of 3 degrees and longitude from 0 to 360 with a sampling step of 1 degree. The advantage of this grid is that the spacing between the longitudes of the grid points is dependent on the latitude of these grid points. To compute the rise/set events probabilities we determined the satellites seen from the receiver locations and we recorded every rising or setting event that would be seen by a real receiver. Two different windows of respectively 15 seconds and 150 seconds corresponding to the exposure intervals were shifted of 15 seconds over the GPS ground track repeat period of 23 h 56 min and 4 s or 86164 s. Thus, using the 15 seconds shift we obtained 86164/15 = 5744,26 so we defined 5744 windows. We also decided to compute two different types of probabilities:

- Worst-case: it consists in recording only the data for the grid location experiencing the worst number of rise/set events. This method is the same as in [MURPHY, 2009].
- Mean: it consists in computing the mean of all data from each grid point.

Our results are presented in Table 28. As you can see, our worst case results are compliant with the probabilities used in the state of the art model. The differences can be explained by the fact that we did not use exactly the same grid. The mean results are quite different. Thus, two approaches can be chosen: the worst-case or what we call "nominal" behaviour. For us, it is more interesting to consider the "nominal" scenario, since in Autoland simulations, extreme values of errors corresponding to the worst cases should be tested in the Limit case and Fault test simulations. Thus, we decided in our simulations to use two different sets of probabilities presented in Table 28:

- [Murphy et al., 2009] probabilities
- ENAC Mean probabilities

The objective here is to be able to compare the results obtained with the two sets of probabilities but also to compare our results with the curves proposed for the legacy model exposed in [MURPHY, 2005]. However, we propose in our final GBAS NSE generation model to consider the Mean probabilities as the reference since our aim here is to properly assess the nominal behaviour of the GBAS and not a worst case. Thus, if not mentioned, the set of probabilities used to derive our results is the set of ENAC Mean probabilities.

Finally, our approach is slightly different from the state of the art model since we decided to distinguish Rise/set and Signal Loss events since we consider that they do not belong to the same category of NSE i.e. Nominal vs. Faulted. In [MURPHY, 2005] a unique function is used to compute the scale factors Kstep which is the combination of the results for all events.

	[Murphy et al., 2009]		ENAC – Worst Case		ENAC - Mean	
Configuration Change	**150 s**	**15 s**	**150 s**	**15 s**	**150 s**	**15 s**
Single Rise Set	0.167	0.0167	0.158	0.0167	0.108	0.0111
Multiple Rise Set	0.022	0.000348	0.028	0.000696	0.006	0.00005

Table 28: rise/set probabilities

6.5.2.2 *Nominal steps: rise/set events*

We used the outputs of our simulations to update a unique histogram for rise and set events, which is represented in Figure 82. The corresponding CDF is represented in Figure 83.
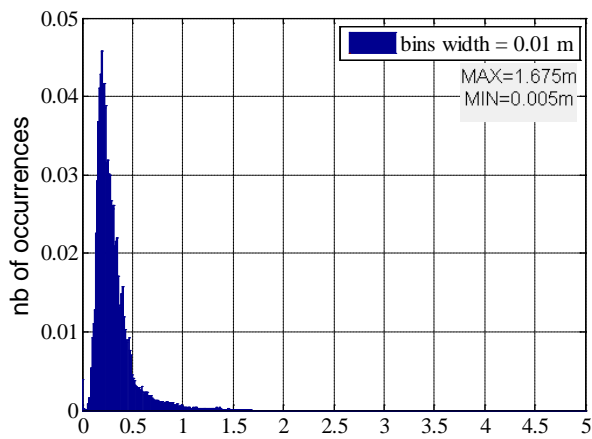
Figure 82: Observed nominal σ_{vert}^{step} distribution for GAST-C and GAD B3 using ENAC mean probabilities
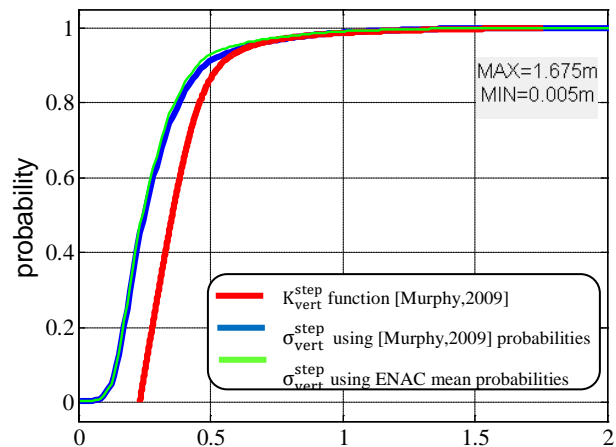
Figure 83: Comparison between observed nominal σ_{vert}^{step} CDF for GAST-C and GAD B3 and K_{vert}^{step} generation function from [MURPHY, 2009]

As we can see, our results are not exactly corresponding to the red function which has been presented in eq. (6-11). Moreover, the impact of mean probabilities instead of worst case is not very important. However, it induces a global decrease in the magnitude of σ_{step_vert} which can be of interest. Thus, we propose to draw the nominal vertical NSE step from its observed distribution and to use our computed probabilities as a reference in the rest of this study.
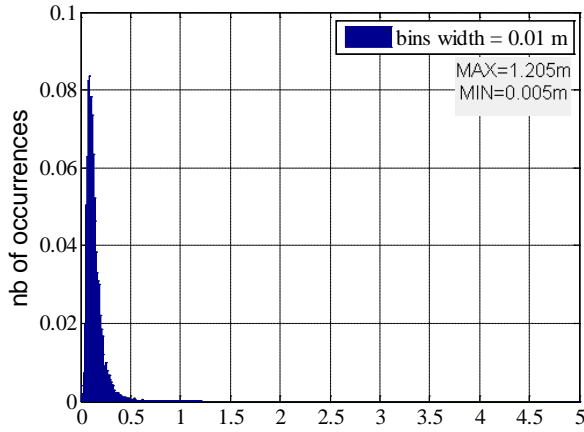
Figure 84: nominal σ_{whor}^{step} histogram for all airports for GAST-C and GAD-B3 using ENAC mean probabilities
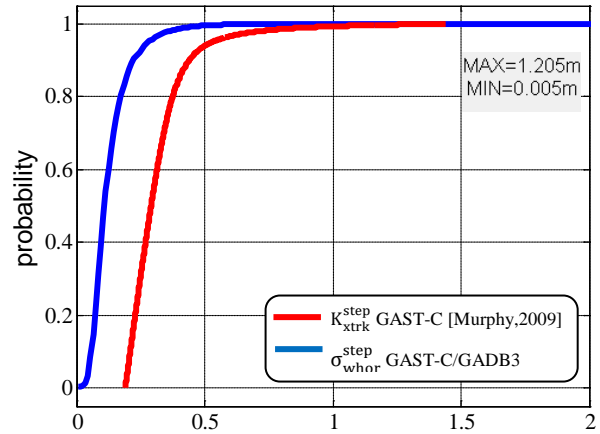
Figure 85: Comparison between observed nominal σ_{whor}^{step} CDF for GAST-C using ENAC mean probabilities and GAD B3 and K_{xtrk}^{step} generation function from [MURPHY, 2009]

Equivalent distributions have been computed for the horizontal steps magnitude and are represented in Figure 84 and Figure 85. As in the nominal case we decided to determine the orientation producing the largest value of σ_{xtrk}^{step} which we called σ_{whor}^{step}. The same mathematical theory as in the nominal NSE case is used to derive the associated results and which consisted in :

- Extracting the horizontal components of the covariance matrix of the position error C
- Computing the eigen vectors and eigen values of the obtained matrix

It allows determining the horizontal coordinate frame resulting from the rotation of the north, east coordinate frame and which produces the highest horizontal error (see section 6.5.1.3.1 for more details). The corresponding histograms and CDF are presented in Figure 84 and Figure 85. Our results are really lower than the initially published model, that's why we propose to use the same method as in the nominal case which is to draw the nominal worst horizontal step from this observed distribution and then to project this nominal worst horizontal step on the horizontal axes using a random angular deviation.

6.5.2.3   *Faulted steps: signal loss events*
The exact same figures have been computed considering signal loss events for the vertical direction and can be found in Figure 86 and Figure 87.

By observing the CDF, we can deduce that the generation function for K_{vert}^{step} of the state of the art model is widely exceeded by our observed CDF.

This is due to the fact that in the legacy model, the generation function has been designed on the basis of weighted results of Rise/set events AND Signal Loss events. The probabilities of Signal Loss events being much lower than the probabilities of Rise/Set events, the impact of the faulted case have been hidden in the final function even if it induced slightly larger steps. That's why we believe that Signal Loss events should be part of the faulted mode generator so as to properly asses their impact.

Figure 86: Observed faulted σ_{vert}^{step} distribution for GAST-C and GAD B3 using ENAC mean probabilities
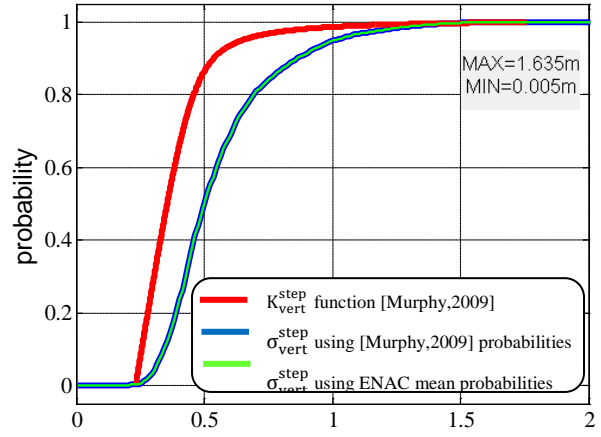
Figure 87: Comparison between observed faulted σ_{vert}^{step} CDF for GAST-C and GAD B3 and K_{vert}^{step} generation function from [MURPHY, 2009]

The obtained histograms and CDF for horizontal components are presented in Figure 88 and Figure 89. There is no particular comment to make here. We can see that our obtained CDF is very close to the function used in the legacy model. To take into account our observed results which are however slightly lower we propose to use our resulting CDF with the same mechanisms as previously to generate the horizontal steps scale factors in the faulted case.
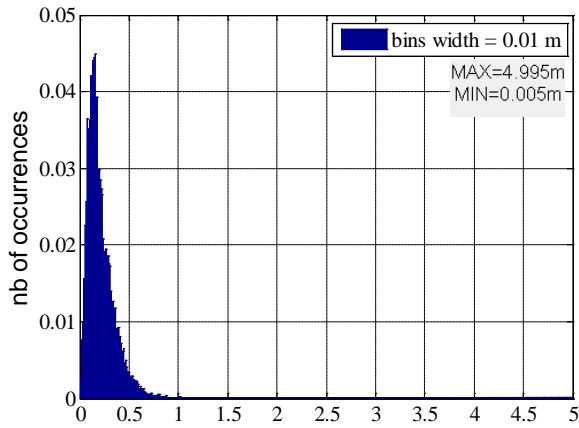
Figure 88: Observed faulted σ_{whor}^{step} distribution for GAST-C and GAD B3
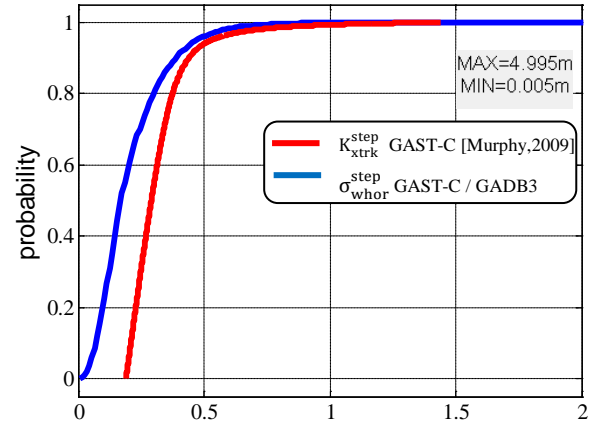
Figure 89: Comparison between observed faulted σ_{whor}^{step} CDF for GAST-C and GAD B3 using ENAC mean probabilities and K_{xtrk}^{step} generation function from [MURPHY, 2009]

6.5.2.4 *Conclusions*

The step simulations have led us to propose different evolutions of the legacy model. The main difference proposed is that it appears necessary to us to distinguish rise/set events from signal loss events. We believe that rise/set events should be considered as part of the nominal GBAS NSE generation while signal loss events should be considered as a contribution to the faulted GBAS NSE.

Therefore, we propose to draw nominal K_{vert}^{step} from the rise/set events observed distribution which is shown in Figure 82. We also propose to draw the nominal step worst horizontal scale factor K_{whor}^{step} from its computed distribution shown in Figure 84 and then project the obtained sigma on the cross-track and along-track axis using a random angular deviation.

Concerning the faulted steps, it still has to be defined how it will be taken into account. One way to deal with it may be to run simulations with extreme values of the faulted K_{vert}^{step} and K_{whor}^{step} related to the observed distributions in the case of signal loss events.

6.6 Evolutions of the model for GAST-D autoland simulations

6.6.1 Simulation assumptions

To take into account the difference of requirements between GAST-C and GAST-D concepts it has been necessary to make some modifications in our simulations. In fact, the main differences are the pseudorange measurement models which are different due to the different code-carrier smoothing time constant. This point has already been discussed in chapter 4. The main issue is that there is currently no validated pseudorange measurement models published for GAST-D. Therefore, we had to make assumptions which are supposed to be conservative as it was explained previously (please refer to sections 4.2.3.3 and 6.5.1.1.2). Some work has been done recently to solve this issue as in [MURPHY, 2010] and new material should be available soon.

However, one of the main innovative requirements brought by GAST-D concept is the geometry screening. To take into account geometry screening it would have been necessary to design an algorithm so as to be capable of rejecting geometries not supported by the simulated aircraft and thus obtain different distributions. The issue is that the results would only be relevant for a particular category of aircrafts. Moreover, the capabilities of aircrafts for geometry screening are still not clearly established and therefore it may have been unpractical to run these simulations. That's why our simulations have been run in the same conditions as for GAST-C that is to say without geometry screening but with the assumed GAST-D pseudorange measurement error models.

It is important to notice that it is stated in [RTCA, 2008] that in the case of GAST-D, GBAS airborne equipment should at least meet the requirements of AAD B. Therefore, we use AAD B instead of AAD A in the case of GAST-D simulations. We use the same level of ground facility performance as for GAST-C simulations, which is GAD B3 but the expression applied is the expression proposed for GAST-D pseudorange measurement model in section 4.2.3.3.

Our results are presented below and compared to the functions proposed in [MURPHY, 2009]. Main conclusions obtained during GAST-C simulations will not be challenged in the case of GAST-D since we assume that they are not impacted. In fact, we believe that the conclusions presented in sections 6.5.1.3.6 and 6.5.2.4 concerning the architecture of the GBAS NSE generator and its functioning for GAST-C are perfectly adapted to GAST-D. Thus, the following GAST-D sections are only intended to present the different distributions and results obtained thanks to our GAST-D simulations. These distributions should simply be substituted in the GAST-C GBAS NSE generator simulator to obtain our proposed GAST-D GBAS NSE generator.

6.6.2 Determination of nominal GBAS NSE statistics for GAST-D

6.6.2.1 Nominal Vertical GBAS NSE magnitude

As we could expect, we can observe in Figure 90 and Figure 91 that the vertical GBAS NSE has increased in comparison to GAST-C due to the new code carrier smoothing time constant of 30 seconds which implies a larger impact of noise on the measurements with respect to the classical 100 seconds smoothing.

We can see that the observed CDF is properly bounded by the generation function proposed in [MURPHY, 2005] but it seems too much conservative to us. Thus, it may be valuable to use the observed distribution so as to take advantage of this reduced NSE.
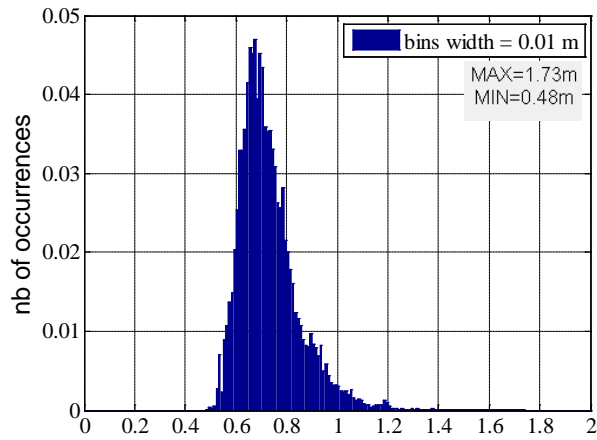
Figure 90: Observed σ_{vert} histogram for GAST-D and GAD B3 using ENAC mean probabilities
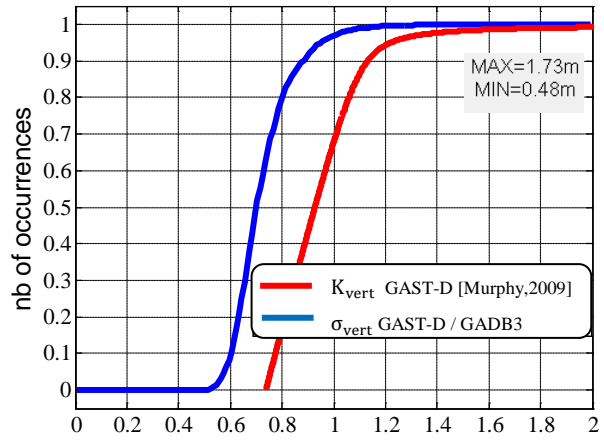
Figure 91: Comparison between observed σ_{vert} CDF for GAST-D/GAD B3 using ENAC mean probabilities and K_{vert} generation function from [Murphy et al., 2009] for GAST-D

6.6.2.2 *Nominal Horizontal GBAS NSE magnitude*

The same reasoning as in the case of nominal vertical NSE can be applied here. In fact, we can see that the function proposed in the legacy model is exceeding our observed CDF by a minimum of 0.30 meters which is huge since it means a reduction of 50% of the horizontal NSE. Therefore, we propose our observed histogram as candidate to generate the nominal worst horizontal scale factor K_{whor} in the case of GAST-D.
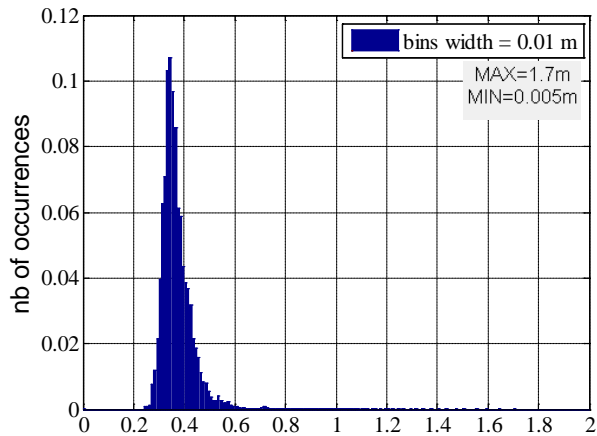
Figure 92: Observed σ_{whor} histogram for GAST-D and GAD B3 using ENAC mean probabilities
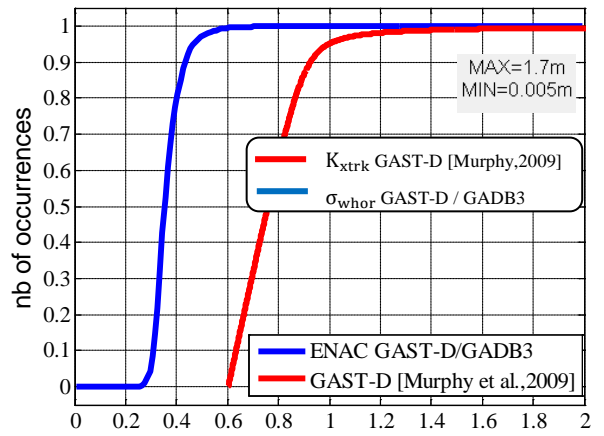
Figure 93: Comparison between observed σ_{whor} CDF for GAST-D/GAD B3 using ENAC mean probabilities and K_{xtrk} generation function from [Murphy et al., 2009]

6.6.3   Determination of steps GBAS NSE statistics for GAST-D
The same assumptions as for GAST-C steps simulations apply here.

6.6.3.1   *Nominal steps: rise/set events*
As in the nominal case, the magnitude of vertical steps due to rise/set events, which we have called nominal vertical steps, has increased in comparison to GAST-C simulations. The same conclusions can be applied here since we can see in Figure 95 and Figure 97 that the generation functions proposed

in the legacy model are much larger than our observed CDF. Therefore, we propose to take advantage of our results by drawing the nominal steps scale factors K_{vert}^{step} and K_{whor}^{step} from their respective observed distributions.
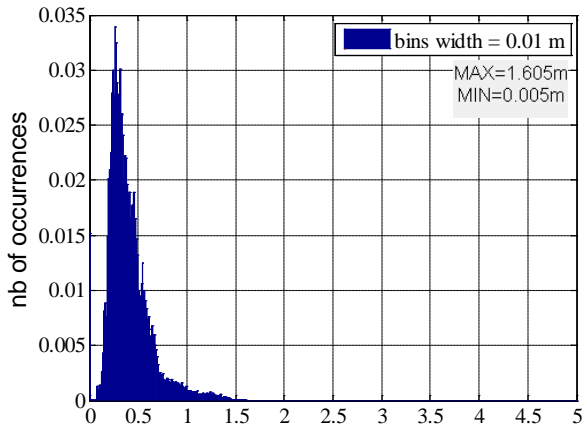
Figure 94: Observed nominal σ_{step_vert} distribution for GAST-D and GAD B3 using ENAC mean probabilities
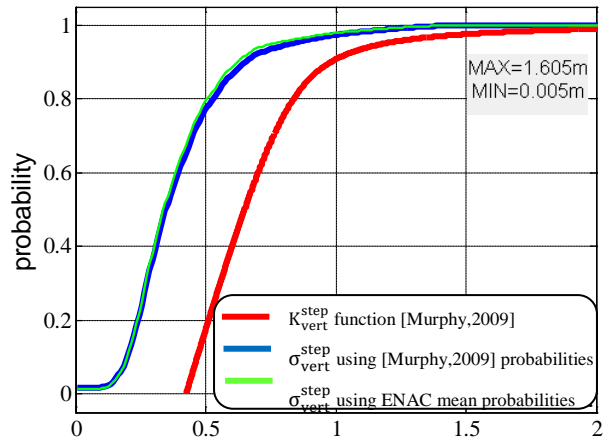
Figure 95: Comparison between observed nominal σ_{vert}^{step} CDF for GAST-D and GAD B3 and K_{vert}^{step} generation function from [Murphy et al.,2009] for GAST-D
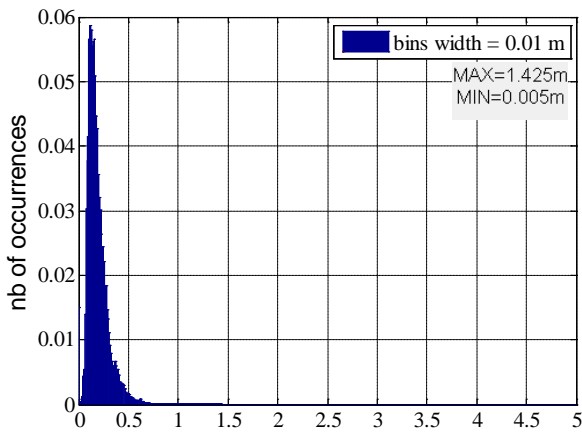
Figure 96: Observed nominal σ_{whor}^{step} distribution for GAST-D and GAD B3 using ENAC mean probabilities
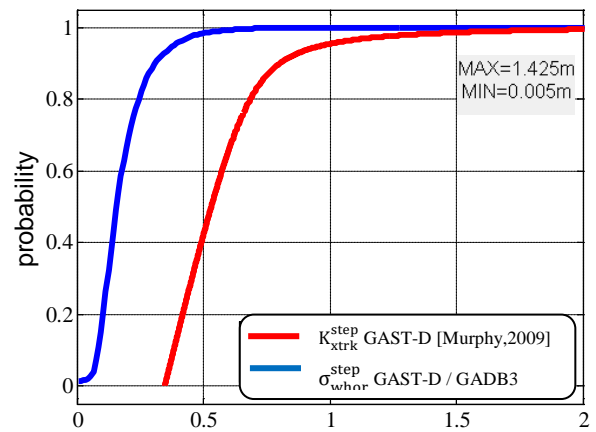
Figure 97: Comparison between observed nominal σ_{whor}^{step} CDF for GAST-D and GAD B3 using ENAC mean probabilities and K_{xtrk}^{step} generation function from [Murphy et al.,2009] for GAST-D

6.6.3.2   Faulted steps: signal loss events

The figures presented in this section allow deducing the same conclusions as in the case of GAST-C simulations. In fact, Figure 99 shows that for faulted steps, the function proposed in the legacy model for generating steps is not adapted since it is exceeded by our results. Then, it is necessary to use our observed CDF so as not to underestimate the impact of these steps.
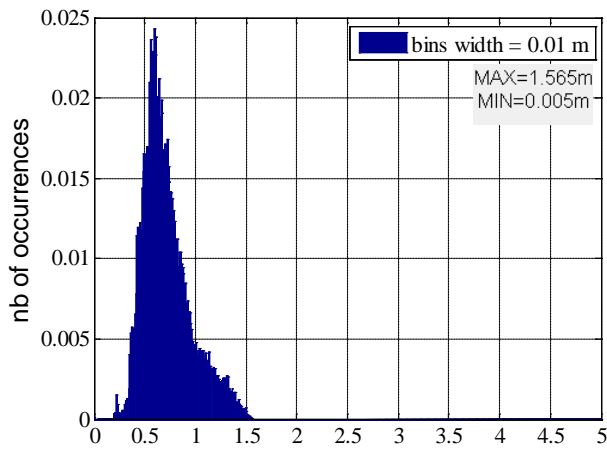
Figure 98: Observed faulted σ_{vert}^{step} distribution for GAST-D and GAD B3 using ENAC mean probabilities
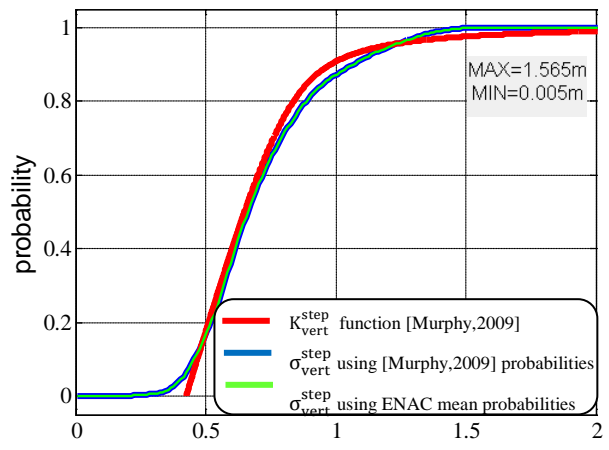
Figure 99: Comparison between observed faulted σ_{vert}^{step} CDF for GAST-D and GAD B3 and K_{vert}^{step} generation function from [Murphy et al.,2009] for GAST-D
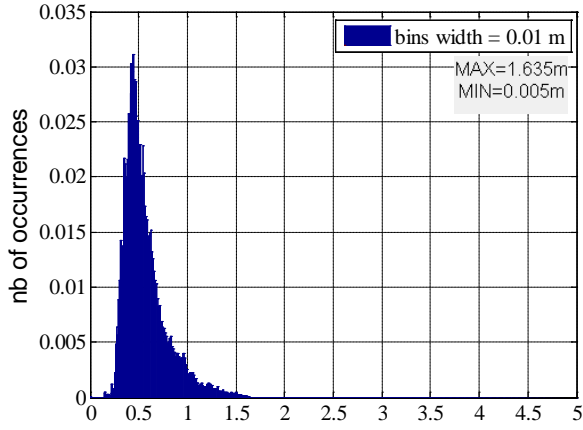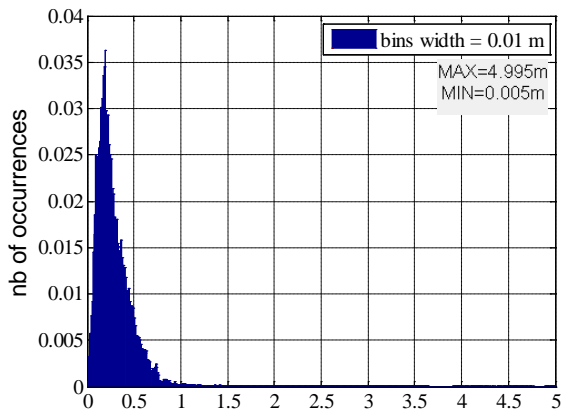

Figure 100: Observed faulted σ_{whor}^{step} distribution for GAST-D and GAD B3 using ENAC mean probabilities
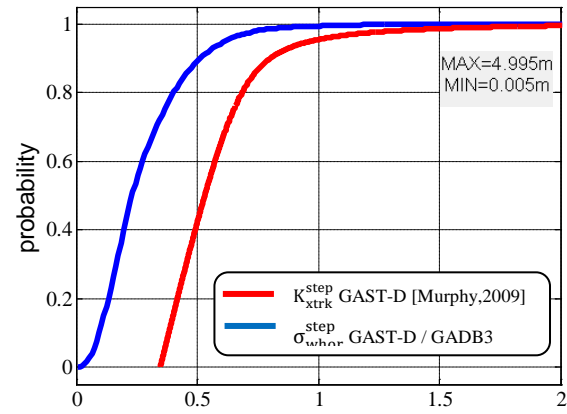
Figure 101: Comparison between observed faulted σ_{whor}^{step} CDF for GAST-D and GAD B3 using ENAC mean probabilities and K_{xtrk}^{step} generation function from [Murphy et al.,2009] for GAST-D

6.7 Synthesis

The state of the art analysis has shown that a GBAS NSE model for autoland performance simulations has already been developed [MURPHY, 2005]. This model assumes that the nominal GBAS NSE can be modelled as three independent sequences scaled by sigmas factors drawn from experimental distributions. It also assumes that the step NSE can be generated by scaling unit step functions with scale factors drawn from observed distributions. To derive our proposed GBAS NSE generator we make the same assumptions since we mathematically justified the rationale for this simulation protocol.

The state of the art model then assumes a 2nd order filter representing tracking loops and code-carrier smoothing, and scaled by sigmas factors drawn from experimental distributions. The horizontal sigmas factors are 0.818 times the vertical sigma.

However, the lack of information on the validation methods used has led us to investigate the validity of this model. Moreover, some aspects of this model need to be updated to take into account recent GBAS standards for CAT II/III.

Our work first focused on observations of the nominal NSE in the frame of GAST-C concept with our software over airports of interest for us. Taking into account the new standard pseudorange measurement error sigmas we observed slightly larger results in the vertical direction. Therefore, we propose to directly draw K_{vert} from its observed distribution. We also propose to distinguish between airports with latitudes $> 70°$ and other airports. Concerning the horizontal sigma distribution, based on our analysis and observations with our software, we propose to draw a separate horizontal sigma to reflect the diversity between vertical and horizontal sigmas. Thus, the idea is to draw the worst horizontal sigma from its observed distribution and to project it on the along track and cross track axis using a random angular deviation. To limit the drawbacks of this method - which implies that K_{vert} and K_{hor} are considered independent – we would implement a limiting technique to prevent observing too frequent large horizontal to vertical sigmas ratio. The proposed technique consists in limiting the ratio K_{hor}/K_{vert} to the worst case ratio of 0.818.

We then studied the step NSE due to constellation configuration change. The state of the art model proposed a function which combined impact of configuration changes due to Rise/Set events and Signal Loss events. We propose to distinguish between nominal steps which are due to Rise/Set events and Faulted steps which are caused by Signal Loss events. Therefore, only Rise/Set events should be taken into account into nominal autoland simulations by drawing K_{vert}^{step} and K_{whor}^{step} from their observed distributions. For horizontal step we propose as in the nominal case to draw the worst K_{whor}^{step} from its observed distribution and then to project it on the horizontal axes.

Finally, we derived equivalent distributions in the case of GAST-D which as expected appeared to be larger than in the case of GAST-C due to the use of the new code carrier smoothing time constant of 30 seconds. Of course, we propose to use the same processes to generate GBAS NSE as in the case of GAST-C with the adequate distributions.

Some parts of the state of the art model have not been completely investigated such as Limit Case/Fault mode generator but it seems to us that this module is well modelled. Concerning the 2nd order filter, additional studies may be necessary to properly represent the time behaviour of a GAST-D receiver.

To conclude, the approach used in the state of the art model is innovative and really adapted to autoland simulations but it is necessary to update some parts of it which have been listed here to reflect recent evolutions of the GBAS models and requirements.

Conclusions and Future Work

This final chapter gathers the conclusions obtained thanks to the study presented in this report. It also addresses the perspectives of future projects which could be led so as to complement the previous results and foster the understanding of satellite navigation problematic specific to civil aviation users.

7.1 Conclusions

An initial study of the on-going projects at the standardization level led to the selection of two particularly promising future GNSS receiver families for civil aviation navigation to be used during approaches with vertical guidance and precision approaches:

- GPS L1 C/A and GBAS receivers: the interest of this solution is that it is anticipated that it will allow to reach SIS requirements down to CAT-II/III
- Combined GPS-GALILEO and RAIM receivers: the main advantages of these receivers are the simplicity of implementation of the integrity monitoring algorithm and of the combination model and the fact that it is autonomous, meaning that no additional external infrastructures are needed. Moreover, thanks to the use of dual constellation, this type of receivers may allow to conduct CAT-I precision approaches

Since our aim was to study the temporal behaviour of the Navigation Sensor Error (NSE) at the output of these receivers, it was necessary to develop adequate models to obtain interesting simulation results.

Concerning the GPS-GALILEO and RAIM part of the study, the necessity to precisely represent the behaviour of the position error led to the development of an IQ correlator outputs simulator. This type of simulator has the advantage to permit to represent GNSS receivers signal processing with a high degree of representativeness with a computational burden which is important but not excessive. The developed GNSS simulator includes the capability to simulate a wide range of GPS and GALILEO signals and different integrity monitoring algorithms including RAIM. A pseudorange error model has been defined to represent the impact of the errors affecting the pseudorange at the input of a GNSS receiver.

The pseudorange measurement error models corresponding to GPS-GALILEO and RAIM receiver have also been detailed depending on the configuration of the receiver (single/dual frequency, single/dual constellation...).

Using the GNSS receiver simulator, a study of the temporal behaviour of the NSE at the output of a combined GPS-GALILEO and RAIM receiver has been studied. In particular, we focused our analysis on the impact of constellation changes on the NSE which are NSE steps. Particular attention was paid to the final part of approaches which begins at the FAF and terminates at the touchdown of the aircraft.

This position solution has proven very good performance in terms of accuracy as expected thanks to the high number of satellites provided by the combination of two different GNSS constellations. Moreover, even if it was not the purpose of this study, we can remark that we have observed a 100% availability of the position solution during our tests. Another interesting result is that the fault-free combined GPS/GALILEO VPE is almost constant during an approach but can be as large as a few meters. The associated estimated standard deviation of the VPE can be a few 10 cm, and the probability that this standard deviation is higher than 40 cm is lower than 5%.

However, simulations have also shown that constellation changes can effectively result in unexpected position error steps, which can be particularly disturbing in the vertical direction. In fact, the maximum observed amplitude of vertical position error steps induced by constellation changes was around 1.9 meters. Equivalently, we recorded the maximum vertical protection level steps due to constellation changes, which is around 4,40 meters.

To deal with the constellation changes and avoid their impact during final approaches, a simple algorithm called the constellation freezing algorithm was derived. This algorithm consists in anticipating the rise or set of satellites which are predicted to occur during the final approach, and to inhibit the use of the associated pseudorange measurements in the position solution.

Thanks to this algorithm, it was possible to completely avoid position steps due to constellation changes during final approach at the expense of the artificial loss of still available satellites when passing the FAF, which do not have a significant impact thanks to the high number of satellites available. Moreover, we even observed a slight reduction of the maximum estimated standard deviation of the Vertical Position Error (VPE) from 0.91 meters to 0.79 meters after the FAF. The overall accuracy of the position solution is not affected by the proposed algorithm, thanks to the trade-off between the improvements of the average UERE through removal of low elevation satellites, while the DOPs remain essentially unchanged.

We must however remark that the maximum standard deviation of the VPE before the FAF has consequently increased. Moreover, the amplitude of the VPE steps due to constellation changes and observed before the FAF has slightly increased due to the artificial and maybe simultaneous removal of several satellites at the FAF.

For the GBAS part of the study, the chosen strategy was different since we were particularly interested in a model representing the GBAS NSE and which could be used for GBAS CAT II/III autoland capabilities demonstration. In fact, a model for GBAS CAT-I autoland simulations was already published a few years ago, but with insufficient information on its validation. This model proposes to model the GBAS NSE for CAT-I autoland simulations in nominal conditions as well as in limit and fault conditions. During this study we were interested in the nominal behaviour of the NSE and that's why we focused on the nominal part of the model. The idea was then to use this state of the art model as the starting point of our work. A complete review of this nominal model is proposed. It can be noticed that GBAS faults have been investigated in another study which can be found in [MURPHY, 2010b].

In fact, the nominal part of the state of the art model is composed of two modules:

- The nominal NSE generator: it is used to generate the nominal NSE at the output of a GAST-C GPS L1 C/A and GBAS receiver without considering the impact of constellation changes. It assumes that it is possible to directly model the three components of the NSE by generating three independent sequences with unity variance and scaled by sigma factors drawn from distributions derived through simulations.
- The step NSE generator: it is used to generate the NSE steps at the output of a GAST-C GPS L1 C/A and GBAS receiver. It assumes that it is possible to directly generate the three components of the NSE steps by multiplying unity steps by scale factors drawn from distributions derived through simulations. These steps are then added to the outputs of the nominal NSE generator so as to obtain the final simulated GBAS NSE.

We then propose in this study a complete mathematical analysis of the model which allows validating the structure of the simulator and the model strategy described above. However, we also propose possible improvements in particular concerning the pseudorange measurement error models used to derive the distributions implemented in the state of the art model.

On the basis of these propositions, we ran new simulations to derive accurate experimental distributions for the scale factors of the nominal NSE generator and the steps NSE generator, so as to reflect the latest evolutions of the GBAS standards. Several simulations were analysed for different airport locations across the world so as to assess the statistics of the scale factors for horizontal and nominal scale factors as well as horizontal and vertical step scale factors. The obtained results demonstrate that the state of the art model is a good representation of the GBAS GAST-C NSE but it is very conservative, which could be potentially a disadvantage to demonstrate the capability to autoland in CAT II-III conditions with GBAS GAST-D equipments.

That is why an alternate model derived from the state of the art model is proposed in this report. The new GBAS NSE model for CAT-I autoland simulations defined here exploits the simulation results obtained during our study. The main evolutions are listed below for the nominal NSE generator and the Steps NSE generator:

Nominal NSE generator:

- K_{vert} scale factors directly drawn from their observed distribution
- A horizontal scale factor K_{worst_hor} is defined which is directly drawn from its observed distribution. Its distribution was derived by determining at each simulation epoch the direction in the horizontal plane producing the worst scale factor.
- The horizontal scale factors K_{xtrk} and K_{atrk} are obtained by randomly projecting the K_{worst_hor} scale factor on the $xtrk$ and $atrk$ axes. The drawback of this method is that the horizontal scale factors are independent with the vertical scale factor which is not true. The advantage is that this method allows to better represent the diversity of vertical/horizontal scale factors.
- To attenuate the impact of the disadvantage of the method proposed for horizontal scale factors generation, a maximum ratio K_{hor}/K_{vert} is defined so as to avoid incoherent scale factors.

Steps NSE generator:

- The first proposal is to exclude constellation changes due to signal loss events from the steps NSE generator since we consider this type of events as faults. We thus define a nominal steps NSE generator representing steps due to constellation changes which are rise/set of GNSS satellites. We then define a faulted steps NSE generator representing steps due to constellation changes induced by signal loss events.
- We then propose to apply the same propositions as the one concerning the nominal NSE generator to the Steps NSE generator and listed above.

The proposed GBAS NSE model for GBAS CAT-I autoland simulations has been completely defined. The final step of our study concerning GBAS was to propose a new model to support the simulations for GBAS CAT II-III autoland simulations. The proposed model has exactly the same structure and functioning as the one for CAT-I simulations. However, the distributions used to draw the different scale factors have been recomputed to take into account pseudorange measurement error models

adapted to CAT II-III simulations. For example the evolution of the code-carrier smoothing time constant from 100 seconds to 30 seconds has been reflected. Thus, a complete GBAS NSE model for GBAS CAT II-III autoland simulations has also been defined.

7.2   Original Contributions

This section gives a brief insight in the original contributions obtained thanks to this study. Each point is extensively described all along this document:

- Development of a GNSS receiver simulator including multi-constellation and multi-frequency position solution.
- Study of the temporal behaviour of GPS/GALILEO and LSR RAIM NSE and assessment of the impact of constellation changes.
- Design of an algorithm to prevent the impact of constellation changes on GPS/GALILEO and LSR RAIM position solution during final approaches.
- Analysis of a state of the art GBAS GAST-C NSE model for GBAS CAT-I autoland simulations
- Proposal of an alternate model for GBAS CAT-I autoland simulations
- Proposal of a GBAS GAST-D NSE model for GBAS CAT-II/III based on the previous so as to reflect evolution of civil aviation requirements concerning GBAS GAST-D service.

The publications made during this PhD work are listed below:

- "Study of a GBAS model for CAT II/III Simulations",
  P. NERI, C. MACABIAU, L. AZOULAI
  Proceedings of ION GNSS 2009, September 2009
- "A new GBAS NSE model for CAT II/III Autoland Simulations",
  P. NERI, C. MACABIAU, L. AZOULAI, J. MULLER
  Proceedings of IEEE/ION PLANS 2010, May 2010
- "Study of the temporal behaviour of GPS/GALILEO NSE and RAIM for LPV200",
  P. NERI, C. MACABIAU, L. AZOULAI
  Proceedings of ION GNSS 2011, September 2011
- "Presentation of a new GBAS NSE model for CATII/III autoland simulations"
  P. NERI, C. MACABIAU, L. AOULAI, J. Muller
  RTCA SC-159 working group 4, Washington D.C., February 2010

7.3   Way Forward

Concerning the RAIM part of this study, additional studies could be conducted to completely assess the potential of combined GPS-GALILEO and RAIM receivers. In fact, it would be interesting to further study the impact of switching from nominal modes to degraded modes by considering the impact of frequency losses or even constellation losses on the position solution during critical phases of flight such as approaches with vertical guidance and precision approaches. Another aspect would be to study the behaviour of the position solution as well as its availability with the progressive introduction of satellites of the GALILEO constellations. It would be an opportunity to determine the critical number of operational GALILEO satellites needed to use combined GPS-GALILEO and RAIM receivers assuming only Major Service Failures affect the constellations. Moreover, we have only considered here the classical LSR RAIM algorithm. More innovative algorithm such as ARAIM may be studied.

Concerning the study of the GBAS NSE model for autoland simulations, some aspects of the model have not completely been validated. In particular, the 2nd order Butterworth filter used to represent the impact of the cascade of the signal processing and code carrier smoothing filter on the GBAS NSE have been overlooked. Different receivers may be studied and modelled to be implemented in the GBAS NSE model. One other aspect that must be investigated is the impact of loss of satellites signals due to aircraft wings outages during turns with important bank angles.

Moreover, our study focused on the nominal behaviour of the GBAS NSE in the frame of CAT-I and CAT-II/III autoland simulations. Nonetheless, to demonstrate autoland capabilities, it is necessary to statistically model the NSE but it is also mandatory to play limit and fault cases. These cases even if described in the state of the art model, have not been addressed in this study since faults were not in the initial scope of this PhD. It may thus be necessary to validate the models proposed in the state of the art model.

Pseudorange measurement errors models were proposed in this study for the measurements made by GAST-D GBAS receivers. These models were determined by adapting GAST-C pseudorange measurement error models, taking into account evolution of the standards defining GAST-D receivers. These models have not been fully validated yet and it is thus important to compare the models with real data coming from real receivers. In fact, these models may be too much conservative with respect to reality.

Finally, it was not possible to include in the scope of this study every potentially promising GNSS receivers, we had thus to select combined GPS and GALILEO and RAIM receivers, as well as GPS L1 C/A and GBAS receivers. However, several other candidates could have been studied in prevision of the future introduction of new GNSS signals with improved characteristics as well as new constellations and augmentation systems. We can mention in particular SBAS systems which are very promising thanks to the development of several SBAS systems (WAAS, EGNOS, GAGAN, MSAS...). In fact, the possibility to develop an SBAS NSE generator equivalent to the model proposed in this report for GBAS has been mentioned. Some guidelines for this development have already been proposed and the complete study may be launched in the near future.

Further studies are thus necessary to be able to fully conclude on which receiver architectures will be implemented in future GNSS receivers for civil aviation users.

References

[ARBESSER, 2006]	B. Arbesser-Ratsburg, "The GALILEO Single Frequency Ionospheric Corection Algorithm", Third European Space Weather Week, November, 2006
[ARINC, 2004]	ARINC Engineering Services, "Navstar GPS Space Segment/Navigation User Interfaces", IS-GPS-200 Revision D, 2004
[AVILA-RODRIGUEZ, 2006]	Avila-Rodriguez, J.-A., S. Wallner, G. Hein, E. Rebeyrol, O. Julien, C. Macabiau, L. Ries, A. Delatour, L. Lestarquit, and J.-L. Issler, "CBOC – An Implementation of MBOC", First CNES Workshop on GALILEO Signals and Signal Processing, Toulouse, France, Oct. 12-13, 2006
[BASTIDE, 2001]	F. Bastide, "Equivalent C/N_0 degradation in presence of interferences", 2001
[BASTIDE, 2002]	Bastide, F., O. Julien, C. Macabiau, B. Roturier, "Analysis of L5/E5 Acquisition, Tracking and Data Demodulation Thresholds", Proceedings of the Institute of Navigation GPS, Portland, OR, USA, Sept. 24-27, 2002
[BETZ, 2000]	J. W. Betz and K.R. Kolodziejski, "Extended Theory of Early-Late Code Tracking for Bandlimited GPS receiver", Navigation: Journal of The Institute of Navigation, Fall 2000
[BETZ, 2002]	J. W. Betz, "Binary Offset Carrier Modulations for Radionavigation", Journal of the Institute of Navigation, VOL.48, No.4, Winter 2001-2002
[BETZ, 2006]	Betz, J., M. Blanco, C. Cahn, P. Dafesh, C. Hegarty, K. Hudnut, A. Jones, V. Kasemsri, R. Keegan, K. Kovach, S. Lenahan, H. Ma, and J. Rushanan (2006), "Description of the L1C Signal", Proceedings of the ION GNSS Conference, Fort Worth, TX, USA, 26-29 Sept
[BLANCH, 2007]	Blanch J., A. Ene, T.Walter, P. Enge, "An optimized multiple hypothesis RAIM algorithm for vertical guidance", ION GNSS 2007, Sept. 2007
[BOOTH, 2000]	J. Booth, T. Murphy, B. Clark, F. Liu, "Validation of the Airframe Multipath Error Allocation for Local Area Differential GPS", Proceedings of the IAIN/ION Meeting, June 2000
[BROWN, 1988]	Brown R. G., P. McBurney, "Self-contained GPS integrity check using maximum solution separation", Navigation: Journal of the institute of navigation, Vol. 35, Spring 1988
[CABLER, 2002]	H. Cabler, B. DeCleene, "LPV : New, Improved WAAS Instrument Approach", Proceedings of ION GPS 2002
[CICTT, 2006]	Commercial Aviation Safety Team/International Civil Aviation Organization, Common Taxonomy Team, "Phases of flight definitions and usage notes", Feb. 2006

[DECLEENE, 2007]   B. DeCleene, "Proposed Revision to GNSS Performance Requirements for Precision Approach", Navigation Systems Panel (NSP) Working Groups 1 and 2 meetings, 2007

[EASA, 2003]   EASA, CS AWO 1, "Joint Aviation Requirements – All Weather Operations", Subpart 1, "Automatic Landing Systems", 2003

[ESA, 2005]   European Space Agency, "GALILEO Integrity Concept", Technical Note, Jul. 2005

[EUROCAE, 2007]   EUROCAE WG-62, "Interim Minimum Operational Performance Specification for Airborne GALILEO Satellite Receiving Equipment", version 0.26, 2007

[EUROCAE, 2009]   GALILEO/GPS AdHoc group of Eurocae WG-62, "Concept of Operations for combined GALILEO/GPS receivers", Issue 4, Version 1.0, 2009

[EUROCAE, 2010]   EUROCAE Web Site, http://www.eurocae.net, Oct.2010

[FAA, 1999]   FAA Advisory Circular 120-28D, "Criteria for Approval of Category III Weather Minima for Takeoff, Landing and Rollout", Jul. 13, 1999

[GEAS, 2010]   "GNSS Evolutionary Architecture Study", Phase 2-Panel report, February 2010

[GJU, 2010]   GALILEO Joint Undertaking, "European GNSS (GALILEO) Open Service, Signal In Space Interface Control Document (OS SIS ICD)", Issue 1.1, Sept. 2010

[GODET, 2006]   Godet J. and Col. M. Crews WG A, "Recommendations on L1 OS/L1C Optimization", Mar. 22, 2006

[GPS SPS, 2008]   "Global Positioning System Standard Positioning Service Performance Standard", Department of Defense, United States Of America, Sept. 2008

[HAHN, 2005]   Hahn J. H., E.D. Powers, "Implementation of the GPS to GALILEO Time Offset (GGTO)", IEEE Frequency Control Symposium and Exposition, August 2005

[HEGARTY, 2006]   C. Hegarty, "Analytical derivation of maximum tolerable in-band interference level for aviation applications of GNSS", 1996

[HEIN, 2006]   Hein G.W., Avila-Rodriguez J.-A., Wallner S., Pratt A.R., Owen J.I.R., Issler J.-L., Betz J.W., Hegarty C.J., Lt Lenahan L.S., Rushanan J.J.,Kraay A.L., Stansell T.A., "The New Optimized Spreading Modulation Recommended for GALILEO L1 OS and GPS L1C", Proceedings of IEEE/ION PLANS 2006 –24-27 Apr. 2006, San Diego, California, USA.

[HILBRECHT, 2006]   Hilbrecht H. and R. Braibanti,Braibanti, "Joint Statement on GALILEO and GPS Signal Optimization by the European Commission and the United States", Mar. 24, 2006

[HOLMES, 1982]	J.K. Holmes, "Coherent spread spectrum systems", John Wiley and Sons, New York, 1982
[HOLMES, 2000]	J.K. Holmes, "Code Tracking Loop Performance Including the Effect of Channel Filtering and Gaussian Interference", Proceedings of the US Institute of Navigation AM, San Diego, CA, USA, Jun. 2000
[HURST,1972]	Hurst R.L. and Knop R.E., "Generation of Random Correlated Normal Variables", Communications of the ACM, Vol.15, No5, 1972
[HWANG, 2006]	Hwang P.Y, R.G. Brown, "Raim FDE revisited: a new breakthrough in RAIM availability performance with NIORAIM", Journal of the institute of navigation, Vol.53, Spring 2006
[ICAO, 2001]	International Civil Aviation Organization, "International Standards and Recommended Practices, Annex 6 to Convention on International Civil aviation, Operation of aircraft", Eight edition, Jul. 2001
[ICAO, 2006]	International Civil Aviation Organization, "International Standards and Recommended Practices, Annex 10 to Convention on International Civil Aviation", Volume I, Radio Navigation Aids, Sixth edition (including Amendment 85 Dated July 12, 2010) , Jul. 2006
[ICAO, 2007]	International Civil Aviation Organization, "International Standards and Recommended Practices", Annex 10 to Convention on International Civil Aviation, Volume I, Radio Navigation Aids, Amendment 82, Nov. 2007
[ICAO, 2008]	International Civil Aviation Organization, "Performance-based Navigation (PBN) Manual", Third Edition, 2008
[ICAO, 2010]	International Civil Aviation Organization Web Site, http://www.icao.int/, October 2010
[ICAO, 2010a]	ICAO Annex 10 GBAS SARPS - draft - May 2010
[ITU, 2012]	International Telecommunication Union, "The Radio Regulations (Edition 2012)", Nov. 2012
[JULIEN, 2005]	Olivier JULIEN, "Design of GALILEO L1F Receiver Tracking Loops", PhD Thesis, University Of Calgary, Jul. 2005
[JULIEN, 2006]	Julien, O., 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 Oct. 2006
[JULIEN, 2007]	O.JULIEN, C. MACABIAU, J-L. ISSLER, L. RIES, "1-bit Processing of Composite BOC (CBOC) Signals and Extension to Time-Multiplexed BOC (TMBOC) signals", ION NTM 2007
[JULIEN, 2010]	O.JULIEN, C. MACABIAU, J-L. ISSLER, L. RIES, "GALILEO E1 OS/SoL Acquisition, Tracking and Data Demodulation Performances for Civil Aviation", NAVITEC 2010

[KLOBUCHAR, 1987]   J.A. KLOBUCHAR, "Ionospheric time-delay algorithm for single-frequency GPS users", IEEE transactions on Aerospace and Electronic Systems, 1987

[LEE, 2004]   Lee Y.C., "Performance of Receiver Autonomous Integrity Monitoring (RAIM) in the Presence of Simultaneous Multiple Satellite faults", ION Annual Meeting, 2004

[LEE, 2007]   Lee Y. C., M. P. McLaughlin, "Feasibility Analysis of RAIM to provide LPV-200 Approaches with Future GPS", ION GNSS 2007, September 2007

[LEICK, 1995]   A.LEICK, "GPS Satellite Surveying", Wiley-Interscience, 2nd edition, 1995

[MACABIAU et al., 2006]   C.Macabiau, M. Moriella, M.Raimondi, C.Dupouy, A. Steingass, A. Lehner, "GNSS Airborne Multipath Errors Distribution Using the High Resolution Aeronautical Channel Model and Comparison to SARPS Error Curve", Proceedings of ION NTM 2006

[MACABIAU, 1997]   Macabiau C., "Analysis of the Feasibility of Using GPS Carrier Phase Ambiguity Resolution Techniques for Precision Approaches", PhD thesis, Institut National Polytechnique de Toulouse, Sep. 1997

[MARTINEAU, 2008]   Martineau A., "Perfomance of Receiver Autonomous Integrity Monitoring (RAIM) for Vertically Guided Approaches", PhD thesis, Institut National Polytechnique de Toulouse, Nov. 2008

[MONTLOIN, 2011]   Leslie Montloin, "Review of State of the Art of A/C navigation", ENAC CNS-LTST Internal Report, 2011

[MURPHY, 2000]   T. Murphy, J. Booth, "GBAS SARPs Review and Validation of Airborne Multipath Requirements", GNSSP WG B, Canberra, Jan. 2000

[MURPHY, 2005]   W.M. Harris, T.A. Murphy, Volume V, "Characterization of the GBAS System Output", Document number D683447-5, Oct. 19, 2005

[MURPHY, 2009]   T.A. MURPHY, "SARPS Support for Airworthiness Assessments GLS Signal Modeling", NSP, Brétigny, 17-27 March 2009

[MURPHY, 2010]   T. Murphy, M. Harris, S. Beauchamp, "Implications of 30-Second Smoothing for GBAS Approach Service Type D", ION ITM 2010

[MURPHY, 2010b]   T. Murphy, C. Shively, L. Azoulai, M. Brenner, "Fault modelling for GBAS Airworthiness Assessments", Proceedings of ION GNSS 2010, September 2010

[NIKIFOROV, 2005]   Nikiforov I., B. Roturier, "Advanced RAIM algorithm: first results", ION GNSS, Sep. 2005

[PAIMBLANC, 2006]   P. PAIMBLANC, "Integrity risk in satellite radionavigation systems", PhD Thesis, Sep. 2006

[REBEYROL, 2007]   E. REBEYROL, PhD thesis, "GALILEO signals and payload

optimization", Oct. 2007

[ROTURIER, 2004]	Roturier B., "APV EGNOS procedures, an efficient and reliable answer to the need for vertical guidance for approaches", Technical revue n°67, Service Technique de la Navigation Aérienne, Nov. 2004
[RTCA, 1998]	"Minimum Aviation System Performance Standards (MASPS) for the Local Area Augmentation System (LAAS)", DO-245 , Sep. 1998
[RTCA, 2003]	"Minimum Aviation System Performance Standards: Required Navigation Performance for Area Navigation", DO-236B, RTCA SC-159, 2003
[RTCA, 2004]	"Minimum Aviation System Performance Standards for Local Area Augmentation System", DO245-A, RTCA SC-159, 2004
[RTCA, 2006]	"Minimum Operational Performance Standards for Global Positioning System/Wide Area Augmentation System Airborne Equipment", DO229-D, RTCA SC-159, 2006
[RTCA, 2008]	"Minimum Operational Performance standards for GPS Local Area Augmentation System Airborne Equipment", DO-253C, RTCA SC-159, Dec 2008
[RTCA, 2010]	RTCA Inc. Web Site, http://www.rtca.org, October 2010
[SALOS, 2010]	S.Salos, C.Macabiau, A.Martineau, B.Bonhoure, D.Kubrak, "Nominal GNSS pseudorange measurement model for vehicular urban application", Proceedings of IEEE/ION PLANS 2010
[SIA, 2010]	Service de l'information aéronautique Web Site, http://www.sia.aviation-civile.gouv.fr/, 2010
[SPILKER, 1996]	J.SPILKER, Global Positioning System : Theory and Application, volume1, AIAA, 1996
[STEINGASS ET AL., 2004]	A.Steingass, A.Lehner, F.Pérez Fontán, E.Kubista, M.Jesús Martin and B.Arbesser-Rastburg, "The High Resolution Aeronautical Multipath Navigation Channel", Proceedings of ION GPS 2004
[STEPHENS, 1995]	S.A. Stephens, J.B. Thomas, "Controlled-Root Formulations for Digital Phase Locked Loops", IEEE Transactions on Aerospace and Electronic Systems, Vol.31, 1995
[WALTER, 1995]	T. Walter, J. Blanch, P. Enge, B. Pervan, L. Gratton, "Weighted RAIM for precision approaches", ION GPS 1995, September
[WALTER, 2008]	T. Walter, J. Blanch, P. Enge, B. Pervan, L. Gratton, Future Architecture to Provide Aviation Integrity", ION NTM 2008
[WINKEL, 2000]	Jon Olafur Winkel, Phd Thesis, "Modeling and Simulating GNSS Signal Structures and Receivers", 2000

Appendix A: Integrity Monitoring

The aim of this section is to present alternate integrity monitoring algorithms which have been developed to be implemented in the receiver simulator but which were not used to derive our main results.

9.1 SBAS

A module has been developed to model the use of a GNSS receiver augmented by SBAS and in particular EGNOS. To enhance the overall performance of GNSS, EGNOS transmits differential corrections to SBAS users as well as integrity messages thanks to geostationary satellites.

EGNOS integrity service is supposed to protect users from:

- GPS Satellites and geostationary satellites failures by detecting and excluding faulty satellites thanks to GPS measurements made by EGNOS reference ground stations.
- Transmission of inaccurate differential corrections due to ground segment failures for example.

We briefly remind here the pseudorange measurement error model used for EGNOS as well as the computation of EGNOS protection levels.

9.1.1 Pseudorange measurement error model

The variance of the residual pseudorange measurement error after application of EGNOS differential corrections can be modelled as:

$$\sigma_i^2 = \sigma_{i,flt}^2 + \sigma_{i,UIRE}^2 + \sigma_{i,air}^2 + \sigma_{i,tropo}^2$$

(9-1)

We detail hereafter the different terms of this expression.

- $\sigma_{i,flt}^2$ is the variance of the residual error after application of fast and long term corrections. In the case where fast and long term corrections have been applied ad that a degradation model is used, then:

$$\sigma_{i,flt}^2 = (\sigma_{UDRE} + \varepsilon_{fc} + \varepsilon_{rrc} + \varepsilon_{ltc} + \varepsilon_{er})^2$$

(9-2)

    Where:

    - ε_{fc} is a degradation parameter for fast corrections such that $\varepsilon_{fc} < 0.35\ m$
    - $\varepsilon_{rrc}, \varepsilon_{ltc}, \varepsilon_{er}$ are degradation parameters than can be taken null if no messages are missed
    - σ_{UDRE} is transmitted in the navigation message

- $\sigma_{i,UIRE}^2$ is the variance of the residual error after application of ionospheric corrections. In the case such ionospheric corrections are applied:

$$\sigma_{i,UIRE}^2 = F_{pp}^2 . \sigma_{i,UIVE}^2$$

(9-3)

    Where:

    - F_{pp} is an obliquity factor depending on the elevation angle of the satellite.
    - $\sigma_{i,UIVE}$ is deduced by interpolating the variance model of the vertical ionospheric delays given at piercepoints and transmitted in the navigation message.

- $\sigma_{i,air}^2$ is the variance of Airborne Receiver Errors. It is composed of different sources of error and in particular noise and multipath:

$$\sigma_{i,air}^2 = \sigma_{i,noise}^2 + \sigma_{i,multi}^2 + \sigma_{i,divg}^2$$

(9-4)

Where:

- The installed multipath error for the airborne receiver is described by a zero-mean normal distribution with variance (in meters):

$$\sigma_{i,multi}^2 = \left(0.13 + 0.53.\,e^{-El_i/10}\right)^2$$

El is the elevation of the satellite in degrees

- $\sigma_{i,divg}^2$ shall be greater than or equal to the differentially-corrected pseudorange error induced by the steady-state effects of the airborne smoothing filter given an ionospheric divergence that is defined to have a constant rate of 0.018m/s [RTCA, 2006]
- $\sigma_{i,noise}^2$ is the variance of a normal distribution that bounds the errors in the tails of the distribution associated with GNSS receiver for the ith satellite including receiver noise, thermal noise, interference, inter-channel biases, extrapolation, time since smoothing filter initialization, and processing errors [RTCA, 2006]

For GPS satellites with minimum signal level and receiver accuracy designator A:

$$\left(\sigma_{i,noise}^2 + \sigma_{i,divg}^2\right)^{1/2} \leq 0.36 \text{ m}$$

(9-5)

For SBAS satellites with minimum signal level and receiver accuracy designator A:

$$\left(\sigma_{i,noise}^2 + \sigma_{i,divg}^2\right)^{1/2} \leq 1.8 \text{ m}$$

(9-6)

- $\sigma_{i,tropo}^2$ is the variance of the residual error after application of tropospheric corrections. If the user applies the tropospheric correction model then,

$$\sigma_{i,tropo} = \sigma_{TVE}.\,m(El_i)$$

(9-7)

With

- $\sigma_{TVE} = 0.12$ m variance of the vertical tropospheric error
- $m(El_i)$ the mapping function depending on the elevation of the ith satellite.

The models associated to the parameters sent in the data messages will not be detailed here.

9.1.2 Protection levels computation

SBAS protection levels computations are standardized and can be found in [RTCA, 2006]. The protection levels are given by:

$$HPL_{SBAS} = K_H.\,d_{major}$$

$$VPL_{SBAS} = K_V . d_U$$

<div align="right">(9-8)</div>

The parameters K_H and K_V are derived from the desired level of performance and therefore depend on the phase of flight. In the case of precision approaches for example:

$$\begin{cases} K_V = 6.0 \\ K_H = 5.33 \end{cases}$$

<div align="right">(9-9)</div>

The parameter d_{major} is obtained by the following expression:

$$d_{major} = \sqrt{\frac{d_{east}^2 + d_{north}^2}{2} + \sqrt{\left(\frac{d_{east}^2 + d_{north}^2}{2}\right)^2 + d_{EN}^2}}$$

<div align="right">(9-10)</div>

Where:

- $d_{east}^2 = \sum_{i=1}^{N} s_{east,i}^2 . \sigma_i^2$ is the variance of a model distribution that overbounds the true error distribution in the east axis.
- $d_{north}^2 = \sum_{i=1}^{N} s_{north,i}^2 . \sigma_i^2$ is the variance of a model distribution that overbounds the true error distribution in the north axis.
- $d_{EN}^2 = \sum_{i=1}^{N} s_{east,i} . s_{north,i} . \sigma_i^2$ is the covariance of a model distribution in the east and north axis.
- $d_U^2 = \sum_{i=1}^{N} s_{U,i}^2 . \sigma_i^2$ variance of a model distribution that overbounds the true error distribution in the vertical axis.
- $s_{east,i}$ is the partial derivative of position error in the east direction with respect to the pseudorange error on the ith satellite.
- $s_{north,i}$ is the partial derivative of position error in the north direction with respect to the pseudorange error on the ith satellite.
- $s_{U,i}$ is the partial derivative of position error in the vertical direction with respect to the pseudorange error on the ith satellite.

In the case of classical weighted least squares position solution, the projection matrix S is defined as:

$$S = \begin{bmatrix} s_{east,1} & s_{east,2} & \cdots & s_{east,N} \\ s_{north,1} & s_{north,2} & \cdots & s_{north,N} \\ s_{U,1} & s_{U,2} & \cdots & s_{U,N} \\ s_{t,1} & s_{t,2} & \cdots & s_{t,N} \end{bmatrix} = (G^T . W . G)^{-1} . G^T . W$$

<div align="right">(9-11)</div>

where

- the ith row of the geometry matrix **G** is defined as follows:

$$G_i = [-cos\,El_i.\sin Az_i \quad -cos\,El_i.\cos Az_i \quad -\sin El_i \quad 1]$$

when positive azimuth is defined clockwise from North.

- $$W = \begin{bmatrix} w_1 & 0 & \cdots & 0 \\ 0 & w_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & w_N \end{bmatrix} \text{ and } w_i = 1/\sigma_i^2$$

9.2 GALILEO GiC

9.2.1 General concept

The chosen method for providing integrity to GALILEO users is to monitor each satellite of the GALILEO constellation thanks to the data collected by the GALILEO Sensor Station (GSS) network, and to transmit the results to the users in the navigation data message, thus providing information on each satellite health.

The GALILEO Integrity Concept (GiC) consists in fact in providing three integrity parameters to the users of the SoL service so as to provide them the capability to monitor the integrity of their position solution. The three parameters are the following:

- Signal-In-Space Accuracy (SISA)
- Signal-In-Space Monitoring Accuracy (SISMA)
- Integrity Flag (IF)

To understand what these quantities represent, it is necessary to introduce the Signal-In-Space Error (SISE). SISE is the equivalent range error due to the difference between the true state vector of a satellite and its prediction derived from the navigation message. It is the maximum error of the SIS in the range domain caused by the satellite, the satellite payload and the navigation message (ie. Ephemeris data, clock correction …)

Knowing the position of the GSSs, the position of the GALILEO satellites and the pseudorange measurement error, the SISE can be estimated. We can now explain the three integrity parameters previously mentioned.

SISA: As the SISE distribution is not a Gaussian distribution, the methodology of over-bounding is applied to describe it. An over-bounding Gaussian distribution is thus defined which is characterized by its standard deviation SISA. SISA is defined as being the prediction of the minimum standard deviation of a Gaussian distribution that over-bounds the SISE distribution for a fault-free SIS.

SISMA: As the SISE cannot be measured directly, it is necessary to estimate it using measurements. The estimation of SISE results in an estimated SISE called eSISE. The difference between SISE and eSISE can be characterized by a distribution which does not have a Gaussian shape. This distribution shall be over-bounded by a Gaussian distribution defined by its standard deviation SISMA.

IF: the Integrity flag (IF) is an indicator of the satellite state, set to "OK" if the satellite is detected fault-free and set to "NOK" if the satellite is detected as malfunctioning (ie. if the maximum eSISE for

a SIS is larger than the Integrity Flag Threshold for this SIS). Finally this flag can be set to "NOT MONITORED" if there are not enough measurements available to perform the necessary tests. The Integrity Flag Threshold can be computed using the eSISE distribution, the distribution of the difference between SISE and eSISE and the desired false alarm probability.

9.2.2   GALILEO GiC user integrity algorithm

9.2.2.1   *Distribution of SISE and eSISE*
This section presents the algorithm implemented in our software on the basis of [ESA, 2005] and [PAIMBLANC, 2006].

- In the fault-free case:

The $SISE$ is assumed to follow a zero-mean normal distribution with standard deviation $SISA$:

$$SISE \sim N(0, SISA)$$

(9-14)

Its probability density function is expressed as:

$$p_x(x) = \frac{1}{\sqrt{2\pi}.SISA} e^{-\frac{1}{2}\left(\frac{x}{SISA}\right)^2}$$

(9-15)

We also have the estimation of $SISE$ by the ground segment which is assumed to follow a normal distribution with mean $SISE$ and standard deviation $SISMA$:

$$eSISE \sim N(SISE, SISMA)$$

The equivalent distribution function of $SISE - eSISE$ is a zero-mean Gaussian with standard deviation $SISMA$ and its probability density function is expressed as:

$$p_y(x, y) = p_y(SISE, y) = \frac{1}{\sqrt{2\pi}.SISMA} e^{-\frac{1}{2}\left(\frac{y-SISE}{SISMA}\right)^2}$$

(9-16)

Taking into account the previous $SISE$ distribution, we can express the $\widehat{SISE}$ probability density function as a mathematical convolution with the $SISE$ probability distribution.

$$p_y(y) = \int_{-\infty}^{\infty} p_x(x).p_y(x,y)dx = \int_{-\infty}^{\infty} p(x,y)\,dx$$

(9-17)

$$p_y(y) = \frac{1}{2\pi\sqrt{SISA^2 + SISMA^2}} e^{-\frac{1}{2}\left(\frac{y}{\sqrt{SISA^2+SISMA^2}}\right)^2}$$

(9-18)

Thus, the *eSISE* is zero-mean Gaussian distributed with standard deviation $\sqrt{SISA^2 + SISMA^2}$.

$$eSISE \sim N(0, \sqrt{SISA^2 + SISMA^2})$$

(9-19)

9.2.2.2   *Determination of the Integrity Flag Threshold*

First, we have to compute the Probability of False Alarm P_{FA} which is the probability that the estimated *SISE* is larger than the threshold while the true *SISE* is smaller than the threshold.

$$P_{FA} = P(eSISE > TH | SISE < TH)$$

(9-20)

It leads to the following result:

$$P_{FA} = \int_{-TH}^{+TH} \frac{1}{\sqrt{2\pi}.SISA} e^{-\frac{1}{2}\left(\frac{x}{SISA}\right)^2} dx - \int_{-TH}^{+TH} \left(\int_{-TH}^{+TH} \frac{1}{2\pi.SISA.SISMA} e^{-\frac{1}{2}\left(\left(\frac{x}{SISA}\right)^2 + \left(\frac{y-x}{SISMA}\right)^2\right)} dx \right) dy$$

(9-21)

To determine the threshold TH for a given P_{FA} with equation (9-21) is quite complicated because TH can only be derived iteratively. Therefore, following simplification is used:

$$P_{FA} = P(eSISE > TH | SISE < TH) \leq P(eSISE > TH)$$

(9-22)

And,

$$P(eSISE > TH) = 2 \int_{TH}^{\infty} \frac{1}{2\pi\sqrt{SISA^2 + SISMA^2}} e^{-\frac{1}{2}\left(\frac{y}{\sqrt{SISA^2+SISMA^2}}\right)^2} dy$$

(9-23)

A value $k_{P_{FA}}$ can be defined by its relation with the false alert probability P_{FA} as:

$$P_{FA} = 2 \int_{k_{P_{FA}}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}x^2} dx$$

(9-24)

The threshold can be expressed as:

$$TH = k_{P_{FA}}.\sqrt{SISA^2 + SISMA^2}$$

(9-25)

The value $k_{P_{FA}}$ can thus be calculated easily.

9.2.2.3   *Vertical direction*

In the following sections we will use the results below:

$$P(|x| > L) = \frac{1}{2}\left(1 - \text{erf}\left(\frac{L+\mu}{\sqrt{2}\sigma}\right)\right) + \frac{1}{2}\left(1 - \text{erf}\left(\frac{L-\mu}{\sqrt{2}\sigma}\right)\right)$$

(9-26)

- $x \sim N(0, \sigma)$
- $\text{erf}(u) = \frac{2}{\sqrt{\pi}}.\int_0^u e^{-x^2}dx$

9.2.2.3.1   Fault-free mode

In this case, the variance-covariance matrix for the measurement noise is the following:

$$C_{FF} = \begin{bmatrix} \sigma_{u,1,T}{}^2 & 0 & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \sigma_{u,i,T}{}^2 & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & \sigma_{u,N,T}{}^2 \end{bmatrix}$$

(9-27)

Where:

- $\sigma_{u,1,T} = \sqrt{SISA_i^2 + \sigma_{u,i,p}^2}$
- $SISA_i$ is the value of $SISA$ for satellite i
- $\sigma_{u,i,p}$ accounts for residual propagation errors and measurement noise on the user-satellite i direction

Thus, the variance-covariance matrix of the user's state vector in the local NED is, according to least squares algorithm:

$$\Sigma_{FF} = [G^T C_{FF} G]^{-1}$$

(9-28)

Where

- G is the design matrix H expressed in the NED reference frame.

208

The vertical positioning error in fault-free mode is then:

$$\epsilon_{V,FF} \sim N(0, \sigma_{V,FF}) \text{ where } \sigma_{V,FF} = \Sigma_{FF}(3,3)$$

(9-29)

The integrity risk due to a vertical positioning error in fault-free mode is therefore the probability for $\epsilon_{V,FF}$ to be superior to the VAL:

$$P_{IR,V,FF} = P(|\epsilon_{V,FF}| > VAL)$$

(9-30)

$$P_{IR,V,FF} = 1 - \text{erf}\left(\frac{VAL}{\sqrt{2}.\sigma_{V,FF}}\right)$$

(9-31)

9.2.2.3.2   Faulty mode

In faulty mode, we consider that a satellite j is biased. The bias b_j considered is the decision threshold TH_j:

$$b_j = k_{P_{FA}}.\sqrt{SISA_j^2 + SISMA_j^2}$$

(9-32)

Then, since the *SISE* estimation process is noisy, the estimated *SISE* of satellite j must take into account the *SISMA*. Then:

$$eSISE_j \sim N(b_j, SISMA_j)$$

(9-33)

Therefore, the variance-covariance in "faulty" mode will be expressed as:

$$C_{FM} = \begin{bmatrix} SISA_1^2 + \sigma_{u,i,p}^2 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & & & & & \vdots \\ \vdots & & SISA_{j-1}^2 + \sigma_{u,j-1,p}^2 & & & & \vdots \\ \vdots & & & SISMA_j^2 + \sigma_{u,j,p}^2 & & & \vdots \\ \vdots & & & & SISA_{j+1}^2 + \sigma_{u,j+1,p}^2 & & \vdots \\ \vdots & & & & & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots\cdots\cdots\cdots & 0 & SISA_L^2 + \sigma_{u,L,p}^2 \end{bmatrix}$$

(9-34)

Thus, the vertical error in the "faulty" mode has a non-zero mean $\mu_{V,j}$, which with the following notation:

$$M = [G^T.C_{FM}{}^{-1}.G]^{-1}.G^T.C_{FM}{}^{-1}$$

209

<div align="right">**(9-35)**</div>

Can be expressed as:

$$\mu_{V,j} = m_{3,j}.b_j$$

<div align="right">**(9-36)**</div>

Where $m_{3,j}$ is the element of the third line and j^{th} column of M.

Then $\sigma_{V,j,FM} = \Sigma_{Fm}(3,3)$, where $\Sigma_{Fm} = \left[G^T.C_{FM}^{-1}.G\right]^{-1}$

<div align="right">**(9-37)**</div>

Thus, the vertical positioning error in the case of a "faulty" satellite j is:

$$\epsilon_{V,j,FM} \sim N(\mu_{V,j}, \sigma_{V,j,FM})$$

<div align="right">**(9-38)**</div>

Consequently, the vertical integrity risk due to a fault on satellite j is:

$$P_{IR,V,j,FM} = P\left(\left|\epsilon_{V,j,FM}\right| > VAL\right)$$

<div align="right">**(9-39)**</div>

$$P_{IR,V,FM} = \frac{1}{2}\left(1 - \text{erf}\left(\frac{VAL + \mu_{V,j}}{\sqrt{2}.\sigma_{V,j,FM}}\right)\right) + \frac{1}{2}\left(1 - \text{erf}\left(\frac{VAL - \mu_{V,j}}{\sqrt{2}.\sigma_{V,j,FM}}\right)\right)$$

<div align="right">**(9-40)**</div>

9.2.2.3.3   Vertical integrity risk

If we assume that all satellites failures are uncorrelated, we obtain the global vertical integrity risk as following:

$$P_{IR,V} = P(Fault - free\ mode).P_{IR,V,FF} + \sum_{j=1}^{N} P(satellite\ j\ failure).P_{IR,V,FM}$$

<div align="right">**(9-41)**</div>

This expression can be simplified under the approximation:

$$P(Fault - free\ mode) \cong 1$$

<div align="right">**(9-42)**</div>

We get:

$$P_{IR,V} = P_{IR,V,FF} + \sum_{j=1}^{N} P_{SatFail,j} \cdot P_{IR,V,FM}$$

(9-43)

Where $P(satellite\ j\ failure) = P_{SatFail,j}$

9.2.2.4   Horizontal plane

9.2.2.4.1   Prerequisite
The error on the horizontal plane is the quadratic sum of the north and east errors:

$$\epsilon_H = \sqrt{\epsilon_N{}^2 + \epsilon_E{}^2}$$

(9-44)

Let x be a two-dimensional Gaussian random vector:

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, C = \begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{pmatrix} \right)$$

(9-45)

Then, the weighted quadratic form $x^T.C^{-1}.x$ is χ^2-distributed with two degrees of freedom. If $\mu_1 = \mu_2 = 0$ it is centered, otherwise it has a non-centrality parameter $\delta = (\mu_1 \quad \mu_2).C^{-1}.\begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}$.

In the central case,

$$P(x^T.C^{-1}.x > L^2) = 1 - S_{\chi_2^2}(L^2) = e^{-\frac{L^2}{2}}$$

(9-46)

And, in the non-central case:

$$P(x^T.C^{-1}.x > L^2) = 1 - S_{\chi_{2,\delta}^2}(L^2)$$

(9-47)

To find an equivalent formula for $x^T.x$ it is necessary to make an approximation: the probabilities expressed in the two previous expressions represent the probability for the quadratic form to be outside the error ellipse defined by C. However, in the case of the alert limit HAL, only one limit value is required, and it is thus possible to consider a circular error. The semi-major axis of the ellipse is a worst bound for the radius of the circular error. Its expression is:

$$\xi^2 = \frac{\sigma_1^2 + \sigma_2^2}{2} + \sqrt{\left(\frac{\sigma_1^2 - \sigma_2^2}{2} \right)^2 + \sigma_{12}{}^2}$$

(9-48)

211

And the corresponding variance-covariance matrix is:

$$Q = \begin{pmatrix} \xi^2 & 0 \\ 0 & \xi^2 \end{pmatrix}$$

(9-49)

Then, $x^T . C^{-1} . x$ simplifies to $\frac{x^T . x}{\xi^2}$. Thus, the probability for $\sqrt{x^T . x}$ to be superior to a given value L is, in the central case:

$$P\left(\sqrt{x^T . x} > L\right) = P\left(\frac{x^T . x}{\xi^2} > \frac{L^2}{\xi^2}\right)$$

(9-50)

$$P\left(\sqrt{x^T . x} > L\right) = 1 - S_{\chi_2^2}\left(\frac{L^2}{\xi^2}\right) = e^{-\frac{L^2}{2\xi^2}}$$

(9-51)

And in the non-central case:

$$P\left(\sqrt{x^T . x} > L\right) = 1 - S_{\chi_{2,\delta}^2}\left(\frac{L^2}{\xi^2}\right)$$

(9-52)

9.2.2.4.2   Fault-free mode

We can use the previous results to derive the horizontal error. In the fault-free case, the variance-covariance matrix of the north and east component is:

$$\begin{pmatrix} \sigma_{n,FF}^2 & \sigma_{ne,FF} \\ \sigma_{ne,FF} & \sigma_{e,FF}^2 \end{pmatrix} = \Sigma_{FF}(1:2,1:2)$$

(9-53)

Where Σ_{FF} is defined in section 9.2.2.3.1.

Thus, the squared semi-major axis is:

$$\xi_{FF}^2 = \frac{\sigma_{n,FF}^2 + \sigma_{e,FF}^2}{2} + \sqrt{\left(\frac{\sigma_{n,FF}^2 - \sigma_{e,FF}^2}{2}\right)^2 + \sigma_{ne,FF}^2}$$

(9-54)

And the variance-covariance (under the conservative assumption of a circular shape):

$$Q_{FF} = \begin{pmatrix} \xi_{FF}^2 & 0 \\ 0 & \xi_{FF}^2 \end{pmatrix}$$

(9-55)

Finally, the horizontal integrity risk in the fault-free case is:

$$P_{IR,H,FF} = P(\epsilon_H > HAL) \leq P\left(\frac{\epsilon_n{}^2 + \epsilon_e{}^2}{\xi_{FF}^2} > \frac{HAL^2}{\xi_{FF}^2}\right) = 1 - S_{\chi_2^2}\left(\frac{HAL^2}{\xi_{FF}^2}\right) = e^{-\frac{HAL^2}{2\xi_{FF}^2}}$$

(9-56)

9.2.2.4.3 Faulty mode

In the faulty mode, the two horizontal error components are assumed to be normally distributed with a non-zero mean. They can be expressed:

$$\begin{pmatrix}\epsilon_{n,j,FM} \\ \epsilon_{e,j,FM}\end{pmatrix} \sim N\left(\begin{pmatrix}\mu_{n,j} \\ \mu_{e,j}\end{pmatrix}, C_{j,FM}\right)$$

(9-57)

Where j is the index of the satellite assumed to fail.

As for the vertical direction, the positioning bias $\mu_{n,j}$ and $\mu_{e,j}$ are simply the projections of the measurement bias b_j by the matrix $M = [G^T.C_{FM}{}^{-1}.G]^{-1}.G^T.C_{FM}{}^{-1}$:

$$\mu_{n,j} = m_{1,j}.b_j$$

$$\mu_{e,j} = m_{2,j}.b_j$$

(9-58)

Then, the same conservative assumption as in the fault-free case is done. With the variance-covariance matrix expressed as follows:

$$C_{j,FM} = \begin{pmatrix}\sigma_{n,j,FM}^2 & \sigma_{ne,j,FM} \\ \sigma_{ne,j,FM} & \sigma_{e,j,FM}^2\end{pmatrix}$$

(9-59)

The squared semi-major axis is the following:

$$\xi_{j,FM}^2 = \frac{\sigma_{n,j,FM}^2 + \sigma_{e,j,FM}^2}{2} + \sqrt{\left(\frac{\sigma_{n,j,FM}^2 - \sigma_{e,j,FM}^2}{2}\right)^2 + \sigma_{ne,j,FM}^2}$$

(9-60)

And the corresponding variance-covariance matrix of a circular shaped error is:

$$Q_{j,FM} = \begin{pmatrix}\xi_{j,FM}^2 & 0 \\ 0 & \xi_{j,FM}^2\end{pmatrix}$$

(9-61)

In the faulty-case, we have a non-centrality parameter δ_j:

$$\delta_j = (\mu_{n,j} \quad \mu_{e,j}).C_{j,FM}^{-1}.\begin{pmatrix}\mu_{n,j} \\ \mu_{e,j}\end{pmatrix}$$

<div align="right">**(9-62)**</div>

Thus, in the faulty mode, the horizontal integrity risk is:

$$P_{IR,H,j,FM} = P(\epsilon_H > HAL) \leq P\left(\frac{\epsilon_n^{\ 2} + \epsilon_e^{\ 2}}{\xi_{j,FM}^2} > \frac{HAL^2}{\xi_{j,FM}^2}\Big|\delta_j\right) = 1 - S_{\chi_{2,\delta_j}^2}\left(\frac{HAL^2}{\xi_{FM}^2}\right)$$

<div align="right">**(9-63)**</div>

9.2.2.4.4   Horizontal integrity risk

If we assume that all satellites failures are uncorrelated, we obtain the global vertical integrity risk as following:

$$P_{IR,H} = P(Fault - free\ mode).P_{IR,H,FF} + \sum_{j=1}^{N} P(satellite\ j\ failure).P_{IR,H,j,FM}$$

<div align="right">**(9-64)**</div>

With the same assumptions as for vertical integrity risk we get:

$$P_{IR,H} = P_{IR,H,FF} + \sum_{j=1}^{N} P_{SatFail,j} \cdot P_{IR,H,j,FM}$$

<div align="right">**(9-65)**</div>

9.2.2.5   *User global integrity risk*

The total integrity risk at the alert limits HAL and VAL is the sum of the vertical and horizontal contributions:

$$\begin{aligned} P_{IR}(HAL,VAL) &= P_{IR,V} + P_{IR,H} \\ &= 1 - \text{erf}\left(\frac{VAL}{\sqrt{2}.\sigma_{V,FF}}\right) + e^{-\frac{HAL^2}{2\xi_{FF}^2}} \\ &+ \frac{1}{2}\sum_{j=1}^{N} P_{SatFail,j} \cdot \left(\left(1 - \text{erf}\left(\frac{VAL + \mu_{V,j}}{\sqrt{2}.\sigma_{V,j,FM}}\right)\right) + \left(1 - \text{erf}\left(\frac{VAL - \mu_{V,j}}{\sqrt{2}.\sigma_{V,j,FM}}\right)\right)\right) \\ &+ \sum_{j=1}^{N} P_{SatFail,j} \cdot \left(1 - S_{\chi_{2,\delta_j}^2}\left(\frac{HAL^2}{\xi_{FM}^2}\right)\right) \end{aligned}$$

<div align="right">**(9-66)**</div>

9.2.2.6   *Protection levels computation*

To compute protection levels using the GALILEO Integrity Concept it is necessary to allocate partial components of the total integrity risk to the horizontal plane and to the vertical plane.

The allocated horizontal integrity risk $P_{HMI,H,alloc}(VAL, HAL)$ and vertical integrity risk $P_{HMI,V,alloc}(VAL, HAL)$ are computed according to the following formulas:

$$P_{HMI,H,alloc}(VAL, HAL) = P_{HMI,alloc}(VAL, HAL).\frac{P_{IR,H}}{P_{IR,V} + P_{IR,H}}$$

(9-67)

$$P_{HMI,V,alloc}(VAL, HAL) = P_{HMI,alloc}(VAL, HAL).\frac{P_{IR,V}}{P_{IR,V} + P_{IR,H}}$$

(9-68)

Where the allocated total integrity risk at the alert limits is $P_{HMI,alloc}(VAL, HAL)$.

The protection levels HPL and VPL can be defined as the horizontal and vertical spatial limits, where the HMI probability is exactly the allocated integrity risk.

$$P_{HMI,H,alloc}(VAL, HAL) = e^{-\frac{HPL^2}{2\xi_{FF}^2}} + \sum_{j=1}^{N} P_{SatFail,j} \cdot \left(1 - S_{\chi_{2,\delta_j}^2}\left(\frac{HPL^2}{\xi_{FM}^2}\right)\right) = f_H(HPL)$$

(9-69)

$$
\begin{aligned}
P_{HMI,V,alloc}&(VAL, HAL)\\
&= 1 - \text{erf}\left(\frac{VPL}{\sqrt{2}.\sigma_{V,FF}}\right)\\
&+ \frac{1}{2}\sum_{j=1}^{N} P_{SatFail,j} \cdot \left(\left(1 - \text{erf}\left(\frac{VPL + \mu_{V,j}}{\sqrt{2}.\sigma_{V,j,FM}}\right)\right) + \left(1 - \text{erf}\left(\frac{VPL - \mu_{V,j}}{\sqrt{2}.\sigma_{V,j,FM}}\right)\right)\right)\\
&= f_V(VPL)
\end{aligned}
$$

(9-70)

The protection levels can be described using the inverse functions:

$$HPL = f_H^{-1}\left(P_{HMI,H,alloc}(VAL, HAL)\right)$$

(9-71)

$$VPL = f_V^{-1}\left(P_{HMI,V,alloc}(VAL, HAL)\right)$$

(9-72)

It is impossible to resolve the inverse functions f_H^{-1} and f_V^{-1} analytically, that's why an iterative method is used to compute the Protection levels HPL and VPL.

In our simulator, we have used the dichotomy technique to compute these protection levels.

9.2.3 Parameters simulation

As we have seen in the two previous sections, to use the GALILEO Integrity Concept user must receive three integrity parameters:

- Signal-In-Space Accuracy (SISA)
- Integrity Flag (IF)
- Signal-In-Space Monitoring Accuracy (SISMA)

For our simulator we need to model the parameters SISA and SISMA.

9.2.3.1 SISA simulation

It is specified in [ESA, 2005] that the overall required integrity and availability performances can be met in the indicated conditions and in particular if $SISA \leq 85\ cm$. In our simulator we have made the conservative choice to take $SISA = 85\ cm$ for all satellites.

This specification is only available for SoL service.

9.2.3.2 SISMA computation

The methodology used here is based on [PAIMBLANC, 2006].

The inpust used by the ground integrity monitoring algorithm are pseudorange residuals $\Delta Y_{i,j}$ where i is the satellite index and j the GSS index:

$$\Delta Y_{i,j} = \rho_{i,j}^{pp} - \rho_{i,j}^{pred}$$

(9-73)

Where:

- $\rho_{i,j}^{pp}$ is the preprocessed pseudorange measurement performed by GSS j on satellite i. These pseudorange measurements have been corrected from most propagation issues and thus remaining errors are mostly thermal noise and satellite clock and ephemeris error.
- $\rho_{i,j}^{pred}$ the corresponding pseudorange prediction computed from ephemeris data at the GSS.

These pseudorange residuals are used to compute:

- $eSISE$, estimation of the Signal-In-Space Error
- $SISMA$
- IF

We define ΔY the vector containing all the pseudorange residuals corresponding to one satellite of the constellation. The aim of the algorithm is to estimate the ΔX vector, which is the difference between the satellite true state vector X and its estimation by the Orbit Determination and Time Synchronisation (OD&TS). Let the Worst User Location (WUL) be the position for which the projection of ΔX on the user satellite axis is maximum: the $SISE$ is the range error obtained by projecting ΔX on the WUL-satellite axis.

We remind that:

$$SISE \sim N(0, SISA)$$

(9-74)

The estimation of ΔX is an erroneous process. The resulting $\widehat{\Delta X}$ can be described as a Gaussian random vector centered on ΔX, with a covariance matrix C:

$$\widehat{\Delta X} \sim N(\Delta X, C)$$

(9-75)

The estimated SISE is defined as the projection of $\widehat{\Delta X}$ on the WUL axis. Since the projection is a linear process, $eSISE$ can be expressed as the sum of $SISE$ and the projection of the positioning error due to measurement noise ϵ_{meas}.

Let X_s be the true satellite state vector and X_0 its estimation by the OD&TS, expressed in the ECEF referential. The first step of the Integrity core algorithm is to determine:

$$\Delta X = X_s - X_0$$

(9-76)

Let Y be the vector of preprocessed pseudorange measurements, and Y_0 the vector of predicted pseudoranges based on OD&TS data. Then, the pseudorange residuals vector is defined as follows:

$$\Delta Y = Y - Y_0$$

(9-77)

Then, the relation between ΔY and ΔX is:

$$\Delta Y = H.\Delta X + B$$

(9-78)

Where:

- H is the classical observation matrix
- B is the measurement noise vector

Let C be the covariance matrix of the measurement noise. Its coefficients are based on a noise model which mainly depends on the elevation angle of the satellite as seen by the GSS:

$$C = \begin{bmatrix} \sigma_1{}^2 & & 0 \\ & \ddots & \\ 0 & & \sigma_N{}^2 \end{bmatrix}$$

(9-79)

Where N is the number of available pseudorange residuals.

Thus the estimated ΔX is:

$$\widehat{\Delta X} = [H^T C^{-1} H]^{-1} H^T C^{-1} \Delta Y$$

(9-80)

The covariance of $\widehat{\Delta X}$ is computed as:

$$cov(\widehat{\Delta X}) = [H^T C^{-1} H]^{-1}$$

(9-81)

The *SISE* is defined as the projection of ΔX on the axis formed by the WUL and the predicted position of the satellite. The WUL is located on the limit of the availability area, which corresponds to the lowest elevation angle [PAIMBLANC, 2006]. The WUL is thus determined by sampling the boundary of the availability area. The resulting set of user positions is used to compute a projection matrix H_u. The projection is performed as follows:

$$\begin{bmatrix} eSISE_1 \\ \vdots \\ eSISE_N \end{bmatrix} = H_u.\widehat{\Delta X}$$

(9-82)

Let n be the maximum of the projected vector in ansolute value:

$$|eSISE_n| = max(|H_u.\widehat{\Delta X}|)$$

(9-83)

Then,

$$eSISE = eSISE_n$$

(9-1)

Let h_u be the corresponding line of the H_u matrix, then *SISMA* is defined as:

$$SISMA = h_u.cov(\widehat{\Delta X}).h_u{}^T$$

(9-84)

Appendix B: Interferences model

Some models have been implemented in the receiver simulator so as to be able to represent the impact of various types of interferences on the simulator outputs. This section details these models.

10.1 Types of interferences

Three main different types of unintentional interferences which can threat GNSS signals in ARNS bands have to be mentioned:

- Carrier Wave interferences in L1 and L5 bands

- Continuous waves interferences that can be Narrowband or Wideband interferences in L1 and L5 bands

- Pulsed interferences in L5 bands

Carrier Wave interference is a sinusoidal waveform with a high power which can be harmful to GNSS receivers when it is located close to the central frequency of the GNSS signals. It can be expressed a:

$$CW(t) = \sqrt{P_{CW}}.e^{j(2\pi f_c t + \theta)}$$

(10-1)

With

- P_{CW} the power of the Carrier Wave
- f_c the central frequency of the Carrier Wave
- θ the phase of the Carrier Wave

Wideband interferences are modelled as white noise with limited bandwidth.

Pulsed interferences are provoked by equipments that are already radiating in L5 band and which are DME/TACAN on E5a and Radars on E5b for example. This type of interference has not been modelled in the receiver simulator and will thus not be further described.

10.2 Initial assumptions

It has been demonstrated that up to a certain power level and for large jammer bandwidth, interferences lead to a degradation of the signal to noise ratio C/N_0. Thus, these interferences can be modeled as additional thermal noise sources and their impact can thus be described by an increase of the thermal noise power spectral density. We remind here and in the following sections the computation of the equivalent signal to noise ratio in presence of interference within the interference mask. The complete demonstration can be found in [BASTIDE, 2001].

We introduce here the Spectral Separation Coefficient (SCC) between the useful signal x and the considered interference i:

$$C_{int}(f) = \int_B S_x(u)S_i(u+f)\,du$$

(10-2)

Where

222

- S_x represents the normalized power spectrum of the signal
- S_i represents the normalized power spectrum of the interference

For a better understanding we remind hereafter the simplified architecture of a classic GNSS receiver.
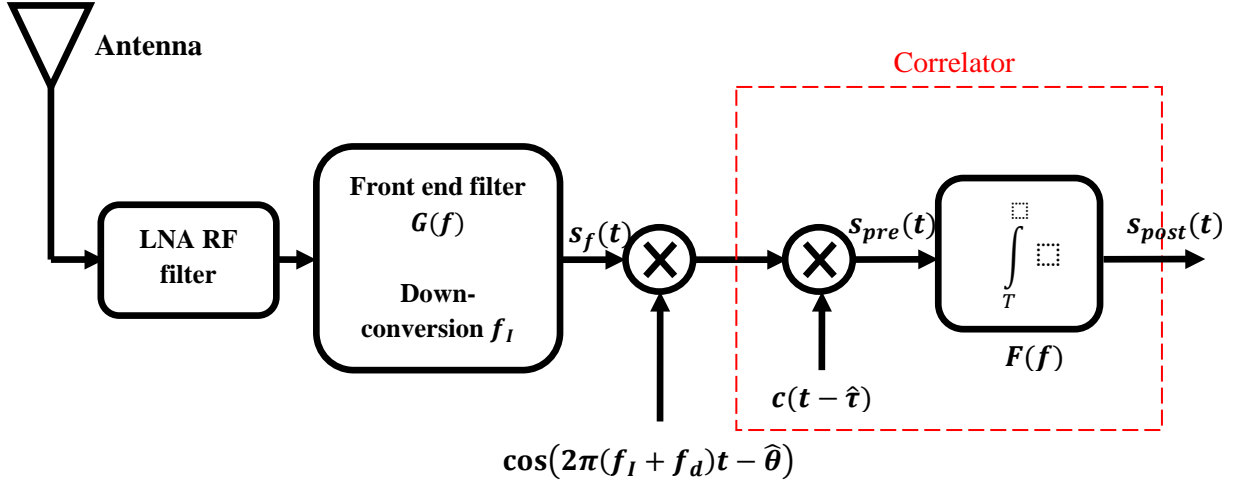
Figure 102: Classical receiver architecture

The following quantities are placed on this scheme:

- $G(f)$ the front-end filter transfer function
- f_I the intermediate frequency for down-conversion
- $s_f(t)$ the filtered and down-converted signal
- $s_{pre}(t)$ the pre-correlation signal
- $s_{post}(t)$ the post-correlation signal
- T the integration time
- $F(f)$ the integrate and dump filter transfer function
- $c(t)$ the code sequence waveform
- $\cos(2\pi f t)$ the carrier waveform
- $\hat{\tau}$ the estimated code phase
- $\hat{\theta}$ the estimated carrier phase

The idea now is to compute the power of each component of the post-correlation signal in presence of noise and interference only.

10.3 Post correlation signal power

We can express the pre-correlation signal $s_{pre}(t)$ as:

$$s_{pre}(t) = s_f(t) \cos\big(2\pi(f_I + f_d)t - \hat{\theta}\big) C(t - \hat{\tau})$$

(10-3)

We can therefore deduce its power spectral density:

$$S_{s_{pre}}(f) = S_{s_f}(f) * \frac{1}{4}\Big(\delta\big(f - (f_I + f_d)\big) + \delta\big(f + (f_I + f_d)\big)\Big) * S_c(f)$$

(10-4)

223

Thus we obtain:

$$S_{S_{pre}}(f) = \frac{1}{4}\left(S_{S_f}(f - (f_I + f_d)) + S_{S_f}(f + (f_I + f_d))\right) * S_c(f)$$

(10-5)

Wiener Lee relation allows expressing the post-correlation signal as a function of the pre-correlation signal and the I&D filter transfer function:

$$S_{S_{post}}(f) = S_{S_{pre}}(f).|F(f)|^2$$

(10-6)

The post-correlation signal power is given by:

$$P_{S_{post}} = \int_{-\infty}^{+\infty} S_{S_{pre}}(f).|F(f)|^2.df$$

(10-7)

If we consider that $F(f)$ is a narrow band filter, two distinct cases have to be taken into account. If $S_{S_{pre}}(f)$ is very large with respect to $F(f)$, $P_{S_{post}}$ can be approximated by:

$$P_{S_{post}} = S_{S_{pre}}(0) \int_{-\infty}^{+\infty} |F(f)|^2.df$$

(10-8)

If it is not the case, no further development of eq can be made

10.4 Post correlation thermal noise power

We consider that the thermal noise $n(t)$ added to the useful signal is white and has a power density N_0. After the front-end filtering, the power spectral density of the noise is:

$$S_{n_f} = \frac{N_0}{2}|G_{BB}(f)|^2 * \left(\delta(f - f_I) + \delta(f + f_I)\right)$$

(10-9)

With $G_{BB}(f)$ the equivalent baseband filter of $G(f)$.

We can then express the noise power spectral density before the correlation process as:

$$S_{n_{pre}}(f) = \frac{N_0}{2}|G_{BB}(f)|^2 * \left(\delta(f - f_I) + \delta(f + f_I)\right) * \frac{1}{4}\left(\delta(f - (f_I + f_d)) + \delta(f + (f_I + f_d))\right) \\ * S_c(f)$$

(10-10)

If we develop this expression we obtain:

224

$$S_{n_{pre}}(f) = \frac{1}{4} \cdot \frac{N_0}{2} |G_{BB}(f)|^2 * \left(\delta(f - 2f_I - f_d) + \delta(f + 2f_I + f_d) + \delta(f - f_d) + \delta(f + f_d) \right)$$
$$* S_c(f)$$

(10-11)

If we compute the value of this PSD in 0 we have:

$$S_{n_{pre}}(0) = \frac{1}{4} \frac{N_0}{2} \int_{-\infty}^{+\infty} \left(|G_{BB}(f)|^2 * \left(\delta(f - 2f_I - f_d) + \delta(f + 2f_I + f_d) + \delta(f - f_d) + \delta(f + f_d) \right) \right) S_c(f)\, df$$

(10-12)

Considering that $S_c(f)$ is band-limited and that $G_{BB}(f)$ is very large we can simplify the expression:

$$S_{n_{pre}}(0) = \frac{1}{4} \frac{N_0}{2} 2 \int_{-\infty}^{+\infty} |G_{BB}(f)|^2 . S_c(f)\, df$$

(10-13)

This expression can be simplified as:

$$S_{n_{pre}}(0) = \frac{N_0}{4} \int_{-B/2}^{+B/2} S_c(f)\, df$$

(10-14)

$$S_{n_{pre}}(0) \approx \frac{N_0}{4} K_C(0) = \frac{N_0}{4}$$

(10-15)

With $K_C(\tau)$ the autocorrelation function of the code sequence.

Using Wiener-Lee relation as previously and Parseval relations:

$$\int_{-\infty}^{+\infty} |F(f)|^2\, df = \int_{-\infty}^{+\infty} |f(t)|^2\, dt = T$$

(10-16)

We can deduce the final Thermal noise post correlation power:

$$P_{n_{post}} = \frac{N_0}{4} T$$

(10-17)

As we can see here, the white noise PSD can be computed by multiplying the post correlation power by $\frac{4}{T}$. The method exposed here can be used to find the equivalent noise PSD in presence of white noise and interference as we will see in further sections.

10.5 Post correlation interference power

10.5.1 General expression
We define here:

- $J(t)$ the received RF interference and $S_J(f)$ its PSD
- $J_f(t)$ the filtered RF interference and $S_{J_f}(f) = |G(f)|^2 S_J(f)$ its PSD

The pre correlation interference PSD is given by:

$$S_{J_{pre}}(f) = S_{J_f}(f) * \frac{1}{4}\left(\delta(f - (f_I + f_a)) + \delta(f + (f_I + f_a))\right) * S_c(f)$$

(10-18)

This expression results in:

$$S_{J_{pre}}(f) = \frac{1}{4}\left(S_{J_f}(f - (f_I + f_a)) + S_{J_f}(f + (f_I + f_a))\right) * S_c(f)$$

(10-19)

If we apply the same steps as in previous sections for useful signal and thermal noise, we finally obtain the following post-correlation interference power

$$P_{J_{post}} = \int_{-\infty}^{+\infty} S_{J_{pre}}(f)|F(f)|^2 . df$$

(10-20)

This expression can be developed by making assumptions on the characteristics of the considered interference.

10.5.2 Continuous Wave interferences that can be Narrowband or Wideband interferences
In this section we apply the general expression for the post-correlation interference power to the particular case of Narrowband and Wideband interferences. This type of interference is defined as interferences such that the bandwidth of the pre-correlation power spectral density of the interference $S_{J_{pre}}(f)$ is much larger than the bandwidth of the transfer function of the I&D filter $F(f)$. Thus we can simplify equation **(10-20)**:

$$P_{J_{post}} = S_{J_{pre}}(0) \int_{-\infty}^{+\infty} |F(f)|^2 . df$$

(10-21)

Using the Parseval relation we have already shown that:

$$\int_{-\infty}^{+\infty} |F(f)|^2 \, df = \int_{-\infty}^{+\infty} |f(t)|^2 \, dt = T$$

(10-22)

Moreover, we have already expressed $S_{J_{pre}}(f)$ which gives:

$$S_{J_{pre}}(0) = \frac{1}{4} \int_{-\infty}^{+\infty} \left(S_{J_f}(f - (f_I + f_d)) + S_{J_f}(f + (f_I + f_d)) \right) S_c(f) \, df$$

(10-23)

By introducing the filtered interference equivalent baseband signal PSD $S_{J_{f,BB}}$ it becomes:

$$S_{J_{pre}}(0) = \frac{1}{4} \int_{-\infty}^{+\infty} \left(\frac{1}{4} S_{J_{f,BB}}(f - (f_I + f_d) - f_J) + \frac{1}{4} S_{J_{f,BB}}(f) + \frac{1}{4} S_{J_{f,BB}}(f + (f_I + f_d) - f_J) \right.$$
$$\left. + \frac{1}{4} S_{J_{f,BB}}(f + (f_I + f_d) + f_J) \right) S_c(f) \, df$$

(10-24)

With f_J the interference central frequency

Considering that the local code sequence PSD $S_c(f)$ has a limited bandwidth, the highest frequency components are automatically eliminated and we can simplify the expression:

$$S_{J_{pre}}(0) = \frac{1}{16} \int_{-\infty}^{+\infty} \left(S_{J_{f,BB}}(f + \Delta f) + S_{J_{f,BB}}(f - \Delta f) \right) S_c(f) \, df$$

(10-25)

With $\Delta f = f_J - (f_I + f_d)$

Let's denote $S_{J_{BB}}$ the PSD of the unfiltered interference equivalent baseband signal:

$$S_{J_{f,BB}}(f) = S_{J_{BB}}(f) |G_{BB}(f)|^2$$

(10-26)

With this expression in equation **(10-25)** and considering that interference PSD, front end filter and code PSD are symmetric we can obtain:

$$S_{J_{pre}}(0) = \frac{1}{8} \int_{-\infty}^{+\infty} |G_{BB}(f)|^2 S_{J_{BB}}(f - \Delta f) S_c(f) \, df$$

(10-27)

Finally,

$$P_{J_{post}} = \frac{T}{8} \int_{-\infty}^{+\infty} |G_{BB}(f)|^2 S_{J_{BB}}(f - \Delta f) S_c(f) \, df$$

(10-28)

We have demonstrated with equation **(10-17)** that the thermal noise PSD can be obtained by multiplying the post-correlation thermal noise power by a factor $\frac{4}{T}$. If we apply the same reasoning to the interferences, we can deduce the PSD of an equivalent white noise that would produce the same impact as our interference:

$$N_{0,equ,J} = P_{J_{post}} \times \frac{4}{T}$$

(10-29)

$$N_{0,equ,J} = \frac{1}{2} \int\limits_{-\infty}^{+\infty} |G_{BB}(f)|^2 S_{J_{BB}}(f - \Delta f) S_c(f) \, df$$

(10-30)

If we denote $S_{J\,BB,norm}$ the normalized PSD of the interference, which means that:

$$\int\limits_{-\infty}^{+\infty} S_{J\,BB,norm} \cdot df = 1$$

(10-31)

And C_J the received interference power such that:

$$S_{J_{BB}}(f) = 2C_J S_{J\,BB,norm}(f)$$

(10-32)

Then,

$$N_{0,equ,J} = C_J \int\limits_{-\infty}^{+\infty} |G_{BB}(f)|^2 S_{J\,BB,norm}(f - \Delta f) S_c(f) \, df$$

(10-33)

We define the interference coefficient J_{coef} as follows:

$$J_{coef} = \int\limits_{-\infty}^{+\infty} |G_{BB}(f)|^2 S_{J\,BB,norm}(f - \Delta f) S_c(f) \, df$$

(10-34)

Then the corresponding white noise density is $N_{0,equ,J} = C_J \cdot J_{coef}$ and the global equivalent white noise density is:

$$N_{0,equ} = N_0 + C_J \cdot J_{coef}$$

(10-35)

Then, the equivalent C/N_0 degradation $Degrad$ is:

$$Degrad = \frac{C/N_{0,equ}}{C/N_0} = \frac{N_0}{N_{0,equ}} = \frac{1}{1 + \frac{C_J}{N_0} \cdot J_{coef}}$$

(10-36)

Or

$$Degrad = \frac{C/N_{0,equ}}{C/N_0} = -10.\log_{10}\left(1 + \frac{C_J}{N_0} \cdot coeff\right)$$

(10-37)

10.5.3 Carrier Wave interferences

In this section we apply the general expression for the post-correlation interference power to the particular case of Carrier Wave interferences. Carrier Wave interferences can be modelled as:

$$J(t) = \sqrt{2C_J}\cos(2\pi f_J t)$$

(10-38)

Its PSD after filtering is given by:

$$S_{J_f}(f) = \frac{C_J}{2}\left(\delta(f - f_J) + \delta(f + f_J)\right)$$

(10-39)

Two different cases have to be distinguished:

- The case of a local C/A code sequence
- The case of a long or non-periodic local code

10.5.3.1 Local C/A-like code and Carrier Wave interference

We assume here that the C/A-like code has a line spectrum each spectral line being spaced of $f_R = \frac{1}{T_R}$.

10.5.3.1.1 Equivalent noise PSD

The PSD of the pre correlation signal using the interference PSD can be expressed as:

$$S_{J_{pre}}(f) = \frac{C_J}{8}\Big(\delta(f - (f_I + f_d) - f_J) + \delta(f - (f_I + f_d) + f_J) + \delta(f + (f_I + f_d) - f_J) \\ + \delta(f + (f_I + f_d) + f_J)\Big) * S_C(f)$$

(10-40)

Since $F(f)$ has a narrow bandwidth, only the following terms of the pre correlation PSD will impact the correlator behaviour:

$$S_{J_{pre}}(f) = \frac{C_J}{8}\Big(\delta(f - (f_I + f_d) + f_J) + \delta(f + (f_I + f_d) - f_J)\Big) * S_C(f)$$

(10-41)

Using the same notations as previously we have:

$$S_{J_{pre}}(f) = \frac{C_J}{8}\left(\delta(f + \Delta f) + \delta(f - \Delta f)\right) * S_C(f)$$

(10-42)

And,

$$S_{J_{pre}}(f) = \frac{C_J}{8}\left(S_C(f + \Delta f) + S_C(f - \Delta f)\right)$$

(10-43)

Then the post correlation power is obtained by using wiener-lee relation and taking the post correlation PSD in **(10-20)**:

$$P_{J_{post}} = \frac{C_J}{8}\int_{-\infty}^{+\infty}\left(S_C(f + \Delta f) + S_C(f - \Delta f)\right).|F(f)|^2\,df$$

(10-44)

We know that the code PSD and the filter transfer function are symmetric:

$$P_{J_{post}} = \frac{C_J}{4}\int_{-\infty}^{+\infty} S_C(f - \Delta f).|F(f)|^2\,df$$

(10-45)

Considering that the C/A-like code has a line spectrum and that the pre detection filter has a narrow bandwidth, then one code PSD line can fall inside the main lobe of the squared modulus pre detection filter and thus have a significant effect.

In this case we have for a code PSD line located at frequency $k_0 f_R$ with $k_0 \in \mathbb{N}$

$$P_{J_{post}} = \frac{C_J}{4}S_C(k_0 f_R).|F(k_0 f_R + \Delta f)|^2$$

(10-46)

The expression of the squared modulus pre detection filter is well known:

$$F(f) = T.\frac{\sin(\pi f t)}{\pi f t}$$

(10-47)

We eventually get,

$$P_{J_{post}} = \frac{C_J T^2}{4}S_C(k_0 f_R).\left|\frac{\sin(\pi(k_0 f_R + \Delta f)t)}{\pi(k_0 f_R + \Delta f)t}\right|^2$$

(10-48)

By applying the $\frac{4}{T}$ factor we have the following equivalent noise power spectral density:

$$N_{0,equ,J} = C_J T . S_C(k_0 f_R) . \left| \frac{\sin(\pi(k_0 f_R + \Delta f)t)}{\pi(k_0 f_R + \Delta f)t} \right|^2$$

(10-49)

10.5.4 Long or non-periodic code and Carrier Wave interference

In this case, the local code PSD is continuous and we can write using equation **(10-20)**:

$$P_{J_{post}} = \frac{C_J}{4} S_C(\Delta f) \int_{-\infty}^{+\infty} |F(f)|^2 \, df$$

(10-50)

$$P_{J_{post}} = \frac{C_J T}{4} S_C(\Delta f)$$

(10-51)

And we obtain the equivalent noise power spectral density:

$$N_{0,equ,J} = C_J S_C(\Delta f)$$

(10-52)

10.6 Implementation of interference models in the receiver simulator

In the software receiver developed during the PhD not all the previous models have been implemented for the moment. These models may be completed to extend the simulation capacity of the simulator for future studies. Currently, the GNSS software simulator allows generating the impact of interferences at the correlator outputs levels which are:

- Carrier Waves

- Wide Band interferences

Appendix C: Validation of the IQ correlator outputs generator

One of the difficulties linked to the receiver simulator was to validate the results obtained with it so as to ensure its representativeness. It appeared that a global validation with all the different models would not be convenient and thus the validation of the whole simulator is mainly based on the validation of each sub model which composes it.

One of the most important of the receiver simulator is the IQ correlator outputs generator since it models the behaviour of the receiver tracking loops. The models of the IQ correlator outputs generator have been presented in Chapter 3 and 4. To simplify the development we decide to first develop a tuneable IQ correlator outputs generator including only one code tracking loop and one carrier tracking loop. This initial module thus only allowed tracking one GNSS signal. This strategy has allowed us to first validate the behaviour of this simplified version and then to duplicate the validated module to obtain our final IQ correlator outputs generator.

This section rapidly presents the simple simulation strategy used to validate our software.

11.1 Validation plan

In fact, to confront our results we disposed of two different references:

- Theoretical tracking loops error
- Previously developed software Matlab which has been used as a basis for the development of our software in C/C++.

Two different sources were used for the theoretical code and carrier tracking error standard deviation σ_τ and σ_θ. The first set of equations is what we called HOLMES formulas. These formulas are based on the assumption of an infinite RF bandwidth. They are simpler and provide a good approximation for the code tracking error and a precise estimation for the carrier tracking error since it is not impacted by front-end filter. HOLMES formulas can be found in [SPILKER, 1996]:

- Dot product discriminator:

$$\sigma_\tau = \sqrt{\frac{B_{DLL}.C_s}{2.C/N_0}\left(1+\frac{1}{C/N_0.T_{DLL}}\right)}$$

(11-1)

- EMLP discriminator:

$$\sigma_\tau = \sqrt{\frac{B_{DLL}.C_s}{2.C/N_0}\left(1+\frac{2}{(2-\Delta).C/N_0.T_{DLL}}\right)}$$

(11-2)

- COSTAS discriminator:

$$\sigma_\theta = \sqrt{\frac{B_{PLL}}{2.C/N_0}\left(1+\frac{1}{2.C/N_0.T_{PLL}}\right)}$$

(11-3)

With:

- B_{DLL} and B_{PLL} are respectively the filter bandwidth of the code and carrier tracking loops
- T_{DLL} and T_{PLL} are respectively the coherent integration time of the code and carrier loops
- $\frac{C}{N_0}$ is the signal-to-noise ratio of the processed signal
- C_s is the correlator chip spacing

The second formulas are the BETZ formulas and they take into account the RF bandwidth considering an ideal rectangular filter. It explains why they provide a much better estimate of the code tracking error standard deviation. These can be found in [JULIEN, 2005]:

- Dot Product discriminator:

$$\sigma_\tau = \sqrt{\frac{B_{DLL} \int_{-B/2}^{B/2} G(f) \sin^2(\pi f C_s) df}{\frac{C}{N_0} \left(2\pi \int_{-B/2}^{B/2} f G(f) \sin(\pi f C_s) df\right)^2} \left(1 + \frac{1}{\frac{C}{N_0} T_{DLL} \left(\int_{-B/2}^{B/2} G(f) df\right)}\right)}$$

(11-4)

- EMLP discriminator:

$$\sigma_\tau = \sqrt{\frac{B_{DLL} \int_{-B/2}^{B/2} G(f) \sin^2(\pi f C_s) df}{\frac{C}{N_0} \left(2\pi \int_{-B/2}^{B/2} f G(f) \sin(\pi f C_s) df\right)^2} \left(1 + \frac{\int_{-B/2}^{B/2} G(f) \cos^2(\pi f C_s) df}{\frac{C}{N_0} T_{DLL} \left(\int_{-B/2}^{B/2} G(f) \cos(\pi f C_s) df\right)^2}\right)}$$

(11-5)

With:

- B_{DLL} is the filter bandwidth of the code tracking loop
- T_{DLL} is the coherent integration time of the code tracking loop
- B is the RF front-end filter bandwidth
- $\frac{C}{N_0}$ is the signal-to-noise ratio of the processed signal
- C_s is the correlator chip spacing
- $G(f)$ is the power spectral density of the processed signal

The next section gathers the validation results obtained through simulation.

11.2 Validation results

To assess the validity of our correlator outputs generator we tested different signals and different configurations of the simulated tracking loops, modifying parameters which have a direct impact on the code and carrier tracking loops errors.

All simulations were performed over $100\ s$ of tracking, but only the last $95\ s$ were used to compute our results so as to eliminate any undesired effects linked to initialization. The results obtained are gathered in the following sections.

11.2.1 BPSK signals

The following table describes the simulation plan used for BPSK signals.

Parameter	Tested values
C/N_0	$[30, 35, 40, 45, 50]$
C_s (chips)	$[0.1, 0.2, 0.25, 0.5]$
B (MHz)	$[4, 10]$
T_{DLL} (s)	$[0.001, 0.020]$
T_{PLL} (s)	$[0.001, 0.020]$

Table 29: BPSK simulation plan

The obtained results are gathered in table 32, 33 and 34.

11.2.2 BOC signals

The following table describes the simulation plan used for BPSK signals.

Parameter	Tested values
C/N_0	$[30, 35, 40, 45, 50]$
C_s (chips)	$[0.21, 0.25]$
B (MHz)	$[4, 10]$
T_{DLL} (s)	$[0.001, 0.100]$
T_{PLL} (s)	$[0.001, 0.020]$

Table 30: BPSK simulation plan

The obtained results are gathered in tables 35, 36 and 37.

11.2.3 CBOC signals

The following table describes the simulation plan used for BPSK signals.

Parameter	Tested values
C/N_0	$[30, 35, 40, 45, 50]$
C_s (chips)	$[1/12]$
B (MHz)	$[14, 24]$
T_{DLL} (s)	$[0.001, 0.100]$
T_{PLL} (s)	$[0.001, 0.020]$

Table 31: BPSK simulation plan

The obtained results are gathered in tables 38 and 39.

$C/N0$	Chip Spacing (chip)	RF Bandwidth (MHz)	Coherent Integration Time (s)		Theory			Simulations							
					Holmes		Betz	Matlab Simulator				C Simulator			
			DLL	PLL	σ_τ (m)	σ_θ (rad)	σ_τ (m)	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error
50	0.5	4	0.001	0.001	0.47	0.010	0.37	0.38	0.01	0.010	0.000	0.41	0.04	0.010	0.000
50	0.25	4	0.001	0.001	0.33	0.010	0.33	0.35	0.02	0.010	0.000	0.32	0.01	0.010	0.000
45	0.5	4	0.001	0.001	0.84	0.018	0.67	0.73	0.06	0.017	0.001	0.75	0.08	0.0189	0.000
45	0.25	4	0.001	0.001	0.59	0.018	0.60	0.53	0.07	0.017	0.001	0.56	0.04	0.0189	0.000
40	0.5	4	0.001	0.001	1.56	0.032	1.24	1.46	0.22	0.032	0.000	1.35	0.11	0.0328	0.000
40	0.25	4	0.001	0.001	1.09	0.032	1.10	1.01	0.09	0.032	0.000	1.01	0.09	0.0324	0.000
35	0.5	4	0.001	0.001	3.11	0.060	2.46	2.58	0.12	0.060	0.000	2.64	0.18	0.0608	0.000
35	0.25	4	0.001	0.001	2.15	0.060	2.15	2.13	0.02	0.060	0.000	1.97	0.18	0.0600	0.000
30	0.5	4	0.001	0.001	7.08	0.122	5.56	5.52	0.04	0.122	0.000	5.58	0.02	0.1206	0.002
30	0.25	4	0.001	0.001	4.80	0.122	4.75	4.36	0.39	0.122	0.000	4.22	0.53	0.1208	0.002

Table 32: BPSK validation results ($B = 4MHz$, $T_{DLL} = 0.001\ s$, $T_{PLL} = 0.001\ s$)

$C/N0$	Chip Spacing (chip)	RF Bandwidth (MHz)	Coherent Integration Time (s)		Theory			Simulations							
					Holmes		Betz	Matlab Simulator				C Simulator			
			DLL	PLL	σ_τ (m)	σ_θ (rad)	σ_τ (m)	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error
50	0.5	4	0.020	0.020	0.46	0.010	0.37	0.44	0.07	0.010	0.000	0.41	0.04	0.010	0.000
50	0.25	4	0.020	0.020	0.33	0.010	0.33	0.33	0.00	0.010	0.000	0.31	0.02	0.010	0.000
45	0.5	4	0.020	0.020	0.82	0.018	0.66	0.75	0.09	0.018	0.00	0.73	0.07	0.019	0.001
45	0.25	4	0.020	0.020	0.58	0.018	0.59	0.60	0.01	0.018	0.000	0.55	0.04	0.019	0.001
40	0.5	4	0.020	0.020	1.47	0.032	1.17	1.38	0.21	0.033	0.001	1.26	0.09	0.033	0.001
40	0.25	4	0.020	0.020	1.04	0.032	1.05	1.04	0.01	0.033	0.001	0.98	0.07	0.033	0.001
35	0.5	4	0.020	0.020	2.63	0.056	2.10	2.38	0.28	0.059	0.003	2.27	0.17	0.059	0.003
35	0.25	4	0.020	0.020	1.86	0.056	1.86	1.86	0.00	0.058	0.002	1.70	0.16	0.057	0.001
30	0.5	4	0.020	0.020	4.78	0.101	3.81	4.20	0.39	0.103	0.002	3.73	0.08	0.100	0.001
30	0.25	4	0.020	0.020	3.37	0.101	3.39	3.27	0.12	0.103	0.002	2.96	0.43	0.101	0.000

Table 33: BPSK validation results ($B = 4MHz, T_{DLL} = 0.020\ s, T_{PLL} = 0.020\ s$)

$C/N0$	Chip Spacing (chip)	RF Bandwidth (MHz)	Coherent Integration Time (s)		Theory			Simulations							
					Holmes		Betz	Matlab Simulator				C Simulator			
			DLL	PLL	σ_τ (m)	σ_θ (rad)	σ_τ (m)	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error
50	0.2	10	0.001	0.001	0.29	0.010	0.24	0.26	0.02	0.010	0.000	0.25	0.01	0.010	0.000
50	0.1	10	0.001	0.001	0.21	0.010	0.21	0.22	0.01	0.010	0.000	0.20	0.01	0.010	0.000
45	0.2	10	0.001	0.001	0.53	0.018	0.43	0.48	0.05	0.018	0.000	0.46	0.03	0.018	0.000
45	0.1	10	0.001	0.001	0.37	0.018	0.38	0.35	0.03	0.018	0.000	0.39	0.01	0.018	0.000
40	0.2	10	0.001	0.001	0.98	0.032	0.79	0.87	0.08	0.032	0.000	0.84	0.05	0.032	0.000
40	0.1	10	0.001	0.001	0.69	0.032	0.69	0.68	0.01	0.032	0.000	0.65	0.04	0.033	0.001
35	0.2	10	0.001	0.001	1.92	0.061	1.54	1.55	0.01	0.062	0.001	1.61	0.07	0.060	0.001
35	0.1	10	0.001	0.001	1.35	0.061	1.35	1.23	0.12	0.060	0.001	1.24	0.11	0.060	0.001
30	0.2	10	0.001	0.001	4.26	0.123	3.41	3.57	0.16	0.122	0.001	3.42	0.01	0.118	0.005
30	0.1	10	0.001	0.001	2.97	0.123	2.98	2.75	0.23	0.124	0.001	2.64	0.34	0.121	0.002

Table 34: BPSK validation results ($B = 4MHz$, $T_{DLL} = 0.001\ s$, $T_{PLL} = 0.001\ s$)

$c/N0$	Chip Spacing (chip)	RF Bandwidth (MHz)	Coherent Integration Time (s)		Theory			Simulations							
			DLL	PLL	Holmes		Betz	Matlab Simulator				C Simulator			
					σ_τ (m)	σ_θ (rad)	σ_τ (m)	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error
50	0.25	4	0.001	0.001	0.19	0.010	0.19	0.16	0.03	0.010	0.000	0.19	0.00	0.010	0.000
50	0.21	4	0.001	0.001	0.17	0.010	0.19	0.16	0.03	0.010	0.000	0.18	0.01	0.10	0.000
45	0.25	4	0.001	0.001	0.34	0.018	0.35	0.27	0.08	0.017	0.001	0.34	0.01	0.018	0.000
45	0.21	4	0.001	0.001	0.30	0.018	0.35	0.27	0.08	0.018	0.000	0.31	0.04	0.018	0.000
40	0.25	4	0.001	0.001	0.63	0.032	0.65	0.55	0.10	0.031	0.001	0.62	0.03	0.032	0.000
40	0.21	4	0.001	0.001	0.56	0.032	0.65	0.49	0.16	0.032	0.000	0.57	0.08	0.032	0.000
35	0.25	4	0.001	0.001	1.22	0.060	1.32	1.06	0.26	0.059	0.001	1.23	0.09	0.061	0.001
35	0.21	4	0.001	0.001	1.09	0.060	1.32	0.97	0.35	0.060	0.000	1.11	0.21	0.061	0.001
30	0.25	4	0.001	0.001	2.67	0.122	3.10	2.37	0.73	0.125	0.003	2.55	0.55	0.126	0.004
30	0.21	4	0.001	0.001	2.39	0.122	3.10	2.16	0.94	0.126	0.004	2.47	0.63	0.123	0.001

Table 35: BOC(1,1) validation results ($B = 4MHz, T_{DLL} = 0.001\ s, T_{PLL} = 0.001\ s$)

$C/N0$	Chip Spacing (chip)	RF Bandwidth (MHz)	Coherent Integration Time (s)		Theory			Simulations							
					Holmes		Betz	Matlab Simulator				C Simulator			
			DLL	PLL	σ_τ (m)	σ_θ (rad)	σ_τ (m)	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error
50	0.25	4	0.100	0.020	0.19	0.010	0.19	0.16	0.03	0.010	0.000	0.19	0.00	0.010	0.000
50	0.20	4	0.100	0.020	0.17	0.010	0.19	0.14	0.05	0.010	0.000	0.15	0.04	0.011	0.001
45	0.25	4	0.100	0.020	0.34	0.018	0.34	0.28	0.06	0.018	0.000	0.31	0.03	0.018	0.000
45	0.20	4	0.100	0.020	0.30	0.018	0.34	0.27	0.07	0.018	0.000	0.29	0.05	0.018	0.000
40	0.25	4	0.100	0.020	0.60	0.032	0.60	0.51	0.09	0.032	0.000	0.58	0.02	0.033	0.001
40	0.20	4	0.100	0.020	0.54	0.032	0.60	0.45	0.15	0.032	0.000	0.51	0.09	0.031	0.001
35	0.25	4	0.100	0.020	1.07	0.056	1.07	1.02	0.05	0.057	0.001	1.08	0.01	0.058	0.002
35	0.20	4	0.100	0.020	0.95	0.056	1.07	0.80	0.27	0.059	0.003	0.92	0.15	0.058	0.002
30	0.25	4	0.100	0.020	1.90	0.101	1.92	1.73	0.19	0.102	0.001	1.77	0.15	0.103	0.002
30	0.20	4	0.100	0.020	1.70	0.101	1.92	1.43	0.49	0.107	0.006	1.58	0.34	0.106	0.005

Table 36: BOC(1,1) validation results ($B = 4MHz, T_{DLL} = 0.100\ s, T_{PLL} = 0.020\ s$)

$C/N0$	Chip Spacing (chip)	RF Bandwidth (MHz)	Coherent Integration Time (s)		Theory			Simulations							
			DLL	PLL	Holmes		Betz	Matlab Simulator				C Simulator			
					σ_τ (m)	σ_θ (rad)	σ_τ (m)	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error
50	0.25	10	0.100	0.020	0.19	0.010	0.16	0.17	0.01	0.010	0.000	0.18	0.02	0.011	0.001
50	0.2	10	0.100	0.020	0.17	0.010	0.14	0.16	0.02	0.010	0.000	0.16	0.02	0.010	0.000
45	0.25	10	0.100	0.020	0.34	0.018	0.28	0.30	0.02	0.019	0.001	0.34	0.06	0.019	0.001
45	0.2	10	0.100	0.020	0.31	0.018	0.25	0.29	0.04	0.020	0.002	0.30	0.05	0.019	0.001
40	0.25	10	0.100	0.020	0.60	0.032	0.50	0.50	0.00	0.034	0.002	0.57	0.07	0.034	0.002
40	0.2	10	0.100	0.020	0.55	0.032	0.45	0.47	0.02	0.033	0.001	0.50	0.05	0.034	0.002
35	0.25	10	0.100	0.020	1.07	0.056	0.89	0.95	0.06	0.058	0.002	0.99	0.10	0.058	0.002
35	0.2	10	0.100	0.020	0.98	0.056	0.79	0.89	0.10	0.060	0.004	0.90	0.11	0.059	0.003
30	0.25	10	0.100	0.020	1.90	0.101	1.58	1.68	0.10	0.105	0.004	1.77	0.19	0.110	0.009
30	0.2	10	0.100	0.020	1.74	0.101	1.42	1.43	0.01	0.106	0.005	1.60	0.18	0.108	0.007

Table 37: BOC(1,1) validation results ($B = 10MHz, T_{DLL} = 0.100\ s, T_{PLL} = 0.020\ s$)

$C/N0$	Chip Spacing (chip)	RF Bandwidth (MHz)	Coherent Integration Time (s)		Theory			Simulations							
					Holmes		Betz	Matlab Simulator				C Simulator			
			DLL	PLL	σ_τ (m)	σ_θ (rad)	σ_τ (m)	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error
50	1/12	14	0.100	0.020	0.08	0.010	0.07	0.06	0.01	0.010	0.000	0.07	0.00	0.010	0.000
50	1/12	24	0.100	0.020	0.08	0.010	0.07	0.06	0.01	0.010	0.000	0.06	0.01	0.010	0.000
45	1/12	14	0.100	0.020	0.15	0.018	0.13	0.12	0.01	0.019	0.001	0.11	0.02	0.019	0.001
45	1/12	24	0.100	0.020	0.15	0.018	0.12	0.11	0.01	0.019	0.001	0.11	0.01	0.019	0.001
40	1/12	14	0.100	0.020	0.26	0.032	0.23	0.20	0.03	0.035	0.003	0.19	0.04	0.033	0.001
40	1/12	24	0.100	0.020	0.26	0.032	0.21	0.18	0.03	0.034	0.002	0.19	0.02	0.034	0.002
35	1/12	14	0.100	0.020	0.46	0.057	0.41	0.35	0.06	0.061	0.004	0.33	0.02	0.059	0.002
35	1/12	24	0.100	0.020	0.46	0.057	0.37	0.33	0.04	0.062	0.005	0.34	0.03	0.060	0.003
30	1/12	14	0.100	0.020	0.82	0.101	0.73	0.63	0.10	0.109	0.008	0.58	0.1	0.110	0
30	1/12	24	0.100	0.020	0.82	0.101	0.66	0.58	0.08	0.111	0.010	0.62	0.04	0.103	0.002

Table 38: CBOC validation results ($C_S = {}^1/_{12}\ chip, T_{DLL} = 0.100\ s, T_{PLL} = 0.020\ s$)

$C/N0$	Chip Spacing (chip)	RF Bandwidth (MHz)	Coherent Integration Time (s)		Theory			Simulations							
					Holmes		Betz	Matlab Simulator				C Simulator			
			DLL	PLL	σ_τ (m)	σ_θ (rad)	σ_τ (m)	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error	σ_τ (m)	σ_τ error	σ_θ (rad)	σ_θ error
50	1/12	14	0.001	0.001	0.08	0.010	0.07	0.06	0.01	0.010	0.000	0.07	0.00	0.010	0.000
50	1/12	24	0.001	0.001	0.08	0.010	0.07	0.06	0.01	0.010	0.000	0.06	0.01	0.010	0.000
45	1/12	14	0.001	0.001	0.15	0.018	0.13	0.11	0.02	0.018	0.000	0.11	0.02	0.019	0.001
45	1/12	24	0.001	0.001	0.15	0.018	0.13	0.11	0.02	0.017	0.001	0.11	0.02	0.019	0.001
40	1/12	14	0.001	0.001	0.27	0.032	0.24	0.19	0.05	0.032	0.000	0.19	0.05	0.033	0.001
40	1/12	24	0.001	0.001	0.27	0.032	0.22	0.20	0.02	0.031	0.001	0.19	0.03	0.034	0.001
35	1/12	14	0.001	0.001	0.53	0.061	0.47	0.38	0.09	0.061	0.000	0.33	0.16	0.059	0.002
35	1/12	24	0.001	0.001	0.53	0.061	0.43	0.39	0.04	0.060	0.001	0.34	0.09	0.060	0.001
30	1/12	14	0.001	0.001	1.15	0.123	1.04	0.81	0.23	0.126	0.003	0.58	0.46	0.110	0.013
30	1/12	24	0.001	0.001	1.15	0.123	0.94	0.84	0.10	0.126	0.003	0.62	0.32	0.103	0.020

Table 39: CBOC validation results ($C_S = 1/12$ $chip$, $T_{DLL} = 0.001$ s, $T_{PLL} = 0.001$ s)