GNSS-based Attitude Estimation using Single Baseline

André Oliveira

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisors: Prof. Paulo Jorge Coelho Ramalho Oliveira
Prof. Pedro Tiago Martins Batista

Examination Committee

Chairperson: Prof. José Fernando Alves da Silva
Supervisor: Prof. Paulo Jorge Coelho Ramalho Oliveira
Member of the Committee: Prof. José Eduardo Charters Ribeiro da Cunha Sanguino

January 2021

Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Agradecimentos

Em primeiro lugar, gostaria de agradecer aos meus orientadores, Professor Paulo Oliveira e Professor Pedro Batista, por terem proposto este tema e desafio. No decorrer deste longo ano, ambos foram incansáveis ao partilharem a sua sabedoria, indicarem o caminho mais acertado e esclarecerem qualquer dúvida.

Esta dissertação representa, até ao momento, o culminar do meu percurso no ensino superior. Assim, gostaria de agradecer a todos os Professores que me ajudaram a desenvolver as ferramentas, ética de trabalho e sentido crítico necessários a este trabalho de investigação, e que sem dúvida continuarão a ser úteis no futuro.

Por último, agradeço à minha família e círculo de amigos mais próximos. Nomeadamente, aos meus pais e irmã, que me apoiaram nos momentos mais difíceis. No contexto da dissertação, agradeço especialmente à minha mãe por ter proporcionado as melhores condições possíveis à realização deste trabalho, e ao meu pai pela ajuda contínua durante a parte experimental desta investigação. Aos meus amigos e colegas, gostaria de agradecer por terem tornado mais tolerável o período de confinamento e cuidados acrescidos, devido à pandemia Covid-19. O apoio de colegas, que passam pelas mesmas dificuldades inerentes à realização da tese nestas circunstâncias, foi crucial, o qual espero ter conseguido retribuir.

Resumo

Esta dissertação foca-se no desenvolvimento, implementação e validação preliminar de um algoritmo de estimação de atitude baseado em Sistemas Globais de Navegação por Satélite (GNSS, de Global Navigation Satellite Systems), adequado para aplicações de navegação.

O problema é primeiramente definido usando uma formulação de mínimos quadrados, da qual segue uma decomposição ortogonal típica deste tipo de problema de minimização não-linear com restrições de números inteiros. A solução proposta combina um filtro de Kalman estendido, guiado por medições de um giroscópio, com um método de determinação de ambiguidades inteiras atual, focando a maneira como estes são fundidos. Posteriormente, estuda-se a observabilidade do filtro recursivo usando o modelo linearizado do sistema. Destaca-se o estudo do caso onde são utilizadas apenas duas antenas *GNSS*. Tratando-se de um problema não trivial, são efetuadas validações adicionais experimentalmente.

A solução é então validada em cenários realistas, usando dados quer simulados quer obtidos experimentalmente. É apresentada uma visão global do *software* de pós-processamento implementado, junto com o ambiente de simulação desenvolvido e o equipamento usado durante os ensaios experimentais. Após uma breve descrição dos ensaios realizados, são apresentados os respetivos resultados e conclusões chave acerca da percentagem de sucesso da determinação de ambiguidades inteiras e consequente precisão da atitude estimada. Novamente, é realçada a análise do uso de apenas duas antenas *GNSS*. Por último, são enumerados os aspetos que não foram abordados devido à complexidade deste trabalho, os quais são sugeridos para um possível trabalho futuro.

Palavras-chave: Estimação de atitude, GNSS, Resolução de ambiguidades, Filtragem, Fusão Sensorial

Abstract

This work deals with the development, implementation and preliminary validation of a Global Navigation Satellite Systems (GNSS)-based attitude estimation algorithm suitable for navigation applications.

The problem is first formulated in a least-squares sense, from which follows an orthogonal decomposition typical to this type of nonlinear minimization problem with integer constraints. The proposed solution combines an Extended Kalman Filter (EKF), fed by gyroscope measurements, with a state of the art integer ambiguity determination method, with special focus given to how these are joined. Then, some insight on the observability of the recursive filter is provided, using a linearized model of the system. The scenario where only two GNSS antennas are used is highlighted. Being a non-trivial topic, further validations are performed experimentally.

The solution is then validated in realistic scenarios using both simulation and experimental data. An overview of the implemented post-processing software is given, along with the developed simulation environment and hardware setup used during the experimental trials. Following a brief description of the performed trials, the respective results and key conclusions about the success rate of integer ambiguity determination as well as the consequent attitude estimation accuracy are then presented. Once again, the evaluation of the single baseline case is emphasized. Lastly, the issues that could not be tackled due to the complexity of this work and remain for future work are enumerated.

Keywords: Attitude estimation, GNSS, Ambiguity resolution, Filtering, Sensor fusion

Contents

List of Tables

List of Figures

$_{\Psi,i}$

Nomenclature

Observation models

P_i^k	Code measurement of satellite k provided by receiver i.
Ψ_i^k	Carrier phase measurement of satellite k provided by receiver i.
ρ_i^k	Distance between the phase centers of antenna i and the transmission antenna of satellite k.
$\boldsymbol{r}_i$	Position of receiver i.
$\boldsymbol{r}^k$	Position of satellite k.
τ_i^k	Propagation time of the signal transmitted by satellite k and received by antenna i.
dt_i	Clock offset of receiver i.
dt^k	Clock offset of satellite k.
I_i^k	Ionospheric delay of the signal transmission between satellite k and antenna i.
T_i^k	Tropospheric delay of the signal transmission between satellite k and antenna i.
$\boldsymbol{R}_3$	Rotation matrix relative to rotation around the Z axis.
$\phi_{i,0}$	Initial offset of the signal phase at receiver i.
ϕ_0^k	Initial offset of the signal phase at satellite k.
n_i^k	Integer ambiguity of the observations given by receiver i, relative to satellite k.
$\epsilon_{P,i}^k$	Unmodelled error in the code measurement model, relative to receiver i and satellite k.
$\epsilon_{\Psi,i}^k$	Unmodelled error in the carrier phase measurement model, relative to receiver i and satellite k.
$\sigma_{P,i}^k$	Standard deviation of code measurement relative to receiver i and satellite k.
$\sigma_{\Psi,i}^k$	Standard deviation of code measurement relative to receiver i and satellite k.
ϕ_0^k	Initial offset of the signal phase at satellite k.
w_e	Earth's mean angular rotation velocity: $w_e = 7.292\,115\,147 \times 10^{-5}\,\mathrm{rad/s}$.
c	Speed of light in vacuum: $c = 299\,792\,458\,\mathrm{m/s}$.

f	Signal carrier's frequency. At the L1 frequency, $f = 1575.42\,\mathrm{MHz}$.
λ	Signal carrier's wavelength.
$\boldsymbol{u}_i^k$	Line of sight vector from antenna i to satellite k.
$\boldsymbol{b}_{ij}$	Baseline vector between antennas i and j written in the reference frame.
$\boldsymbol{l}_{ij}$	Baseline vector between antennas i and j written in the body frame.
m	Number of satellites -1.
n	Number of antennas -1, or number of baselines.
$\boldsymbol{Y}$	Matrix of double differenced observations.
$\boldsymbol{S}$	Design matrix linking the integer ambiguities to the observations.
$\boldsymbol{N}$	Matrix of double differenced integer ambiguities.
$\boldsymbol{\Upsilon}$	Matrix of line of sight vectors.
$\boldsymbol{U}$	Design matrix linking the baseline coordinates to the observations.
$\boldsymbol{B}$	Matrix of baseline coordinates in the reference frame.
$\boldsymbol{F}$	Matrix of baseline coordinates in the body frame.
$\boldsymbol{Q}_{\boldsymbol{vv}}$	Covariance matrix of vector $\boldsymbol{v}$.
$\boldsymbol{Q}_{\boldsymbol{MM}}$	Covariance matrix of vector $vec(\boldsymbol{M}$, for a matrix $\boldsymbol{M}$.
$\boldsymbol{I}_n$	$n \times n$ identity matrix.

Attitude representation

$\boldsymbol{R}$	Rotation matrix.
$\boldsymbol{q}$	Unit quaternion.
q_w	Real part of the unit quaternion.
$\boldsymbol{q}_v$	Imaginary or vector part of the unit quaternion.
$\boldsymbol{R}(\boldsymbol{q})$	Rotation matrix corresponding to unit quaternion $\boldsymbol{q}$.
$\boldsymbol{e}$	Euler axis.
φ	Roll angle.
θ	Pitch angle.
ψ	Yaw angle.

Attitude estimation

$\boldsymbol{\omega}$	True angular velocity.
p, q, r	Angular velocities around the X, Y and Z axis.
$\boldsymbol{\omega}_m$	Measured angular velocity.
$\boldsymbol{b}$	Gyroscope bias.
$\boldsymbol{\eta}_v,\ \boldsymbol{\eta}_u$	White noise processes.
$\check{\boldsymbol{M}}$	Estimator of matrix $\boldsymbol{M}$ in the integer constrained problem.
$\hat{\boldsymbol{M}}$	Estimator of matrix $\boldsymbol{M}$ in the float solution problem.
$\hat{\boldsymbol{v}}$	Estimator of vector $\boldsymbol{v}$ in the float solution problem.
$\boldsymbol{R}(\boldsymbol{N})$	Conditional rotation matrix.
$\boldsymbol{q}(\boldsymbol{N})$	Conditional unit quaternion.
$\boldsymbol{x}_k$	Nominal system state vector.
$\boldsymbol{n}$	Vector of double differenced ambiguities.
$\boldsymbol{\Psi^*}_{k+1,k}$	Jacobian matrix of the quaternion discrete kinematics in relation to quaternion.
$\boldsymbol{L}(\boldsymbol{x}_k)$	Jacobian matrix of the quaternion discrete kinematics in relation to the gyro bias.
$\boldsymbol{F}_k$	Propagation step state transition matrix.
$\boldsymbol{Q}_k$	Process noise covariance matrix.
$\boldsymbol{K}_k$	Kalman gain.
$\boldsymbol{z}_k$	Extended kalman filter measurements vector.
$\boldsymbol{H}_k$	Update step observation matrix.
$\boldsymbol{R}_k$	Measurements noise covariance matrix.
$\boldsymbol{g}(\boldsymbol{q})$	Nonlinear function relating the unit quaternion to the measurements.
$\boldsymbol{G'}(\boldsymbol{q})$	Jacobian matrix of the measurements in relation to the quaternion.
$\boldsymbol{J}_{\boldsymbol{q}}$	Jacobian matrix of the quaternion rotation in relation to the quaternion.

Glossary

ARW	Angular Random Walk
C-ILS	Constrained Integer Least Squares
C-LAMBDA	Constrained LAMBDA
DD	Double Differenced
ECEF	Earth-Centered Earth-Fixed
EKF	Extended Kalman Filter
ENU	East-North-Up
FOAM	Fast Optimal Attitude Matrix
GNC	Guidance, Navigation and Control
GNSS	Global Navigation Satellite Systems
GPS	Global Positioning System
IAD	Integer Ambiguity Determination
ILS	Integer Least Squares
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
KF	Kalman Filter
LAMBDA	Least-squares AMBiguity Decorrelation Adjustment
LOS	Line of Sight
LS	Least Squares
LTI	Linear Time-Invariant
MC-LAMBDA	Multivariate Constrained LAMBDA
OPP	Orthogonal Procrustes Problem
QUEST	Quaternion Estimator
RF	Radio Frequency
RINEX	Receiver Independent Exchange Format
RRW	Rate Random Walk
RTK	Real Time Kinematic
UAV	Unmanned Aerial Vehicle
WLS	Weighted Least Squares

Chapter 1

Introduction

In the context of Guidance, Navigation and Control (GNC) systems engineering, navigation deals with monitoring the movement of a craft or vehicle. Historically, the concept of navigation was usually associated with the art of directing ships at open sea, through determining its position and course in order to be able to safely get from one point to another. Most recently, through the technological advances that led to the development of rockets, spacecrafts, or autonomous vehicles, the concept of navigation has been extended in order to include land, aeronautic and space navigation as well. Notably, the advent of airplanes in the early 20th century sparked a growing interest in this field, enabling innovative solutions for airborne vehicles. Even more so after World War II, with the development of new inertial and electronic navigation systems. Nowadays, navigation solutions are often digital, implemented on embedded systems and allow flying the skies, exploring space, and for anyone who carries a smartphone to know their approximate location.

At a given time, a complete navigation solution should be able to provide the vehicle's position and velocity, as well as its attitude or orientation relative to a certain reference frame. The required information varies according to the specific application. Autopilot systems, spacecraft launch vehicles, satellites, unmanned aerial vehicles (UAVs) or guided missiles applications are just a few that can be highlighted. When it comes to attitude estimation, the common approaches usually integrate information from at least two sensors. Inertial navigation systems (INS) used in vehicles such as ships or aircrafts commonly use rate gyros coupled with magnetometers, accelerometers and other devices in order to estimate the attitude parameters of the body. For vehicles such as rockets or other spacecrafts, typically used sensors include also gyroscopes and magnetometers, along with Earth sensors or star trackers and Sun sensors, giving the vehicle's relative orientation with respect to celestial bodies. Estimation theory, and Kalman filtering in particular, establish tools based on a thoroughly studied mathematical formulation for combining the measurements from different sensors, also known as sensor fusion.

On the other hand, navigation satellite systems are more commonly used for autonomous geospacial positioning. Previously, radio navigation systems such as the LOng RAnge Navigation (LORAN) or OMEGA [1], currently no longer in use, relied on the same principle of broadcasting radio signals, and measuring the delays relative to the signal's propagation. The first navigation satellite system, TRAN-

SIT, was based on measuring the Doppler frequencies of the signal broadcast by a satellite traveling in a known orbit [2]. The change in frequency over time, combined with knowledge of the precise satellite's position, allowed determining the location of the signal receiver.

Modern systems use signals from several satellites in order to determine the position instantaneously, using a kind of trilateration process [3]. They are also simpler and more direct, as the information broadcast by the satellites is all that is needed by the receiver to perform the computations. Examples of such satellite systems providing global coverage, broadly designated Global Navigation Satellite Systems (GNSS), are the Global Positioning System (GPS), owned by the United States government, or Russia's GLONASS, among others. Having been around for decades, applications of these systems include navigation systems, surveying and mapping or providing precise time references, among others. Apart from precise positioning applications, estimation of a vehicle's attitude is also possible by using multiple GNSS antennas.

This work focuses on using the information provided by the Global Navigation Satellite Systems for attitude estimation. The reasons behind this choice are given next. Then, a brief description of what solving this problem entails is given, followed by an overview of the state of the art methods currently in use. The goals of this thesis are enumerated, and lasty the structure of this document is described.

1.1 Motivation

The problem of attitude estimation using GNSS has been tackled extensively over the last decades, such that the advantages and disadvantages of doing so are clearly known. While the accuracy of a GNSS-based attitude system by itself may not be comparable to other modern solutions integrating several of the sensors enumerated before, this system requires very low maintenance. Global coverage is always available and the reliability of the satellites is guaranteed by control ground stations. Furthermore, these solutions are driftless, as opposed to a rate gyro, and the GNSS antennas and receivers are overall considerably less expensive that other space-grade sensors such as star trackers, INS or Ring Laser Gyros (RLGs) [4]. Notably, even using single-frequency low-cost GPS receivers, precisions of $1.6°$ are possible [5].

These types of attitude estimation solutions are also highly versatile. As their INS counterparts, GNSS-based attitude systems have been implemented on board size constrained, low-weight UAVs [6], ships [7], aircrafts [8], spacecrafts [9] and even very small nano-satellites [10]. Furthermore, GNSS observations can be fused with information from Inertial Measurement Units (IMU) [7, 10], compensating for some of the shortcomings of these sensors, for example correcting the gyroscope drift. Since the use of GPS and other satellite navigation systems is already the golden standard when it comes to precise positioning systems, the antennas and receivers are already commonly installed as part of the overall navigation solution for these vehicles. Thus, augmenting the attitude solution so as to include GNSS measurements comes at a very low additional cost while providing the benefits previously described. For these reasons, GNSS-based attitude estimation still draws attention from the navigation community [11].

The large amount of possible combinations between the type and number of GNSS receivers and antennas, integrated IMU or other sensors and integer atittude determination (IAD) schemes present in the literature allow for the development and testing of novel solutions, even after decades of GPS availability. In the present work, a solution based on the novel Multivariate Constrained LAMBDA (MC-LAMBDA) method [8] is proposed, along with a recursive filter integrating rate-gyro measurements. The motivation behind this solution lies with studying how integration of a kinematics model allows full attitude determination using a single baseline (two antennas) in certain scenarios, as well as how it impacts the accuracy of the solution.

1.2 Topic Overview

In order to relate the collected GNSS measurements to the orientation of the vehicle, the antennas must be mounted solid to the body. Then, after implementation of the attitude system, the processing can occur autonomously. High accuracy GNSS-based attitude estimation solutions rely on carrier phase measurements of the signal transmitted by each satellite. These provide a measure of the antenna to satellite range, although ambiguously, being off by an integer multiple of the signal wavelength. The noisier code measurements are often also used to help with determining the integer ambiguities present in the carrier phase measurements. As such, the correct determination of the integer ambiguities is crucial.

Some of the methods proposed in the literature employ a search algorithm which determines the integer ambiguities based on the geometry between the GNSS antennas. The selected ambiguities minimize a cost function built from the GNSS observables model, presented later on. In other solutions propose obtaining a preliminary, less accurate solution, which becomes a starting point for the search algorithm. Having the correct set of integer ambiguities, estimation of the body's orientation relies once again on the observation models with relate the observations with the body attitude. The process of obtaining both the integer ambiguities and the attitude estimates can also be aided by other sensors, such as the gyroscope or magnetometer. These and other variants of possible GNSS-based attitude estimation schemes are described next.

1.3 State of the Art

Several techniques have been proposed in the literature in order to solve the attitude estimation problem using GNSS observations.

Most solutions focus on using at least three GNSS antennas in order to obtain the complete orientation of the body frame in relation to a reference frame, usually presented in the form of Euler angles. Using only two antennas, it is generally not possible to do so, unless other sensors such as gyroscopes, accelerometers and magnetometers are used. These solutions are of particular interest for applications where low volume and complexity are important concerns. Other factors influencing the accuracy and stability of the attitude estimates are the antenna configuration, the integer ambiguity resolution method used and the interferences and experimental scenario affecting signal quality.

In the following, an overview of the different possibilities, found in the literature, is given. Generally, the GNSS-based attitude estimation methods can be divided into two categories.

The first group consists of baseline estimation methods, which are concerned with initially estimating the coordinates of the baselines between the GNSS antennas. This process relies on an observation model which relates the code and carrier phase measurements with the baseline coordinates in the reference frame, as shown in the next Chapter. Them, these types of solutions can be further categorized as instantaneous epoch-by-epoch methods, or recursive methods. A widely used epoch-by-epoch method is detailed in [12], where the baseline coordinates are obtained from a linear least-squares (LS) adjustment based on the baseline-based observation model. Initially, a preliminary float solution is obtained by dropping the integer constraints. Afterwards, the integer ambiguities are determined, from which follows an improved attitude solution. This decomposition of the estimation problem in several steps is common throughout the literature.

On the other hand, Eling et al. [6] designed a recursive filter in order to estimate the baselines using the GNSS measurements collected over time.In this Extended Kalman Filter (EKF) solution, the overall attitude system also integrates attitude estimates obtained using real time kinematic (RTK) GPS positioning techniques, IMU readings and magnetic field measurements. By doing so, angle precisions of $0.2°$ are obtained and a single baseline can be used to estimate the full attitude of the body. For more details on this solution, see [6]. In this first group of methods, the estimated baseline coordinates are then used to obtain the body attitude. One of several available methods can be used to estimate the rotation that relates the coordinates in the body and reference frames. Some possibilities are described in [9, 13–15] and include, for example, Davenport's q-method or singular value decomposition-based methods.

Concerning the second category of methods, the attitude is estimated directly from the GNSS observations. This is done by replacing the baseline coordinates in the reference frame, present in the GNSS observation models, with the rotation of the known body frame baselines.The resulting GNSS-based attitude model is then used to estimate the body's attitude. The LS approach introduced by Teunissen in [16] precisely aims to solve this minimization problem. Again dropping the integer and nonlinear constraints, a float solution is estimated epoch-by-epoch using the normal equations. The decomposition of the residual squared norm followed in this approach is the basis on which the work of this thesis is built, as detailed later.

Other very interesting possibilities are presented in [9], based on both the baseline and attitude models. For example, an iterative LS adjustment based on the linearization of an attitude model using a quaternion parametrization is described. Since it is an epoch-by-epoch algorithm, no kinematics models of the system are used in this solution. The accuracy of this method is then compared with other solutions, such as that of an EKF using the same GNSS-based attitude model mentioned earlier, integrated with a kinematics model of the body's angular motion. As expected, the method using a recursive filter improves the Euler angles accuracy, to around $0.31°$.

All the enumerated solutions require that the integer ambiguities relative to the number of wavelengths between antenna and satellite be correctly determined, in order to use the highly accurate carrier phase observations. Due to the integer constraint of the unknown ambiguities, the methods in the literature usually involve some kind of search algorithm. Exceptions to this are the methods used in [9, 10], where the ambiguities are determined by first estimating the baseline vectors, and then inverting the baseline model to compute an approximate ambiguity solution. Afterwards, this solution is rounded to the nearest integers. This approach is valid only for baseline lengths on the same order of magnitude as the GNSS signal's wavelength. Regarding the search approaches, efficient and highly successful integer ambiguity determination (IAD) algorithms were proposed by Teunissen. The Least AMBiguity Decorrelation Adjustment (LAMBDA) finds the integer set of ambiguities minimizing a cost function which weights the distance to a preliminary float solution [12]. In turn, the Constrained LAMBDA (C-LAMBDA) [17] also takes into account the known baseline lengths, restricting the possible integer ambiguity solutions. The Multivariate Constrained LAMBDA (MC-LAMBDA) [8] also proposed by Teunissen makes use of both the baseline lengths and the relative geometry between the baselines, increasing the success rate of IAD. A comparison between these two methods is given in [18]. Using the simulated data, the MC-LAMBDA computes the correct ambiguities set 100% of the time, compared to the slightly worse 98.9% using the C-LAMBDA method.

The method proposed in this thesis draws from the recursive approaches enumerated in this Chapter, since the attitude is estimated more accurately and it becomes possible to integrate information from other sensors. For ambiguity determination, the MC-LAMBDA method is used and slightly modified in order to fit the approximate solution provided by the recursive filter.

1.4 Objectives

Overal, this thesis aims at developing an attitude system which successfully integrates GNSS measurements with IMU sensors, providing adequate accuracy for high precision applications such as autonomous driving or UAVs. At this point of development, the goal is to focus on designing and validating the solution in post-processing of the acquired data.

In the context of GNSS-based attitude estimation, the particular objectives of this work are summarized as

- Designing a solution which takes advantage of the precise carrier phase measurements as well as angular velocity measurements taken from a gyroscope;
- Defining and implementing a state of the art integer ambiguity resolution method with high success rate;
- Utilizing the determined integer ambiguities to provide a attitude solution with, at least, sub-degree precision;
- Analytically assessing the conditions in which the developed solution is capable of estimating the full attitude of the vehicle;

- Complete implementation of the attitude system using suitable software;
- Implemention of a simple GNSS observables simulator for validation of the developed solution in a controlled environment;
- Performing actual experiments using GNSS hardware and a gyroscope in scenarios with different attitude dynamics; and
- Evaluating the performance of attitude estimation using both simulated and experimental data.

1.5 Thesis Outline

Keeping in mind the objectives defined previously, this dissertation is organized as follows.

Chapter 2 serves as an introduction of the basic notions and theory relevant for the problem at hand, such as possible attitude parametrizations and GNSS observation models.

This problem is then strictly formulated in Chapter 3, highlighting the GNSS-based attitude model on which the attitude estimation solution is based.

The proposed framework to solving the problem stated in the previous Chapter is detailed in Chapter 4. The structure of the solution is described, involving an EKF which provides approximate attitude and float ambiguity solutions, followed by the MC-LAMBDA method used for integer ambiguity resolution. Furthermore, some insights on the observability properties of the filter are provided by means of a linearized model.

Afterwards, Chapter 5 deals with presenting the overall structure of the designed software. The simulation and experimental setups through which GNSS and gyroscope data is obtained are also detailed.

Utilizing the setup described in the previous Chapter, the simulation and experimental trials done for validation of the algorithm are presented in Chapter 6. The corresponding results are then shown and discussed.

Finally, the main results concerning this research are written in Chapter 7. The dissertation would not be complete without also enumerating this work's contributions, as well as suggestions for improvements and work to be done in the future.

Chapter 2

Theoretical Background

This Chapter provides an overview of some mathematical concepts and topics that must be clearly defined in order to formulate the main problem that the current thesis aims to solve.

Initially, possible ways of defining the attitude between two coordinate frames that are relevant to this work are briefly reviewed in Section 2.1. As there are some representations that allow for a number of possible definitions, namely the Euler angles and the quaternion, the conventions adopted are defined.

In Section 2.2, the coordinate frames in which vector quantities are to be expressed are enumerated. The relevant relations between frames, used, in this case, to transform vector coordinates, are also presented.

The proposed method for Attitude Estimation is based on Global Navigation Satellite Systems (GNSS) observables, namely pseudorange and carrier phase measurements. To build a mathematical model which allows to relate these measurement to the unknown quantities that are to be estimated, one must understand some GNSS fundamentals. These are presented in Section 2.3.

2.1 Attitude Representation

In the context of attitude estimation, the attitude of a body is the relative orientation of a reference frame fixed to the body, the appropriately called body frame, to a local-level frame such as the East-North-Up (ENU) frame. More broadly, one can define the relative orientation between any two coordinate frames using one of several possible attitude parametrizations, see [19]. Highlighted hereon are the direct cosine matrix, Euler angle and quaternion representations.

The direct cosine matrix, or rotation matrix, $\boldsymbol{R}_a^b$, will be defined as giving the relative orientation of a coordinate frame "b" in respect to a coordinate frame "a". Having the coordinates of a vector $\boldsymbol{r}$ written in frame "a", $\boldsymbol{r}_a$, and considering that the frames differ only in orientation, the coordinates of the vector in frame "b", $\boldsymbol{r}_b$, are computed by

$$\boldsymbol{r}_b = \boldsymbol{R}_a^b \boldsymbol{r}_a. \tag{2.1}$$

The 3×3 rotation matrix $\boldsymbol{R}_a^b$ completely describes the relative orientation between the two coordinate frames unambiguously and without singularities. One drawback of using this representation is that it includes nine parameters, the entries of the rotation matrix, even though there are in fact only three degrees of freedom, due to the six magnitude and orthogonality constraints. Other representations, such as the Euler angles or quaternion parametrization, only require three or four parameters, respectively, decreasing the computational load.

A coordinate frame ”a” can be rotated into any other frame ”b” by performing a maximum of three successive rotations around the coordinate axes. The magnitudes of each rotation are defined as the Euler angles. The freedom of choice regarding which axes to rotate, in which order and whether the rotation is defined around the axes of the initial frame or of each intermediate frame give rise to several possibilities. As such, the attitude can be described by different triplets of Euler angles, depending on the chosen convention. Through this document, the chosen Euler angles are the ones relative to sequential intrinsic rotations around the z, y and x axes of right-handed frames. Defined in this way, the Euler angles are commonly known as the yaw, pitch and roll angles (ψ, θ and φ). Representing each sequential rotation using the direct cosine matrix allows to write the overall rotation matrix $\boldsymbol{R}_a^b$ as a function of the Euler angles (2.2).

$$\boldsymbol{R}_a^b = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\varphi - \cos\varphi\sin\psi & \sin\psi\sin\varphi + \cos\psi\cos\varphi\sin\theta \\ \cos\theta\sin\psi & \cos\psi\cos\varphi + \sin\psi\sin\theta\sin\varphi & \cos\varphi\sin\psi\sin\theta - \cos\psi\sin\varphi \\ -\sin\theta & \cos\theta\sin\varphi & \cos\theta\cos\varphi \end{bmatrix} \tag{2.2}$$

The Euler angles are commonly used to describe the attitude because they provide a convenient physical interpretation of the parameters. However, they can be troublesome to manage if used during computations because singularities can occur at particular orientations. This must be taken into account if the attitude of the body is unpredictable or if it is subject to high angular variations. In these cases, the more common approach is using a quaternion representation.

The unit quaternion representing the orientation between two frames can be expressed as (2.3), where $\boldsymbol{q}_v$ is the imaginary or vector part of the quaternion and q_w denotes the real part. As seen by this definition, throughout this document the Hamilton convention of the unit quaternion will be used. For details on different conventions and differences between them, see [20].

$$\boldsymbol{q} = q_w + iq_x + jq_y + kq_z = \begin{bmatrix} q_w \\ \boldsymbol{q}_v \end{bmatrix}, \quad \boldsymbol{q}_v = \begin{bmatrix} q_x \\ q_y \\ q_z \end{bmatrix}. \tag{2.3}$$

A physical interpretation of the quaternion can be grasped by expressing its relation to the Euler rotation vector. Any arbitraty rotation between two frames can be described by a rotation of an angle α around a certain axis, whose direction is specified by the unit vector $\boldsymbol{e}$. This can also be seen as an exponential map of the rotation vector to the quaternion [20], as written in (2.4).

$$\boldsymbol{q}\{\alpha\boldsymbol{e}\} \triangleq \mathrm{Exp}(\alpha\boldsymbol{e}) = e^{\alpha\boldsymbol{e}/2} = \begin{bmatrix} \cos\alpha/2 \\ \boldsymbol{e}\sin\alpha/2 \end{bmatrix}. \tag{2.4}$$

From the definition in (2.4), it follows that the quaternion related to the orientation between two frames has a unit-norm constraint, $\|\boldsymbol{q}\| = 1$. Quaternions that obey this constraint represent a unique rotation and are said to belong to the unit-quaternion group, denoted as $\mathbb{H}^*$. Following its definition and relation with the Euler vector, the rotation matrix describing the passive transformation between two frames can be written as a function of the corresponding quaternion as

$$\boldsymbol{R}(\boldsymbol{q}) = (q_w^2 - \boldsymbol{q}_v^\top \boldsymbol{q}_v)\boldsymbol{I}_3 + 2\boldsymbol{q}_v\boldsymbol{q}_v^\top + 2q_w[\boldsymbol{q}_v\times]. \tag{2.5}$$

Note that in the context of attitude estimation, this rotation matrix shall represent the local-to-global transformation. The notation $[\boldsymbol{v}\times]$ in (2.5) denotes the skew operator representing the cross-product matrix (2.6).

$$[\boldsymbol{v}\times] = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix} \tag{2.6}$$

Like the rotation matrix, the unit quaternion can be used to rotate vectors or transform vector coordinates. Following the Hamilton convention, the coordinates in frame ”a” $\boldsymbol{r}_a$ can be transformed into frame ”b” as written in (2.7)

$$\boldsymbol{r}_b = \boldsymbol{R}(\boldsymbol{q})\boldsymbol{r}_a = \boldsymbol{q} \circ \boldsymbol{r}_a \circ \boldsymbol{q}^* = \begin{bmatrix} q_w \\ \boldsymbol{q}_v \end{bmatrix} \circ \begin{bmatrix} 0 \\ \boldsymbol{r}_a \end{bmatrix} \circ \begin{bmatrix} q_w \\ -\boldsymbol{q}_v \end{bmatrix}, \tag{2.7}$$

where the operator $\circ$ denotes the quaternion product, defined in (2.8), and $\boldsymbol{q}^*$ is the quaternion conjugate $\boldsymbol{q}^* = [q_w \;\; -\boldsymbol{q}_v^\top]^\top$. In the case of the operation defined in (2.7), since the rotated vector $\boldsymbol{r}_b \in \mathbb{R}^3$, only the vector or imaginary part of the result of the quaternion product is considered.

$$\boldsymbol{p} \circ \boldsymbol{q} = \begin{bmatrix} p_w q_w - \boldsymbol{p}_v^\top \boldsymbol{q}_v \\ p_w\boldsymbol{q}_v + q_w\boldsymbol{p}_v + \boldsymbol{p}_v \times \boldsymbol{q}_v \end{bmatrix} \tag{2.8}$$

The quaternion product expression (2.8) can also be written in matrix product form (2.9).

$$\boldsymbol{p} \circ \boldsymbol{q} = [\boldsymbol{q}]_R\,\boldsymbol{p}, \quad [\boldsymbol{q}]_R = \begin{bmatrix} q_w & -q_x & -q_y & -q_z \\ q_x & q_w & q_z & -q_y \\ q_y & -q_z & q_w & q_z \\ q_z & q_y & -q_x & q_w \end{bmatrix} \tag{2.9}$$

2.2 Coordinate Frames

The Earth-fixed or Earth-fixed Earth-centered (ECEF) frame is, as the name entails, fixed with respect to the Earth. As the Earth rotates around its spin axis, so does the Earth-fixed frame. The origin of the frame coincides with the position of the Earth’s center of mass, its X-axis points towards the mean meridean of Greenwich while the Z-axis follows the direction of the mean spin axis of the Earth. The remaining axis results from the right-handedness of the orthogonal frame.

Along with this frame, one can define a more convenient surface which allows to write positions with respect to the Earth's surface. In geodesy, the reference surface is an ellipsoid of revolution [3]. The shape of the ellipsoid is commonly defined by its semi-major axis a, and flattening f or eccentricity e. Different values for these parameters result in different geodesic systems or *datums*. Each geodetic system tries to minimize the error incurred by approximating the Earth's surface as an ellipsoid whether globally or in a particular area of the Earth. The coordinate system used by the GPS, and the one used in this research, is the WGS84 datum. The relation between the ECEF frame and the reference ellipsoid is shown in Figure 2.1.

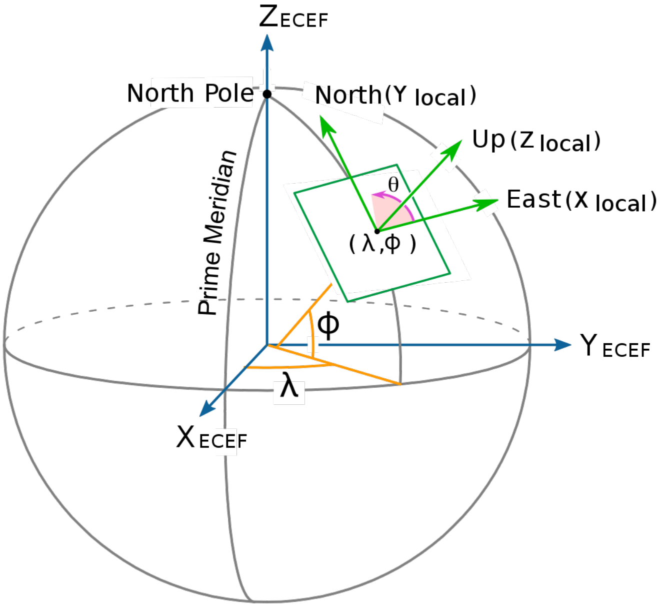


Figure 2.1: Earth-fixed coordinate frame and local-level ENU frame.[21]

The represented geodetic coordinates latitude ϕ and longitude λ correspond to the angle formed by a particular point with respect to the ECEF X-axis and the equatorial plane, respectively. Together with the height h above the ellipsoid, which is measured perpendicularly to the ellipsoid tangent plane, any position can be clearly defined. A given point P with geodetic coordinates ϕ, λ and h is written in cartesian coordinates in the ECEF frame according to (2.10) [3]. The quantity N corresponds to the distance between the ECEF z-axis and the ellipsoid surface, along the normal at P.

$$\begin{cases} X = (N+h)\cos\phi\cos\lambda \\ Y = (N+h)\cos\phi\sin\lambda \\ Z = ((1-f)^2 N + h)\sin\phi \end{cases} \quad , \quad N = a/\sqrt{1 - f(2-f)\sin^2\phi} \tag{2.10}$$

A coordinate frame whose axis are defined along the tangent and normal directions to the ellipsoid's surface, as in Figure 2.1, is often useful for referencing the attitude of a body or vehicle. In this case, a East-North-Up (ENU) frame is chosen, whose axes and corresponding directions are also represented in the figure. The X and Y axes are tangent to the surface of the ellipsoid and point East and North, respectively. The origin of this local frame will be placed at the position of the reference antenna's phase center, as described later on. Expressions for the transformation of vector coordinates between the ECEF and local frame are simple to deduce, since the relative rotation between the frames is given by the geodetic parameters ϕ and λ.

As mentioned before, the body frame is the coordinate frame attached to the platform or vehicle whose attitude is to be estimated. The configuration of the antennas and auxiliary sensors in relation to the body frame, as well as the respective coordinate transformations between the body frame and local frame will be described in the next Chapter.

2.3 Global Navigation Satellite Systems

In the following, a brief description of a general Global Navigation Satellite System is given. The GPS [22] is highlighted, as the measurements used in the implementation and testing of the solution are the ones given by this particular constellation. Then, an overview of the signal processing that occurs in a typical GNSS receiver is presented, leading to a mathematical model of its measurements. Known error sources and the double differences used to reduce their effect are described.

2.3.1 System Description

Global Navigation Satellite System is a term referring to a system of satellites used to autonomously estimate positions and other quantities with global coverage. GNSS receivers are able to estimate the geodetic parameters relative to their position using radio signals transmitted from the satellites. Some examples of satellite constellations built for this purpose are the United States' GPS, Russia's GLONASS, China's BeiDou and the European Union's Galileo.

To achieve global coverage, the United States' constellation consists of at least 24 GPS satellites distributed across six equally-spaced orbital planes at an altitude of approximately 20 200 km [23]. Apart from the satellites composing the space segment, there is also a control segment with control and monitoring stations and the user segment, which includes the receivers and user applications.

Each GPS satellite transmits radio signals that are phase modulated according to a sequence of bits, a pseudorandom code, which is known by the receiver. The signal also contains navigation data that includes the time of transmission (TOT) of the code epoch and the ephemeris used to compute the satellite's position. The receiver uses the pseudorandom code knowledge to create a replica of the transmitted signal. Knowing the modulation of the carrier, it aligns the replica with the received signal. In practice, this can be done by maximizing the output of a filter that tries to undo the modulation of the carrier phase. After alignment, a Delay Lock Loop (DLL) maintains synchronization. For further details, see [24]. This signal processing is the basis of obtaining the pseudorange measurements.

The actual signal modulation is twofold. Two different bit sequences are used – the Coarse/Acquisition (C/A) code with a chipping rate of 1.023 MHz and the Precision (P) code which is 10 times faster. Furthermore, all GPS satellites use the same carrier frequencies. At present, they transmit at the L1, L2 and L5 frequencies, which are 1575.42 MHz, 1227.6 MHz and 1176.45 MHz, respectively. For distinguishability, each satellite uses a different pseudorandom sequence for modulation (PRN code).

Concerning the carrier phase measurements, other signal processing techniques are used to bring the received signal to lower frequencies and track the phase using phase locked loops (PLL). Such techniques fall out of the scope of the present work and are detailed extensively in the literature [24, 25]. Instead, our focus will be on the usage of these measurements to achieve high accuracy attitude estimates.

2.3.2 Observation Equations and Error Sources

The observables that the typical GNSS receiver provides, used to compute the accurate position of the antennas, are the pseudorange and carrier phase measurements. Before writing their mathematical models, the typical error sources affecting these measurements, as well as their approximate magnitude will be briefly discussed. All values are based on [3].

Starting from the transmitting end of the GNSS system, it must be mentioned that the satellite clocks, although precise up to a relative accuracy of 10^{-12}–10^{-13}, are not perfect. The accumulated error over a day multiplied by the speed of light can amount to several meters. Moreover, the orbital parameters transmitted by the satellite also contain small errors that grow until corrected periodically by a ground station. This deviation also accounts for a ranging error of several meters [26].

The largest errors are otherwise due to delay of the signal when traveling through the atmosphere. The first significant delay occurs when the signal crosses the ionosphere and is related to its electron density. One important detail is that the delay is inversely proportional to f^2, f being the frequency of the signal carrier, meaning that a dual-frequency receiver is capable of estimating the ionospheric delay by measuring, for example, on the L1 and L2 frequencies. The troposphere also changes the speed of the incoming signal due to its temperature, pressure and humidity. A simple model considers the tropospheric delay as varying with the elevation angle [27].

Using a low-cost receiver, both the ionospheric and tropospheric delays must be estimated using approximate models or its effects mitigated by means of differencing, as shown later on. Otherwise, the ranging error can increase by another few meters, which is critical in high accuracy applications.

The last significant errors related with the propagation of the signals are multipath errors. What arrives at a receiver is often the result of the transmitted signal reflected on different surfaces before ultimately arriving at the receiver through different paths. In this way, the signal interferes with itself. This effect is difficult to model, and so the antidote is often to improve the antenna design and the receiver site so as to decrease the multipath.

On the user segment level, the receiver clock used is much less precise than the very expensive satellite clocks. Unlike the satellite clock offsets which can be estimated using parameters transmitted in the GPS message, the receiver clock offset must be estimated along with the other unknowns in the measurement model. This necessity makes it so that the positioning problem using pseudorange data can only be solved within reasonable accuracy by gathering measurements from at least 4 satellites on each epoch.

The approximate ranging errors due to the discussed effects are presented in Table 2.1 as given in [3]. Assuming that the error sources are independent , the root mean square (RMS) corresponds to the user equivalent range error (UERE) shown in the table. Next, the adopted pseudorange and carrier phase models are described.

Table 2.1: Approximate magnitude of the most significant errors affecting GNSS measurements.

Error source	Ranging Error (m)
Satellite Clock	2
Satellite Ephemeris Data	2
Ionosphere	4
Troposphere	0.5–1
Multipath	0–2
UERE (RMS)	5

As mentioned before, the pseudorange is determined by comparing the receiver internally generated signal, the replica, with the signal received from the satellite. The delay between both signals is determined by aligning the known code modulation of the GNSS signal. For this reason, these pseudorange measurements are also called code measurements. Knowing the propagation velocity of the signal and its transmission time, the time delay is converted into a distance. One possible mathematical model of the pseudorange measurement [3] can be given by the following equation,

$$P_i^k(t) = \rho_i^k + c(dt_i(t) - dt^k(t-\tau_i^k)) + I_i^k + T_i^k + \epsilon_{P,i}^k, \tag{2.11}$$

where $P_i^k(t)$, is the pseudorange measurement at time t and $\rho_i^k = \left\|\boldsymbol{R}_3(\omega_e\tau_i^k)\boldsymbol{r}^k(t-\tau_i^k) - \boldsymbol{r}_i(t)\right\|$ is the distance between receiver i and satellite k. The position vectors of the receiver and satellite, $\boldsymbol{r}_i(t)$ and $\boldsymbol{r}^k(t-\tau_i^k)$, are written in the ECEF frame. The delay τ_i^k is the propagation time corresponding to the distance ρ_i^k, and the rotation matrix $\boldsymbol{R}_3(\omega_e\tau_i^k)$ accounts for the effect of the Earth's rotation, ω_e, when computing the distance between a receiver and satellite whose positions are evaluated at different times in the ECEF frame. In (2.11), $dt_i(t)$ and $dt^k(t-\tau_i^k)$ are the clock offsets of the receiver and satellite, respectively. The troposphere and ionosphere delays are given by T_i^k and I_i^k. Other errors in the modelling of the pseudorange, such as multipath effects, are grouped in the error term ϵ_i^k and will be discussed later.

The carrier phase range is computed by mixing the replica and received signals to obtain a measure of the phase of the incoming signal, given by the receivers in the form of a cycle count. Knowing the wavelength λ corresponding to the GNSS signal frequency (for the present work, consider only the L1 frequency $f = 1575.42\,\text{MHz}$), it is possible to convert the phase into a distance. However, the phase measurement is ambiguous by multiples of the wavelength since the exact cycle count between receiver and satellite cannot be determined by the signal processing of the receiver. Therefore, the carrier phase range can be given by

$$\Phi_i^k(t) = \rho_i^k + c(dt_i(t) - dt^k(t-\tau_i^k)) - I_i^k + T_i^k + \lambda(\phi_{i,0} - \phi_0^k) + \lambda n_i^k + \epsilon_{\Phi,i}^k, \tag{2.12}$$

where $\phi_{i,0}$ and ϕ_0^k are offset terms of the signal phase at the satellite transmitter and receiver, respectively, and n_i^k is the integer cycle ambiguity. The ionosphere effect delays the signal but introduces a phase advance, and so its contribution I_i^k to both measurements is opposite in sign, as seen in (2.11) and (2.12).

The carrier phase measurement is typically much more precise than the code measurements that provide the pseudorange. This makes the carrier phase measurements indispensable for applications where precision is of the essence, such as attitude estimation. However, the integer ambiguities must be determined.

2.3.3 Double Differenced Observations

High accuracy, when it comes to positioning and attitude estimation, is possible only when as many error sources as possible are either modelled or mitigated. One way of doing so is using several receivers. Since some error sources are satellite or receiver dependent, it results that by subtracting observations, some terms in the observation equations are eliminated.

Considering two receivers, i and j, simultaneously tracking two satellites, k and l, one can define the double differences (DD) both in terms of pseudorange and carrier phase range measurements as

$$P_{ij}^{kl} = P_i^k - P_i^l - P_j^k + P_j^l, \qquad \Phi_{ij}^{kl} = \Phi_i^k - \Phi_i^l - \Phi_j^k + \Phi_k^l. \tag{2.13}$$

Assuming that the measurements at the receivers are received at the same time epoch, the clock delays and the initial phases in (2.12) are eliminated since they only depend whether on the receiver or satellite and are considered constant over short time spans. Therefore, the double differenced observations can be modelled by

$$\begin{cases} P_{ij}^{kl} = \rho_{ij}^{kl} + I_{ij}^{kl} + T_{ij}^{kl} + \epsilon_{P,ij}^{kl} \\ \Phi_{ij}^{kl} = \rho_{ij}^{kl} + I_{ij}^{kl} + T_{ij}^{kl} + \lambda n_{ij}^{kl} + \epsilon_{\Phi,ij}^{kl} \end{cases}, \tag{2.14}$$

where the double differenced range, tropospheric and ionospheric delays, and integer ambiguities are defined similarly to the DD observations in (2.13).

If the antennas are considered to be sufficiently near to each other (less than 1–$2\mathrm{km}$) [3], as is the case for attitude estimation purposes, one can consider that the troposphere and ionosphere delays are approximately the same. That is, $T_i^k \approx T_i^l \approx T_j^k \approx T_k^l$ and $I_i^k \approx I_i^l \approx I_j^k \approx I_k^l$. The resulting DD measurements are written in (2.15) for any two antennas i, j and satellites k, l.

$$\begin{cases} P_{ij}^{kl} = \rho_{ij}^{kl} + E_{P,ij}^{kl} \\ \Phi_{ij}^{kl} = \rho_{ij}^{kl} + \lambda n_{ij}^{kl} + E_{\Phi,ij}^{kl} \end{cases} \tag{2.15}$$

In the next Chapter, these observations are related to the unknown body frame orientation and the problem that this thesis is central to is formulated.

Chapter 3

Problem Definition

In what follows, the models that link the available measurements to the unknown attitude are made explicit and the problem to be solved is described.

Consider a platform where a 3-axis gyroscope and $n+1$ GNSS antennas, together with the respective receivers, are mounted. The gyroscope and receivers are fixed to the platform, which is in turn static with respect to the body frame coordinate system, as represented in Figure 3.1. Antenna i is designated as the reference antenna. For simplicity, the corresponding GNSS receiver is also called reference receiver i. The origin of the body frame coincides with the phase center of the reference antenna, as shown in the Figure.

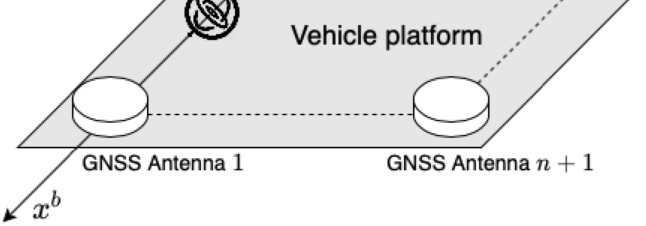


Figure 3.1: Antennas and gyroscope configuration in the body frame.

Consider also the local ENU frame centered at the phase center of the reference antenna and whose orientation is defined according to the WGS84 ellipsoid. The rotation between the body frame and the local frame can be described using the rotation matrix $\boldsymbol{R}_b^l$, where l denotes the local frame and b the body frame. As such, the local frame is considered the reference frame. Since $\boldsymbol{R}_b^l$ is the only rotation matrix in discussion, the subscript and superscript indicating the coordinate frames will be dropped. The transformation of vector coordinates between both frames can be written using either the rotation matrix or the corresponding unit quaternion as in (3.1), according to the conventions defined in the previous Section.

$$\boldsymbol{r}_l = \boldsymbol{R}_b^l \boldsymbol{r}_b \;, \qquad \boldsymbol{r}_l = \boldsymbol{q} \circ \boldsymbol{r}_b \circ \boldsymbol{q}^* . \tag{3.1}$$

Considering that the body frame rotates with respect to the reference frame with angular velocity $\boldsymbol{\omega}(t) = [p(t)\; q(t)\; r(t)]^\top \in \mathbb{R}^3$, measured in the body frame, the time derivative of the quaternion representing the orientation between the two frames is given by (3.2). This relation defines the attitude variation due to relative rotation between the two frames.

$$\dot{\boldsymbol{q}}(t) = \frac{1}{2}\begin{bmatrix} 0 \\ \boldsymbol{\omega}(t) \end{bmatrix} \circ \boldsymbol{q}(t) = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}(t))\boldsymbol{q}(t), \quad \boldsymbol{\Omega}(\boldsymbol{\omega}(t)) = \begin{bmatrix} 0 & -\boldsymbol{\omega}(t)^\top \\ \boldsymbol{\omega}(t) & -[\boldsymbol{\omega}(t)\times] \end{bmatrix} . \tag{3.2}$$

The 3-axis gyroscope placed on the platform gives a measure of the angular velocity $\boldsymbol{\omega}(t)$, albeit with some error. The measurements are stochastic and there are several models that try to describe its behaviour over time. The standard model used to describe the measurements of a three-axis rate-integrating gyro is given by (3.3) [19]. In these equations, $\boldsymbol{\omega}_m(t)$ denotes the gyro measurements and $\boldsymbol{b}(t)$ is the bias affecting these measurements, while $\boldsymbol{\omega}(t)$ is considered the true angular velocity.

$$\begin{cases} \boldsymbol{\omega}_m(t) = \boldsymbol{\omega}(t) + \boldsymbol{b}(t) + \boldsymbol{\eta}_v(t) \\ \dot{\boldsymbol{b}}(t) = \boldsymbol{\eta}_u(t) \end{cases} , \quad \begin{cases} \boldsymbol{\eta}_v(t) \sim N(\boldsymbol{0}, \sigma_v^2 \boldsymbol{I}) \\ \boldsymbol{\eta}_u(t) \sim N(\boldsymbol{0}, \sigma_u^2 \boldsymbol{I}) \end{cases} , \forall t \tag{3.3}$$

While this is a continuous-time model, in reality the angular velocity sampled by a gyroscope is given in discrete-time. This adjustment to the model will be made later on. At this point, the continuous-time model is sufficient to describe the problem.

Replacing the true angular velocity in (3.2) using the gyroscope measurements, one can write the quaternion kinematics as a function of the measured angular velocity (3.4).

$$\dot{\boldsymbol{q}}(t) = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_m(t) - \boldsymbol{b}(t))\boldsymbol{q}(t) - \frac{1}{2}\boldsymbol{\Gamma}(\boldsymbol{q}(t))\boldsymbol{\eta}_v(t), \quad \boldsymbol{\Gamma}(\boldsymbol{q}) = \begin{bmatrix} -q_x & -q_y & -q_z \\ q_w & -q_z & q_y \\ q_z & q_w & -q_x \\ -q_y & q_x & q_w \end{bmatrix} \tag{3.4}$$

Now, considering the model for the GNSS double differenced observations in (2.15), some approximations can be made. The geometry between the antennas placed on the platform and each satellite is represented in figure 3.2. For any pair of antennas i and j, the baseline can be defined as $\boldsymbol{b}_{ij} = \boldsymbol{r}_j - \boldsymbol{r}_i$, written in the local ENU frame. Assuming that the distance between the antennas is much shorter than the distance between each antenna and each satellite, that is, $\|\boldsymbol{b}_{ij}\| \ll \rho_i^k,\ \rho_j^k$, one can make the reasonable approximation of parallel links between each satellite and both receivers. Doing this, the single difference $\rho_{ij}^k = \rho_i^k - \rho_j^k$ can be written as

$$\rho_{ij}^k \approx \boldsymbol{u}_i^k \cdot \boldsymbol{b}_{ij}, \quad \boldsymbol{u}_i^k = \frac{\boldsymbol{R}_3(\omega_e \tau_i^k)\boldsymbol{r}^k - \boldsymbol{r}_i}{\rho_i^k}, \tag{3.5}$$

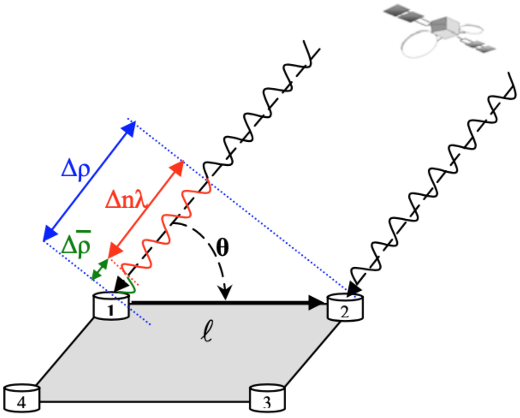


Figure 3.2: Approximation of parallel signals for very short baseline. [5]

where $\boldsymbol{u}_i^k$ is the line of sight (LOS) vector between receiver i and satellite k. Due to the short baseline assumption, it is also assumed that the LOS vector is approximately the same for every antenna, given any one satellite, and so the subscript i in $\boldsymbol{u}_i^k$ is dropped from here on. The previous observations model (2.15) is then rewritten as (3.6).

$$\begin{cases} P_{ij}^{kl} = (\boldsymbol{u}^k - \boldsymbol{u}^l)\boldsymbol{b}_{ij} + E_{P,ij}^{kl} \\ \Phi_{ij}^{kl} = (\boldsymbol{u}^k - \boldsymbol{u}^l)\boldsymbol{b}_{ij} + \lambda n_{ij}^{kl} + E_{\Phi,ij}^{kl} \end{cases} \tag{3.6}$$

The model can now be expanded to fit observations from two receivers i and j tracking $m+1$ satellites (3.7).

$$\boldsymbol{y} = \boldsymbol{S}\boldsymbol{n} + \boldsymbol{U}\boldsymbol{b} + \boldsymbol{\epsilon} \quad ; \quad \boldsymbol{y} \in \mathbb{R}^{2m}, \quad \boldsymbol{n} \in \mathbb{Z}^m, \quad \boldsymbol{b} \in \mathbb{R}^3 \tag{3.7}$$

Among $m+1$ satellites and two receivers, only m independent differences can be made. Choosing the reference satellite as satellite k, $\boldsymbol{y}$ is the vector of double differenced independent observations $\boldsymbol{y} = [P_{ij}^{k1}, \dots, P_{ij}^{km}, \Phi_{ij}^{k1}, \dots, \Phi_{ij}^{km}]^\top$, and $\boldsymbol{n}$ contains the corresponding DD integer ambiguities. Matrices $\boldsymbol{S}$ and $\boldsymbol{U}$ are defined in (3.8). The LOS vectors are contained in $\boldsymbol{\Upsilon}$.

$$\boldsymbol{S} = \begin{bmatrix} \boldsymbol{0}_m \\ \lambda \boldsymbol{I}_m \end{bmatrix}, \quad \boldsymbol{U} = \begin{bmatrix} \boldsymbol{\Upsilon} \\ \boldsymbol{\Upsilon} \end{bmatrix}, \quad \boldsymbol{\Upsilon} = \begin{bmatrix} \boldsymbol{u}^k - \boldsymbol{u}^1 \\ \vdots \\ \boldsymbol{u}^k - \boldsymbol{u}^m \end{bmatrix} \tag{3.8}$$

To characterize the DD measurements error $\boldsymbol{\epsilon}$, the single measurements will be considered uncorrelated across different receivers and satellites. The code measurements are modelled as Gaussian random variables, uncorrelated in time. As such, the covariance matrix of a vector of code measurements at a single epoch can be written as $\mathrm{Var}([P_i^1, \dots, P_i^{m+1}, \dots, P_j^1, \dots, P_j^{m+1}]^\top) = \boldsymbol{Q_P}$, where $\boldsymbol{Q_P} = \mathrm{diag}\left[(\sigma_{P,i}^1)^2, \dots, (\sigma_{P,i}^{m+1})^2, \dots, (\sigma_{P,j}^1)^2, \dots, (\sigma_{P,j}^{m+1})^2\right]$. This definition is analogous for the carrier phase measurements, with covariance matrix $\mathrm{Var}([\Phi_i^1, \dots, \Phi_i^{m+1}, \dots, \Phi_j^1, \dots, \Phi_j^{m+1}]^\top) = \boldsymbol{Q_\Phi}$. However, differencing techniques introduce correlation between the measurements, increasing the noise. Using the differencing matrix $\boldsymbol{D}$ defined in (3.9), and defining $\boldsymbol{1}_m = [1, 1, \dots, 1]^\top$, the covariance matrix of the

DD pseudorange observations follows in (3.10).

$$\boldsymbol{D} = \begin{bmatrix}\boldsymbol{1}_m, -\boldsymbol{I}_m\end{bmatrix} \quad (3.9) \qquad \mathrm{Var}([P_{ij}^{k1}, \dots, P_{ij}^{km}]^\top = 2\boldsymbol{D}\boldsymbol{Q_P}\boldsymbol{D}^\top \quad (3.10)$$

The covariance matrix of the measurements vector $\boldsymbol{y}$ in (3.7) can now be written as in (3.11).

$$\boldsymbol{Q_{yy}} = \mathrm{Var}(\boldsymbol{y}) = \begin{bmatrix} 2\boldsymbol{D}\boldsymbol{Q_P}\boldsymbol{D}^\top & \boldsymbol{0} \\ 0 & 2\boldsymbol{D}\boldsymbol{Q_\Phi}\boldsymbol{D}^\top \end{bmatrix} \tag{3.11}$$

At last, the measurement model can once again be extended to consider $n+1$ receivers tracking $m+1$ satellites. Building on the previous single baseline model, the multi-baseline model is written in (3.12).

$$\begin{cases} \boldsymbol{Y} = \boldsymbol{S}\boldsymbol{N} + \boldsymbol{U}\boldsymbol{B} + \boldsymbol{\Xi} \quad ; \quad \boldsymbol{Y} \in \mathbb{R}^{2m\times n}, \quad \boldsymbol{N} \in \mathbb{Z}^{m\times n}, \quad \boldsymbol{B} \in \mathbb{R}^{3\times n} \\ \mathrm{Var}(vec(\boldsymbol{Y})) = \boldsymbol{Q_{YY}} \end{cases} \tag{3.12}$$

The baseline matrix $\boldsymbol{B}$ contains the n independent baselines $\boldsymbol{B} = [\boldsymbol{b}_{i1}, \boldsymbol{b}_{i2}, \dots, \boldsymbol{b}_{in}]$, choosing the reference receiver i. Each column of the measurements $\boldsymbol{Y}$ and the ambiguities $\boldsymbol{N}$ refers to the baseline of the corresponding column of $\boldsymbol{B}$. The $vec()$ operator stacks the columns of the matrix it is applied to, allowing in this case to write the covariance matrix of the measurements $\boldsymbol{Y}$. The new measurements are correlated even further by the common receiver i in the baselines – the expression for $\boldsymbol{Q_{YY}}$ is presented as (3.13), where $\otimes$ is the Kronecker product.

$$\boldsymbol{Q_{YY}} = \begin{bmatrix} \boldsymbol{P}_n \otimes 2\boldsymbol{D}\boldsymbol{Q_P}\boldsymbol{D}^\top & \boldsymbol{0} \\ 0 & \boldsymbol{P}_n \otimes 2\boldsymbol{D}\boldsymbol{Q_\Phi}\boldsymbol{D}^\top \end{bmatrix}, \quad \boldsymbol{P}_n = \begin{bmatrix} 1 & 0.5 & \dots & 0.5 \\ 0.5 & 1 & & \vdots \\ \vdots & & \ddots & 0.5 \\ 0.5 & \dots & 0.5 & 1 \end{bmatrix}_{(n\times n)} \tag{3.13}$$

From this point, there are overall two ways of estimating the attitude of the body frame in relation to the reference frame.

The first consists of using the baseline model (3.13) to estimate the baseline coordinates in the local frame, $\boldsymbol{B}$. In this case, the problem to solve can be formulated as a minimization of the squared error

$$\langle \check{\boldsymbol{N}}, \check{\boldsymbol{B}} \rangle = \underset{\boldsymbol{N}\in\mathbb{Z}^{m\times n}, \boldsymbol{B}\in\mathbb{R}^{3\times n}}{\arg\min} \|\boldsymbol{Y} - \boldsymbol{S}\boldsymbol{N} - \boldsymbol{U}\boldsymbol{B}\|^2_{\boldsymbol{Q_{YY}}}, \tag{3.14}$$

where the squared norm is in this case defined as

$$\|\boldsymbol{A}\|^2_{\boldsymbol{Q}} := \left\|\boldsymbol{Q}^{-\frac{1}{2}}\boldsymbol{A}\boldsymbol{Q}^{-\frac{1}{2}}\right\|^2_F, \tag{3.15}$$

with $\|\boldsymbol{A}\|^2_F$ being the Frobenius norm, for a general matrix $\boldsymbol{A}$. The type of minimization problem in (3.14) has been called a Integer Least Square (ILS) due to the integer constraint [16, 28, 29]. After finding the baseline solution $\check{\boldsymbol{B}}$, estimating the attitude is a matter of solving a minimization problem that can be formulated, for example, as Wahba's problem [30]. Denoting the known baseline coordinates in the body frame as $\boldsymbol{F} = [\boldsymbol{l}_{i1}, \boldsymbol{l}_{i2}, \ldots, \boldsymbol{l}_{in}]$, the cost function to minimize is

$$J(\boldsymbol{R}) = \frac{1}{2} \sum_{k=1}^{n} a_k \left\| \check{\boldsymbol{b}}_{ik} - \boldsymbol{R}\boldsymbol{l}_{ik} \right\|^2 , \tag{3.16}$$

with $\boldsymbol{R}$ being the 3×3 rotation matrix that describes the body frame to reference frame relative orientation. Several methods are discussed in the literature to solve this problem [31].

The second formulation, and the one used in this work, consists of using the GNSS-based attitude model [8]. The model corresponds to replacing the baseline coordinates written in the reference frame by the rotated baselines written in the body frame. That is, replacing $\boldsymbol{B} = \boldsymbol{R}\boldsymbol{F}$, the GNSS-based attitude model is written in (3.17).

$$\begin{cases} \boldsymbol{Y} = \boldsymbol{S}\boldsymbol{N} + \boldsymbol{U}\boldsymbol{R}\boldsymbol{F} + \boldsymbol{\Xi} \quad ; \quad \boldsymbol{Y} \in \mathbb{R}^{2m \times n}, \quad \boldsymbol{N} \in \mathbb{Z}^{m \times n}, \quad \boldsymbol{B} \in \mathbb{R}^{3 \times n}, \\ \mathrm{Var}(vec(\boldsymbol{Y})) = \boldsymbol{Q}_{\boldsymbol{YY}}. \end{cases} \tag{3.17}$$

Then, the optimization problem becomes the one in (3.18). This problem has additional constraints compared to (3.14) due to the orthonormality property of rotation matrices. It has been named as a Constrained Integer Least Squares (C-ILS) problem in the literature [8].

$$\langle \check{\boldsymbol{N}}, \check{\boldsymbol{R}} \rangle = \underset{\boldsymbol{N} \in \mathbb{Z}^{m \times n}, \boldsymbol{R} \in SO(3)}{\arg \min} \left\| \boldsymbol{Y} - \boldsymbol{S}\boldsymbol{N} - \boldsymbol{U}\boldsymbol{R}\boldsymbol{F} \right\|^2_{\boldsymbol{Q}_{\boldsymbol{YY}}} . \tag{3.18}$$

This thesis proposes to solve this optimization problem by using also the gyroscope measurements introduced in the attitude kinematics model (3.4), in a recursive manner. This framework is presented in the next Chapter.

Chapter 4

Attitude Estimation

The proposed method for attitude estimation, solving the problem formulated in the previous Chapter, is built on the work presented by Teunissen in [16]. The separation of the original problem in three parts, consisting of float solution, integer ambiguity determination and the fixing of the solution, is presented in the first Section.

Based on this division, the general framework proposed in this research to solve this problem is then presented in the following Section.

The method of obtaining the float solution using both the GNSS observations and the gyroscope measurements in a recursive way is presented in sections 4.3 and 4.4. A study of the observability of the linearized system is also included.

Lastly, the integer ambiguity determination method used is described in Section 4.5. The changes made to the original MC-LAMBDA algorithm due to using an EKF for obtaining the float solution are described. The solution fixing is also adapted to the chosen techniques, detailed in Section 4.6.

4.1 The Least-Squares Approach

The GNSS-based attitude model used in the minimization problem (3.18) is introduced in [16], along with a way of solving it. As mentioned before, this is a C-ILS problem and so there is as of yet no closed form solution. Because of the integer nature of the ambiguities, the methods to solve it always involve some kind of search algorithm, as will be seen later on. The method proposed in this thesis takes advantage of the same orthogonal decomposition of the objective function as in [16], and so this process will be briefly described.

First, consider the minimization problem formulated in (4.1), where the $vec()$ operator is applied to the original observation model in (3.18). Note that since the model is now in column vector form, the norm $\|(.)\|$ represents the Euclidean or L^2 norm. In the case of a weighted norm, it is computed as $\|(.)\|^2_{\boldsymbol{Q}} = (.)^\top \boldsymbol{Q}^{-1}(.)$.

$$\langle \check{\boldsymbol{N}}, \check{\boldsymbol{R}} \rangle = \underset{\boldsymbol{N} \in \mathbb{Z}^{m \times n}, \boldsymbol{R} \in SO(3)}{\arg \min} \|vec(\boldsymbol{Y} - \boldsymbol{S}\boldsymbol{N} - \boldsymbol{U}\boldsymbol{R}\boldsymbol{F})\|^2_{\boldsymbol{Q}_{\boldsymbol{YY}}} . \tag{4.1}$$

Then, an orthogonal decomposition of the squared error is done, as in [28, 29]. Applying some properties of the $vec(.)$ operator [29], the squared norm can be decomposed as written in (4.2),

$$\begin{aligned}
&\min_{\boldsymbol{N}\in\mathbb{Z}^{m\times n},\boldsymbol{R}\in SO(3)} \|vec(\boldsymbol{Y}-\boldsymbol{S}\boldsymbol{N}-\boldsymbol{U}\boldsymbol{R}\boldsymbol{F})\|^2_{\boldsymbol{Q}_{\boldsymbol{YY}}} = \\
&= \min \|vec(\boldsymbol{Y}-\boldsymbol{S}\hat{\boldsymbol{N}}-\boldsymbol{U}\hat{\boldsymbol{R}}\boldsymbol{F})\|^2_{\boldsymbol{Q}_{\boldsymbol{YY}}} + \min_{\boldsymbol{N}\in\mathbb{Z}^{m\times n}} \Big(\|vec(\hat{\boldsymbol{N}}-\boldsymbol{N})\|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}} + \\
&+ \min_{\boldsymbol{R}\in SO(3)} \|vec(\hat{\boldsymbol{R}}(\boldsymbol{N})-\boldsymbol{R})\|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}} \Big),
\end{aligned} \tag{4.2}$$

where $\hat{\boldsymbol{N}}$ and $\hat{\boldsymbol{R}}$ are the float solutions, defined as the solution of the minimization problem in which the integer constraints on $\boldsymbol{N}$ and nonlinear constraints on $\boldsymbol{R}$ are dropped (4.3).

$$\langle \hat{\boldsymbol{N}}, \hat{\boldsymbol{R}} \rangle = \arg \min_{\boldsymbol{N}\in\mathbb{R}^{m\times n},\boldsymbol{R}\in\mathbb{R}^{3\times 3}} \|vec(\boldsymbol{Y}-\boldsymbol{S}\boldsymbol{N}-\boldsymbol{U}\boldsymbol{R}\boldsymbol{F})\|^2_{\boldsymbol{Q}_{\boldsymbol{YY}}} \tag{4.3}$$

The covariance matrices $\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}$ and $\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N}}$ in (4.2) are relative to $\hat{\boldsymbol{N}}$ and the conditional solution $\hat{\boldsymbol{R}}(\boldsymbol{N})$, respectively. The conditional solution is obtained solving a least-squares problem where it is assumed that the integer ambiguities $\boldsymbol{N}$ are known but the orthonormality constraints are dropped, formulated in (4.4).

$$\hat{\boldsymbol{R}}(\boldsymbol{N}) = \arg \min_{\boldsymbol{R}\in\mathbb{R}^{3\times 3}} \|vec(\boldsymbol{Y}-\boldsymbol{S}\boldsymbol{N}-\boldsymbol{U}\boldsymbol{R}\boldsymbol{F})\|^2_{\boldsymbol{Q}_{\boldsymbol{YY}}} \tag{4.4}$$

A link can be made between the float solutions in (4.3) and the conditional solution (4.4) if both minimization problems are solved using the standard weighted least-squares (WLS) normal equations. For completeness, the normal equations of a WLS problem shall be described. Let $\boldsymbol{r} = \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}$ be the residual one wishes to minimize, with $\boldsymbol{x} \in \mathbb{R}^p$ the parameters to estimate, $\boldsymbol{y}$ the observations and $\boldsymbol{\Sigma}_{\boldsymbol{y}}$ the respective covariance matrix. The WLS solution is the result of the minimization of the squared error function (4.5). The solution, estimate $\hat{\boldsymbol{x}}$, and its covariance matrix $\boldsymbol{\Sigma}_{\hat{\boldsymbol{x}}}$ are written as (4.6) and (4.7) [3].

$$\hat{\boldsymbol{x}} = \arg \min_{\boldsymbol{x}\in\mathbb{R}^p} \|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|^2_{\boldsymbol{\Sigma}_{\boldsymbol{y}}} \tag{4.5}$$

$$\hat{\boldsymbol{x}} = \left(\boldsymbol{A}^\top \boldsymbol{\Sigma}_{\boldsymbol{b}}^{-1} \boldsymbol{A}\right)^{-1} \boldsymbol{A}^\top \boldsymbol{\Sigma}_{\boldsymbol{b}}^{-1} \boldsymbol{b} \tag{4.6}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{x}}} = \left(\boldsymbol{A}^\top \boldsymbol{\Sigma}_{\boldsymbol{b}}^{-1} \boldsymbol{A}\right)^{-1} \tag{4.7}$$

Solving both minimization problems (4.3) and (4.4) in this way and replacing terms using the expressions of the float solution, the conditional solution $\hat{\boldsymbol{R}}(\boldsymbol{N})$ can be written as a function of $\hat{\boldsymbol{N}}$ and $\hat{\boldsymbol{R}}$ (4.8).

$$vec(\hat{\boldsymbol{R}}(\boldsymbol{N})) = vec(\hat{\boldsymbol{R}}) - \boldsymbol{Q}_{\hat{\boldsymbol{R}}\hat{\boldsymbol{N}}} \boldsymbol{Q}^{-1}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}} vec(\hat{\boldsymbol{N}}-\boldsymbol{N}) \tag{4.8}$$

The corresponding covariance matrix $\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}$ is computed by applying the covariance propagation law, resulting in

$$\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})} = \boldsymbol{Q}_{\hat{\boldsymbol{R}}\hat{\boldsymbol{R}}} - \boldsymbol{Q}_{\hat{\boldsymbol{R}}\hat{\boldsymbol{N}}} \boldsymbol{Q}^{-1}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}} \boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{R}}}. \tag{4.9}$$

The covariance matrices $\boldsymbol{Q}_{\hat{\boldsymbol{R}}\hat{\boldsymbol{R}}}$, $\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}$ and $\boldsymbol{Q}_{\hat{\boldsymbol{R}}\hat{\boldsymbol{N}}}$ are obtained by error propagation of the least-squares estimates in (4.3), which corresponds to using equation (4.7). Note that $\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{R}}} = \boldsymbol{Q}^\top_{\hat{\boldsymbol{R}}\hat{\boldsymbol{N}}}$, as per the definition of the cross-covariance.

Returning to the original problem, the minimization in (4.2) is taken with respect to the matrix of integer DD ambiguities $\boldsymbol{N}$ and rotation matrix $\boldsymbol{R}$. From equation (4.2), and having the float solutions, the integer estimator of $\boldsymbol{N}$ is then given by

$$\check{\boldsymbol{N}} = \underset{\boldsymbol{N} \in \mathbb{Z}^{m \times n}}{\arg\min} \left(\| vec(\hat{\boldsymbol{N}} - \boldsymbol{N}) \|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}} + \| vec(\hat{\boldsymbol{R}}(\boldsymbol{N}) - \check{\boldsymbol{R}}(\boldsymbol{N})) \|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}} \right), \tag{4.10}$$

with $\check{\boldsymbol{R}}(\boldsymbol{N})$ the solution of

$$\check{\boldsymbol{R}}(\boldsymbol{N}) = \underset{\boldsymbol{R} \in SO(3)}{\arg\min} \; \| vec(\hat{\boldsymbol{R}}(\boldsymbol{N}) - \boldsymbol{R}) \|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}}. \tag{4.11}$$

If $\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}$ turns out to be a scaled unit matrix or is diagonal, (4.11) is an Orthogonal Procrustes Problem (OPP) [13], a particular case of Wahba's problem [31]. There are several closed form solutions available to solve this type of problem, such as the Quaternion Estimator (QUEST) method [14] or the Fast Optimal Attitude Matrix (FOAM) [15], to name a few. However, if $\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}$ is fully populated, as is often the case [29], the solution of (4.11) has to be computed numerically. Several ways of doing so are also presented in the work by Teunissen and Georgi [8]. The technique adopted in the present case will be detailed later in this document.

The first term in (4.10) measures the distance between the ambiguities' float solution and its nearest integer matrix, weighted according to the covariance matrix of the float solution. The second term weights the difference between the rotation matrix conditional solution and its orthonormal "projection". Thus, this minimization returns the integer ambiguities that best fit the configuration between the antennas and their lengths, since this knowledge is implicit in the rotation matrix. Compared to the LAMBDA problem, also introduced by Teunissen [12], this multivariate constrained method of determining the integer ambiguities is expected to have a higher success rate.

The MC-LAMDBA method detailed in [8] proposes solving the complex ILS problem (4.10) in an efficient way using bounding cost functions to decrease the computational load of evaluating the original cost function a great number of times. In fact, each evaluation of the objective function in (4.10) entails solving the nonlinear estimation problem (4.11), which is to be avoided if possible. A detailed description of this algorithm is presented later in Section 4.5.

In the next Section, an overview is given of the overall method proposed in this thesis to solve the problem in (4.1). It builds on the decomposition of the squared error in (4.2), solving first for the float solution and then determining the integer ambiguities defined in (4.10).

4.2 Proposed Framework

In the previous Section it was discussed how, due to the decomposition of the squared error, the first step to solving the minimization problem consists of determining the float solutions $\hat{\boldsymbol{N}}$ and $\hat{\boldsymbol{R}}$. In the typical approach, they are computed using the WLS normal equations. This is a static method in the sense that it solves for the float solutions in an epoch-by-epoch basis. However, in the problem definition it was assumed that the attitude kinematics of the body are known and can be written using angular velocity measurements, provided by a 3-axis rate-gyro. Using this information, the float solution can be computed in a recursive way by applying an EKF, as detailed in Section 4.4. One way of thinking about the Kalman Filter (KF) is as a least-squares estimate that is updated as new information, or measurements, are acquired [3]. As such, using a recursive filter to compute the float solutions should still allow the usage of the decomposition in (4.2).

Next to obtaining the float solutions, the integer DD ambiguities $\check{N}$ in (4.10) are determined using the MC-LAMBDA method.

After obtaining the correct integer ambiguities, the float solutions' accuracy can be improved by using the fixed ambiguities as measurements and feeding them to the filter, due to the structure of the EKF. Caution must be taken when feeding the fixed ambiguities to the filter since they will increase the chance that the solution of the MC-LAMBDA in the next time step coincides with the fixed ambiguities of the previous step. If the first set of fixed ambiguities is incorrect, it might take some time for the algorithm to converge to the right solution. Therefore, there is a time period where MC-LAMBDA solutions are collected, before selecting the one most likely to be correct at a time step $t = t_{fix}$. Then, the conditional solution of this set of ambiguities is fed to a second update step of the EKF. This is the method of solution fixing described in Section 4.6. In the time steps following the fixing, the solution fed to the second update step is instead taken directly from the MC-LAMBDA, since the ambiguities have already been fixed.

A representation of the several steps done in order to solve problem (4.1), as enumerated before, is presented in Figure 4.1. In the following sections, each part of the framework is detailed.

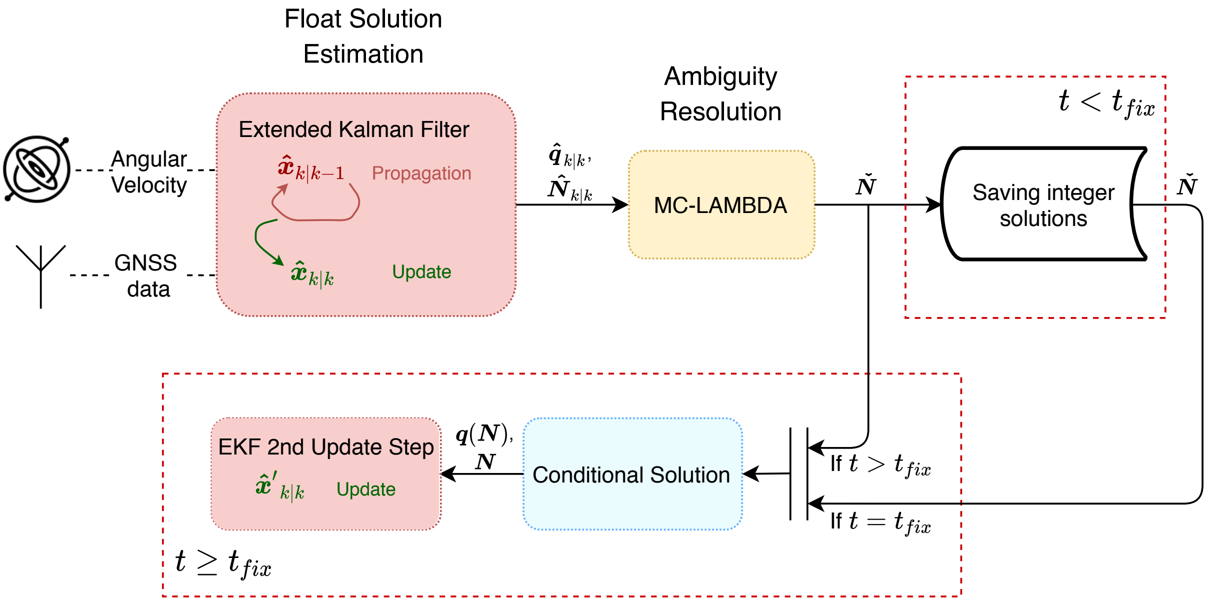


Figure 4.1: Flowchart of the proposed attitude estimation solution.

4.3 Nominal System Model

In the following, the system dynamics and observation model on which the EKF is based are formally derived. Before writing the equations of the nonlinear system in discrete-time, the quaternion kinematics are discretized.

4.3.1 Discrete Quaternion Kinematics

The continuous-time quaternion kinematic equations introduced in (3.2) assume that the angular velocity $\boldsymbol{\omega}$ is known at every time instant. However, in practice the angular velocity measurements are sampled in discrete-time by the gyroscope. To be able to use these measurements to predict the body attitude, the quaternion kinematics model needs to be written in discrete-time.

Whitmore [32] presents a closed-form integrator for the quaternion kinematics equation, which can be written in a recursive form appropriate for numerical integration. This method consists of finding an analytical solution for the differential equation (3.2) by introducing the integrating factor (4.12). The anti-symmetric matrix $\boldsymbol{\Omega}(\boldsymbol{\omega}(t))$ was defined back in (3.2).

$$\boldsymbol{I}(t,t_0)=\exp\left[\int_{t_0}^{t}-\frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}(t))dt\right] \tag{4.12}$$

By applying the integrating factor to the kinematics equation (3.2), integrating from a time t_0 to time t, expanding the exponential in Maclaurin series form and simplifying it, a closed-form solution of the kinematics equation is found (4.13), with $\boldsymbol{\delta\theta}(t,t_0)$ and $\Theta(t,t_0)$ written in (4.14).

$$\boldsymbol{q}(t)=\boldsymbol{\Psi}(t,t_0)\boldsymbol{q}(t_0)=\left(\cos\frac{\|\boldsymbol{\delta\vartheta}(t,t_0)\|}{2}\boldsymbol{I}_4+\frac{2}{\|\boldsymbol{\delta\vartheta}(t,t_0)\|}\sin\frac{\|\boldsymbol{\delta\vartheta}(t,t_0)\|}{2}\boldsymbol{\Theta}(t,t_0)\right)\boldsymbol{q}(t_0) \tag{4.13}$$

$$\begin{gathered}\boldsymbol{\delta\vartheta}(t,t_0)=\begin{bmatrix}\delta\phi(t,t_0)\\ \delta\theta(t,t_0)\\ \delta\psi(t,t_0)\end{bmatrix},\quad \boldsymbol{\Theta}(t,t_0)=\frac{1}{2}\begin{bmatrix}0 & -\boldsymbol{\delta\vartheta}(t,t_0)^\top\\ \boldsymbol{\delta\vartheta}(t,t_0) & -[\boldsymbol{\delta\vartheta}(t,t_0)\times]\end{bmatrix}\\ \delta\phi(t,t_0)=\int_{t_0}^{t}p(t)dt,\quad \delta\theta(t,t_0)=\int_{t_0}^{t}q(t)dt,\quad \delta\psi(t,t_0)=\int_{t_0}^{t}r(t)dt\end{gathered} \tag{4.14}$$

It is simple to see that the closed-form solution in (4.13) can be made recursive between instants t_k and t_{k+1} by replacing $t_0=t_k$ and $t=t_{k+1}$. Then, the closed-form solution for the quaternion kinematics in discrete-time form is given in (4.15), where $\bar{\boldsymbol{\Psi}}_{k+1,k}$ is the quaternion's transition matrix and all the needed definitions are written in (4.16). Note that the subscripts $(\cdot)_k$, $(\cdot)_{k+1}$ or $(\cdot)_{k+1,k}$ denote the time instants, for example t_k, at which each variable is evaluated.

$$\boldsymbol{q}_{k+1}=\boldsymbol{\Psi}_{k+1,k}\,\boldsymbol{q}_k=\left(\cos\frac{\|\boldsymbol{\delta\vartheta}_{k+1,k}\|}{2}\boldsymbol{I}_4+\frac{2}{\|\boldsymbol{\delta\vartheta}_{k+1,k}\|}\sin\frac{\|\boldsymbol{\delta\vartheta}_{k+1,k}\|}{2}\boldsymbol{\Theta}_{k+1,k}\right)\boldsymbol{q}_k \tag{4.15}$$

$$
\begin{aligned}
&\boldsymbol{\delta\vartheta}_{k+1,k} = \begin{bmatrix} \delta\phi_{k+1,k} \\ \delta\theta_{k+1,k} \\ \delta\psi_{k+1,k} \end{bmatrix}, \quad \boldsymbol{\Theta}_{k+1,k} = \frac{1}{2}\begin{bmatrix} 0 & -\boldsymbol{\delta\vartheta}_{k+1,k}^\top \\ \boldsymbol{\delta\vartheta}_{k+1,k} & -[\boldsymbol{\delta\vartheta}_{k+1,k}\times] \end{bmatrix} \\
&\delta\phi_{k+1,k} = \int_{t_k}^{t_{k+1}} p(t)dt, \quad \delta\theta_{k+1,k} = \int_{t_k}^{t_{k+1}} q(t)dt, \quad \delta\psi_{k+1,k} = \int_{t_k}^{t_{k+1,k}} r(t)dt
\end{aligned} \tag{4.16}
$$

It is proven that the quaternion kinematics using $\boldsymbol{\Psi}_{k+1,k}$ maintains the norm of the quaternion, as required. That is, $\boldsymbol{q}_{k+1}^\top \boldsymbol{q}_{k+1} = \boldsymbol{q}_k^\top \boldsymbol{q}_k$. This can be easily seen by rewriting the discrete-time quaternion kinematics (4.15) as a quaternion product. Such is achieved by taking into account the property introduced in (2.9), from which (4.17) follows, where it becomes clear that the norm of $\tilde{\boldsymbol{q}}_{k+1,k}$ is unitary.

$$
\boldsymbol{q}_{k+1} = \boldsymbol{\Psi}_{k+1,k}\,\boldsymbol{q}_k = \boldsymbol{q}_k \circ \tilde{\boldsymbol{q}}_{k+1,k}, \quad \tilde{\boldsymbol{q}}_{k+1,k} = \begin{bmatrix} \frac{\boldsymbol{\delta\vartheta}_{k+1,k}}{\|\boldsymbol{\delta\vartheta}_{k+1,k}\|} \sin\frac{\|\boldsymbol{\delta\vartheta}_{k+1,k}\|}{2} \\ \cos\frac{\|\boldsymbol{\delta\vartheta}_{k+1,k}\|}{2} \end{bmatrix} \tag{4.17}
$$

As a final note, a different approach is followed in [20], which is based in a Taylor series expansion of (3.2) around time instant t_k ($\boldsymbol{q}_k$), for computing the value at time t_{k+1} ($\boldsymbol{q}_{k+1}$). Assuming that the angular velocity is constant during the time step between t_k and t_{k+1}, a very similar expression to the one in (4.17) is found, the only difference being due to the method used to numerically integrate the angular velocity.

The motivation behind the discretization of the kinematics equation is being able to use the gyroscope measurements. As such, the discrete angular velocity measurements taken from the gyroscope must also be modelled. The discrete-time equivalent of the continuous-time gyroscope model (3.3) is given in (4.18), as presented in [33]. It is defined as a function of the noise parameters σ_v and σ_u written in the continuous-time model. As before, the k subscript denotes the value at the k-th time step, and both $\boldsymbol{z}_{v_k}$ and $\boldsymbol{z}_{u_k}$ are white Gaussian noise processes with covariance matrix given by the identity matrix.

$$
\begin{aligned}
&\begin{cases} \boldsymbol{\omega}_{m_{k+1}} = \boldsymbol{\omega}_{k+1} + \bar{\boldsymbol{b}}_{k+1,k} + \boldsymbol{w}_{v_{k+1}} \\ \boldsymbol{b}_{k+1} = \boldsymbol{b}_k + \boldsymbol{w}_{u_{k+1}} \end{cases}, \\
&\bar{\boldsymbol{b}}_{k+1,k} = \frac{1}{2}(\boldsymbol{b}_{k+1} + \boldsymbol{b}_k), \quad \boldsymbol{w}_{v_{k+1}} = \left(\frac{\sigma_v^2}{\Delta t} + \frac{1}{12}\sigma_u^2 \Delta t\right)^{1/2} \boldsymbol{z}_{v_{k+1}}, \quad \boldsymbol{w}_{u_{k+1}} = \sigma_u \Delta t^{1/2} \boldsymbol{z}_{u_{k+1}}
\end{aligned} \tag{4.18}
$$

To introduce the gyroscope measurements in model (4.15), the angular rates integrals in (4.16) must be evaluated approximately since the angular velocity measurements are only available at time instants t_k and $t_{k+1}, \forall k \geq 0$. Using the trapezoid rule over a time step $\Delta t = t_{k+1} - t_k$ allows computing the approximate integrated angular velocities as in (4.19).

$$
\delta\bar{\phi}_{k+1,k} = \frac{\Delta t}{2}(p_k + p_{k+1}), \quad \delta\bar{\theta}_{k+1,k} = \frac{\Delta t}{2}(q_k + q_{k+1}), \quad \delta\bar{\psi}_{k+1,k} = \frac{\Delta t}{2}(r_k + r_{k+1}) \tag{4.19}
$$

The discrete-time gyroscope measurements model (4.18) can be used to write the exact angular velocity as $\boldsymbol{\omega}_{k+1} = \boldsymbol{\omega}_{m_{k+1}} - \bar{\boldsymbol{b}}_{k+1,k} - \boldsymbol{w}_{v_{k+1}}$. Replacing in (4.19), the integrated angular velocity can now be written as (4.20). The resulting skew-symmetric matrix $\bar{\boldsymbol{\Theta}}_{k+1,k}$ is now written from the integrated measured angular velocities (4.21).

$$\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k} = \frac{\Delta t}{2}\left[(\boldsymbol{\omega}_{m_k} - \bar{\boldsymbol{b}}_{k,k-1} - \boldsymbol{w}_{v_k}) + (\boldsymbol{\omega}_{m_{k+1}} - \bar{\boldsymbol{b}}_{k+1,k} - \boldsymbol{w}_{v_{k+1}})\right] \tag{4.20}$$

$$\bar{\boldsymbol{\Theta}}_{k+1,k} = \frac{1}{2}\begin{bmatrix} 0 & -\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}^{\top} \\ \boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k} & -[\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\times] \end{bmatrix} \tag{4.21}$$

Then, following the same approach as before, the model of the quaternion kinematics in discrete-time using gyro measurements is given by (4.22), with the previously defined $\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}$ and $\bar{\boldsymbol{\Theta}}_{k+1,k}$.

$$\boldsymbol{q}_{k+1} = \bar{\boldsymbol{\Psi}}_{k+1,k}\ \boldsymbol{q}_k = \left(\cos\frac{\|\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\|}{2}\boldsymbol{I}_4 + \frac{2}{\|\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\|}\sin\frac{\|\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\|}{2}\bar{\boldsymbol{\Theta}}_{k+1,k}\right)\boldsymbol{q}_k \tag{4.22}$$

4.3.2 Nonlinear Model

Using the body attitude kinematics and the observation model that relates the GNSS observations with the attitude and integer ambiguities, a nonlinear model can be defined. The state vector representing the variables of the system is given by $\boldsymbol{x}_k = \left[\boldsymbol{q}_k^{\top},\ \boldsymbol{b}_k^{\top},\ \boldsymbol{n}_k^{\top}\right]^{\top}$, where $\boldsymbol{q}_k$ is the unit quaternion giving the attitude between the reference and body frame, $\boldsymbol{b}_k$ is the 3-axis gyroscope bias, and $\boldsymbol{n}_k$ is the vector of double differenced integer ambiguities defined as $\boldsymbol{n}_k \equiv vec(\boldsymbol{N})$, for $\boldsymbol{N}$ in (3.12).

The dynamics of the state vector are written in (4.23), according to the quaternion kinematics and discrete-time gyroscope model in (4.22) and (4.18), respectively.

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} \boldsymbol{q}_{k+1} \\ \boldsymbol{b}_{k+1} \\ \boldsymbol{n}_{k+1} \end{bmatrix} = \begin{bmatrix} \bar{\boldsymbol{\Psi}}_{k+1,k}\ \boldsymbol{q}_k \\ \boldsymbol{b}_k + \boldsymbol{w}_{u_{k+1}} \\ \boldsymbol{n}_k \end{bmatrix} = \begin{bmatrix} \bar{\boldsymbol{f}}(\boldsymbol{q}_k, \boldsymbol{b}_k, \boldsymbol{w}_{v_{k+1}}) \\ \boldsymbol{b}_k + \boldsymbol{w}_{u_{k+1}} \\ \boldsymbol{n}_k \end{bmatrix} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{w}_{v_{k+1}}, \boldsymbol{w}_{u_{k+1}}) \tag{4.23}$$

The output vector consists of the GNSS observables described in Chapter 3. Namely, the double differenced pseudorange and carrier phase range measurements, considering the short baseline approximation. The measurements model considered in this system is based on the GNSS-based attitude model (3.18). To relate the measurements with the state vector $\boldsymbol{x}_k$, one can write the rotation of the baseline vectors in the body frame, $\boldsymbol{RF}$, using the quaternion product introduced in (2.7). The resulting observations model (4.24) is nonlinear in relation to the state vector and time variant. Note that the following definitions have been made: the measurements vector $\boldsymbol{z}_k \equiv vec(\boldsymbol{Y})$ and the measurement noise $\boldsymbol{v}_k \equiv \boldsymbol{\Xi}$.

$$\begin{gathered} \boldsymbol{z}_k = \boldsymbol{S}\boldsymbol{n}_k + \boldsymbol{g}(\boldsymbol{q}_k) + \boldsymbol{v}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{v}_k, \\ \boldsymbol{S} = \begin{bmatrix} \boldsymbol{0}_{mn} \\ \lambda\boldsymbol{I}_{mn} \end{bmatrix}, \quad \boldsymbol{g}(\boldsymbol{q_k}) = \begin{bmatrix} \boldsymbol{1}_2 \otimes \begin{bmatrix} \boldsymbol{\Upsilon}\ \boldsymbol{q_k} \circ \boldsymbol{l}_{i1} \circ \boldsymbol{q_k}^* \\ \vdots \\ \boldsymbol{\Upsilon}\ \boldsymbol{q_k} \circ \boldsymbol{l}_{in} \circ \boldsymbol{q_k}^* \end{bmatrix} \end{bmatrix} \end{gathered} \tag{4.24}$$

The complete nominal system model is then written as (4.25), containing both the state dynamics and output models.

$$\begin{cases} \boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{w}_{v_{k+1}}, \boldsymbol{w}_{u_{k+1}}) \\ \boldsymbol{z}_k = \boldsymbol{h}(\boldsymbol{x}_k) + \boldsymbol{v}_k \end{cases} \tag{4.25}$$

4.4 Extended Kalman Filter for Float Estimation

In this Section, the equations of the filter used to compute the float solution of the proposed attitude estimation framework are presented. Note that in this context, the notation $\hat{(\cdot)}$ represents the Extended Kalman Filter estimate and not the float solution as in the previous sections. Furthermore, an observability analysis of the linearized system is also given.

4.4.1 Prediction Step

To compute the *a priori* estimates, the usual EKF expressions apply (4.26) [34], where $\hat{\boldsymbol{x}}_{k|k-1}$ and $\boldsymbol{P}_{k|k-1}$ are the predicted state vector and error covariance matrix, respectively.

$$\begin{aligned} \hat{\boldsymbol{x}}_{k|k-1} &= \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1|k-1}) \\ \boldsymbol{P}_{k|k-1} &= \boldsymbol{F}_k \boldsymbol{P}_{k-1|k-1} \boldsymbol{F}_k^\top + \boldsymbol{Q}_k \end{aligned} \tag{4.26}$$

Having defined the dynamics nominal system, the functions used for the propagation of the estimates can be easily written by taking the expected value of (4.23). The resulting model is given in (4.27), with $\boldsymbol{\Psi}^*(\hat{\boldsymbol{x}}_{k-1|k-1})$ defined in (4.28).

$$\hat{\boldsymbol{x}}_{k|k-1} = \begin{bmatrix} \hat{\boldsymbol{q}}_{k|k-1} \\ \hat{\boldsymbol{b}}_{k|k-1} \\ \hat{\boldsymbol{n}}_{k|k-1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Psi}^*(\hat{\boldsymbol{x}}_{k-1|k-1})\hat{\boldsymbol{q}}_{k-1|k-1} \\ \hat{\boldsymbol{b}}_{k-1|k-1} \\ \hat{\boldsymbol{n}}_{k-1|k-1} \end{bmatrix} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1|k-1}) \tag{4.27}$$

$$\begin{gathered} \boldsymbol{\Psi}^*(\hat{\boldsymbol{x}}_{k-1|k-1}) = \cos\frac{\|\boldsymbol{\delta\vartheta}^*{}_{k,k-1}\|}{2}\boldsymbol{I} + \frac{2}{\|\boldsymbol{\delta\vartheta}^*{}_{k,k-1}\|}\sin\frac{\|\boldsymbol{\delta\vartheta}^*{}_{k,k-1}\|}{2}\boldsymbol{\Theta}^*{}_{k,k-1}, \\ \boldsymbol{\Theta}^*{}_{k,k-1} = \frac{1}{2}\begin{bmatrix} 0 & -[\boldsymbol{\delta\vartheta}^*{}_{k,k-1}]^\top \\ \boldsymbol{\delta\vartheta}^*{}_{k,k-1} & -[\boldsymbol{\delta\vartheta}^*{}_{k,k-1}\times] \end{bmatrix}, \quad \boldsymbol{\delta\vartheta}^*{}_{k,k-1} = \frac{\Delta t}{2}\left(\boldsymbol{\omega}_{m_k} + \boldsymbol{\omega}_{m_{k-1}}\right) - \Delta t\hat{\boldsymbol{b}}_{k-1|k-1} \end{gathered} \tag{4.28}$$

It is plain to see that the state dynamics are not linear in relation to the state variables, since $\bar{\boldsymbol{\Psi}}_{k+1,k}$ has a nonlinear dependence on the gyro bias $\boldsymbol{b}_k$. The nonlinear model as is can be used to compute the predicted state estimate but not the covariance matrix $\boldsymbol{P}_{k|k-1}$. As such, $\bar{\boldsymbol{f}}(\boldsymbol{q}_k, \boldsymbol{b}_k, \boldsymbol{w}_{v_{k+1}})$ must be linearized.

Since the EKF tracks the expected value of the random variables composing the state vector, the white noise random processes $\boldsymbol{w}_{v_{k+1}}$ and $\boldsymbol{w}_{u_{k+1}}$ can be omitted from the equations of the nonlinear model. Making this assumption, the derivatives of the nonlinear function in relation to $\boldsymbol{q}_k$ and $\boldsymbol{b}_k$ are given in (4.29), with matrix $\boldsymbol{L}(\boldsymbol{q}_k, \boldsymbol{b}_k)$ written in (4.30).

$$\frac{\partial \bar{\boldsymbol{f}}(\boldsymbol{q}_k, \boldsymbol{b}_k, \boldsymbol{w}_{v_{k+1}})}{\partial \boldsymbol{q}_k} = \bar{\boldsymbol{\Psi}}_{k+1,k}(\boldsymbol{b}_k), \qquad \frac{\partial \bar{\boldsymbol{f}}(\boldsymbol{q}_k, \boldsymbol{b}_k, \boldsymbol{w}_{v_{k+1}})}{\partial \boldsymbol{b}_k} = \boldsymbol{L}(\boldsymbol{q}_k, \boldsymbol{b}_k) \tag{4.29}$$

$$\begin{aligned} \boldsymbol{L}(\boldsymbol{q}_k, \boldsymbol{b}_k) = {} & \frac{\Delta t \sin \frac{\|\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\|}{2}}{2\|\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\|} \left[\boldsymbol{q}_k \boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}^\top - 2\boldsymbol{\Gamma}(\boldsymbol{q}_k)\right] - \\ & \frac{\Delta t}{2\|\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\|^2} \left[\cos \frac{\|\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\|}{2} - \frac{2}{\|\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\|} \sin \frac{\|\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\|}{2}\right] \boldsymbol{\Gamma}(\boldsymbol{q}_k)\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}\boldsymbol{\delta}\bar{\boldsymbol{\vartheta}}_{k+1,k}^\top, \end{aligned} \tag{4.30}$$

The Jacobian matrix of the state vector dynamics (4.23) is then written as (4.31).

$$\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}_k} = \begin{bmatrix} \bar{\boldsymbol{\Psi}}_{k+1,k}(\boldsymbol{b}_k) & \boldsymbol{L}(\boldsymbol{q}_k, \boldsymbol{b}_k) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \tag{4.31}$$

Then, $\boldsymbol{F}_k$ is simply the Jacobian of the state vector evaluated at the most recent state estimate, as in (4.32). Since the noise processes were omitted, having zero mean, $\bar{\boldsymbol{\Psi}}$ is replaced by $\boldsymbol{\Psi}^*$ as per their respective definitions.

$$\boldsymbol{F}_k = \left.\begin{bmatrix} \boldsymbol{\Psi}^*(\boldsymbol{x}_k) & \boldsymbol{L}(\boldsymbol{x}_k) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}\right|_{\boldsymbol{x}_k = \hat{\boldsymbol{x}}_{k-1|k-1}} \tag{4.32}$$

Due to the quaternion kinematics, It is not simple to rewrite the state vector dynamics (4.23) in the form of additive noise in order to extract $\boldsymbol{Q}_k$, and so an approximate method will be used. According to the continuous-time quaternion kinematics (3.4) and gyroscope model (3.3), the continuous-time state dynamics are given by (4.33).

$$\begin{aligned} \dot{\boldsymbol{x}}(t) = \begin{bmatrix} \dot{\boldsymbol{q}}(t) \\ \dot{\boldsymbol{b}}(t) \\ \dot{\boldsymbol{n}}(t) \end{bmatrix} &= \begin{bmatrix} \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{w}_m(t) - \boldsymbol{b}(t)) & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{q}(t) \\ \boldsymbol{b}(t) \\ \boldsymbol{n}(t) \end{bmatrix} + \boldsymbol{G}(t)\boldsymbol{\eta}(t), \\ \boldsymbol{G}(t) &= \begin{bmatrix} -\frac{1}{2}\boldsymbol{\Gamma}(\boldsymbol{q}(t)) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{\eta}(t) = \begin{bmatrix} \boldsymbol{\eta}_v(t) \\ \boldsymbol{\eta}_w(t) \end{bmatrix} \end{aligned} \tag{4.33}$$

Then, in order to compute $\boldsymbol{Q}_k$, it is assumed that the sampling interval Δt is small enough so that the approximation in (4.34) gives valid results [34], where $\boldsymbol{Q}$ is the covariance matrix relative to the continuous-time noise vector $\boldsymbol{\eta}(t)$.

$$\boldsymbol{Q}_k = \boldsymbol{G}_k \boldsymbol{Q} \boldsymbol{G}_k^\top \Delta t \tag{4.34}$$

4.4.2 Update Step

The expressions in (4.35) are used to compute the *a posteriori* estimates of the state vector $\hat{\boldsymbol{x}}_{k|k}$ and the estimation error covariance matrix $\boldsymbol{P}_{k|k}$.

$$
\begin{aligned}
\boldsymbol{K}_k &= \boldsymbol{H}_k^\top \left(\boldsymbol{H}_k \boldsymbol{P}_{k|k-1} \boldsymbol{H}_k^\top + \boldsymbol{R}_k\right)^{-1} \\
\hat{\boldsymbol{x}}_{k|k} &= \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k \left(\boldsymbol{z}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1})\right) \\
\boldsymbol{P}_{k|k} &= \left(\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k\right) \boldsymbol{P}_{k|k-1} \left(\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{H}_k\right)^\top + \boldsymbol{K}_k \boldsymbol{R}_k \boldsymbol{K}_k^\top
\end{aligned}
\tag{4.35}
$$

The predicted measurements, $\boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1})$, are computed using the measurement model (4.24) of the nominal system equations. Replacing with the estimated state vector, we obtain

$$
\begin{gathered}
\boldsymbol{h}(\hat{\boldsymbol{x}}_{k|k-1}) = \hat{\boldsymbol{z}}_k = \boldsymbol{S}\hat{\boldsymbol{n}}_{k|k-1} + \boldsymbol{g}(\hat{\boldsymbol{q}}_{k|k-1}), \\
\boldsymbol{S} = \begin{bmatrix} \boldsymbol{0}_{mn} \\ \lambda \boldsymbol{I}_{mn} \end{bmatrix}, \quad \boldsymbol{g}(\hat{\boldsymbol{q}}_{k|k-1}) = \begin{bmatrix} \boldsymbol{1}_2 \otimes \begin{bmatrix} \boldsymbol{\Upsilon}\, \hat{\boldsymbol{q}}_{k|k-1} \circ \boldsymbol{l}_{i1} \circ \hat{\boldsymbol{q}}^*_{k|k-1} \\ \vdots \\ \boldsymbol{\Upsilon}\, \hat{\boldsymbol{q}}_{k|k-1} \circ \boldsymbol{l}_{in} \circ \hat{\boldsymbol{q}}^*_{k|k-1} \end{bmatrix} \end{bmatrix}
\end{gathered}
\tag{4.36}
$$

The output equation is nonlinear in $\boldsymbol{q_k}$ due to the quaternion product. In order to write the sensitivity matrix $\boldsymbol{H_k}$, this equation must be linearized. The derivative of the nonlinear term is written in (4.37) [20].

$$
\begin{gathered}
\frac{d}{d\boldsymbol{q}}\boldsymbol{g}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{1}_2 \otimes \begin{bmatrix} \boldsymbol{\Upsilon} \boldsymbol{J_q}(\boldsymbol{q}, \boldsymbol{l}_{i1}) \\ \vdots \\ \boldsymbol{\Upsilon} \boldsymbol{J_q}(\boldsymbol{q}, \boldsymbol{l}_{in}) \end{bmatrix} \end{bmatrix} = \boldsymbol{G'}(\boldsymbol{q}), \\
\boldsymbol{J_q}(\boldsymbol{q}, \boldsymbol{l}) = 2\left[q_w \boldsymbol{l} - [\boldsymbol{l}\times]\boldsymbol{q}_v, -q_w[\boldsymbol{l}\times] + \boldsymbol{q}_v^\top \boldsymbol{l}\boldsymbol{I} + \boldsymbol{q}_v \boldsymbol{l}^\top - \boldsymbol{l}\boldsymbol{q}_v^\top\right], \quad \boldsymbol{J_q}(\boldsymbol{q}, \boldsymbol{l}) \in \mathbb{R}^{3\times 4}
\end{gathered}
\tag{4.37}
$$

Matrix $\boldsymbol{H}_k$ is defined as the Jacobian of the measurements equation (4.24) evaluated at the most recent state estimate. From the expression of $\boldsymbol{h}(\boldsymbol{x}_k)$ and replacing the computed derivative, $\boldsymbol{H}_k$ is given as (4.38).

$$
\boldsymbol{H}_k = \left[\boldsymbol{G'}(\boldsymbol{x}_k) \quad \boldsymbol{0} \quad \boldsymbol{S}\right]\Big|_{\boldsymbol{x}_k = \hat{\boldsymbol{x}}_{k|k-1}}
\tag{4.38}
$$

Lastly, the measurements noise covariance matrix $\boldsymbol{R}_k$ is equivalent to the $\boldsymbol{Q_{YY}}$ matrix defined in (3.13).

The EKF is not optimal and gives reasonable results only if the estimated state around which the model is linearized is near to the true state. How small this difference needs to be depends on the non-linearity of the underlying system [35].

In the next Section, the observability of an approximate perturbation model of the nonlinear system is analyzed.

4.4.3 Observability Analysis for Single vs Multiple Baseline

In this work, the experiments done with the objective of acquiring data to validate the developed algorithm were performed using only two GNSS antennas. If the attitude is estimated using the typical epoch-by-epoch approach detailed in Section 4.1, it is well known that a single baseline is not enough to determine the full attitude of the body. However, the present work proposes an EKF approach that fuses gyroscope data with the GNSS observations. The goal of the following is then to gain a better understanding of the observability of the nonlinear system in the case of using a single baseline model, as well as multiple baselines.

Regarding observability, two separate concepts must be distinguished. First, the observability of a nonlinear system, which is closely related to the *indistinguishability* of points belonging to the space of possible states. Two points $\boldsymbol{x}^0$ and $\boldsymbol{x}^1$ are said to be indistinguishable if they realize the same input-output map. That is, if for every possible system input, the system with initial conditions $\boldsymbol{x}^0$ or $\boldsymbol{x}^1$ produces the same output. For more on this, refer to e.g. [36]. Then, one can also study the observability of the EKF based on the nonlinear system. The relation between observability of the nonlinear system and uniform boundedness of the error covariances of the EKF, as well as the convergence of the error are thoroughly explained in [35].

The observability of a nonlinear system is typically analysed using the *observability rank condition* [36]. It envolves computing the rank of the matrix of Lie derivatives $L^k_{\boldsymbol{f}_j} h_i$ [37], where $\boldsymbol{f}_j$ are the drift and input vectorfields in the dynamics equation of the system, and h_i are the functions of the output map.

For the nonlinear system at hand, determining the rank of the corresponding observability matrix analytically is a complex problem. As such, we will resort to a linearized model of the original system. Assuming that the linearization is done around a point near the true state of the system, the observability of the linearized model can provide some insight on the characteristics of the nonlinear system. In the following, the linearized model will be assumed deterministic and Linear Time-Invariant (LTI), and therefore relative to a local perturbation.

Let $\tilde{\boldsymbol{x}}_k = \left[\tilde{\boldsymbol{q}}_k^\top,\, \tilde{\boldsymbol{b}}_k^\top,\, \tilde{\boldsymbol{n}}_k^\top\right]^\top$ be the state vector corresponding to an operation point of the considered system. Then, the trajectory of the system along this operation point obeys the state vector dynamics written in (4.23). A deviation from the operation point, $\Delta\boldsymbol{x}_{k+1} = \boldsymbol{x}_{k+1} - \tilde{\boldsymbol{x}}_{k+1}$, can be locally approximated by

$$\Delta\boldsymbol{x}_{k+1} \approx \left.\begin{bmatrix} \boldsymbol{\Psi}^*(\boldsymbol{x}_k) & \boldsymbol{L}(\boldsymbol{x}_k) & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}\right|_{\boldsymbol{x}_k = \tilde{\boldsymbol{x}}_k} \Delta\boldsymbol{x}_k \;= \boldsymbol{\Phi}\Delta\boldsymbol{x}_k. \tag{4.39}$$

The perturbation of the measurements can be written in a similar way, which leads to the linearized model (4.40). As before, no noise is considered in the equations.

$$\Delta\boldsymbol{z}_k = \left.\begin{bmatrix} \boldsymbol{G}'(\boldsymbol{x}_k) & \boldsymbol{0} & \boldsymbol{S} \end{bmatrix}\right|_{\boldsymbol{x}_k = \tilde{\boldsymbol{x}}_k} \Delta\boldsymbol{x}_k \;= \boldsymbol{H}\Delta\boldsymbol{x}_k \tag{4.40}$$

A simple criteria for analyzing the observability of an LTI system is given in [34] and consists of computing the rank of the observability matrix $\boldsymbol{\mathcal{O}}$ defined as

$$\boldsymbol{\mathcal{O}} = \begin{bmatrix} \boldsymbol{H} \\ \boldsymbol{H}\boldsymbol{\Phi} \\ \vdots \\ \boldsymbol{H}\boldsymbol{\Phi}^{n-1} \end{bmatrix}, \tag{4.41}$$

where n is the dimension of the system's state vector. If $\boldsymbol{\mathcal{O}}$ is full rank, that is, rank $\boldsymbol{\mathcal{O}} = n$, then the linear system is observable.

First note that simplifying the notation $\boldsymbol{\Psi} \equiv \boldsymbol{\Psi}(\boldsymbol{x}_k)$, $\boldsymbol{L} \equiv \boldsymbol{L}(\boldsymbol{x}_k)$ and given the structure of $\boldsymbol{\Phi}$ in (4.39), the matrix power $\boldsymbol{\Phi}^k$ can be written as

$$\boldsymbol{\Phi}^k = \begin{bmatrix} (\boldsymbol{\Psi}^*)^k & \left((\boldsymbol{\Psi}^*)^{k-1} + (\boldsymbol{\Psi}^*)^{k-2} + ... + \boldsymbol{I}\right)\boldsymbol{L} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}. \tag{4.42}$$

Therefore the product $\boldsymbol{H}\boldsymbol{\Phi}^k$, using the definition in (4.40), is given by

$$\boldsymbol{H}\boldsymbol{\Phi}^k = \begin{bmatrix} \boldsymbol{G}'(\boldsymbol{\Psi}^*)^k & \boldsymbol{G}'\left((\boldsymbol{\Psi}^*)^{k-1} + (\boldsymbol{\Psi}^*)^{k-2} + ... + \boldsymbol{I}\right)\boldsymbol{L} & \boldsymbol{0} \end{bmatrix}, \ \forall k > 0 \tag{4.43}$$

and the observability matrix is written as

$$\boldsymbol{\mathcal{O}} = \begin{bmatrix} \boldsymbol{G}' & \boldsymbol{0} & \boldsymbol{S} \\ \boldsymbol{G}'\boldsymbol{\Psi}^* & \boldsymbol{G}'\boldsymbol{L} & \boldsymbol{0} \\ \boldsymbol{G}'(\boldsymbol{\Psi}^*)^2 & \boldsymbol{G}'(\boldsymbol{\Psi}^* + \boldsymbol{I})\boldsymbol{L} & \boldsymbol{0} \\ \vdots & \vdots & \vdots \\ \boldsymbol{G}'(\boldsymbol{\Psi}^*)^{n-1} & \boldsymbol{G}'\left((\boldsymbol{\Psi}^*)^{n-2} + ... + \boldsymbol{I}\right)\boldsymbol{L} & \boldsymbol{0} \end{bmatrix}. \tag{4.44}$$

Defining the function $f(\boldsymbol{x}) = \boldsymbol{\mathcal{O}}\boldsymbol{x}$, it is known that $f(\boldsymbol{x})$ is injective *iif* $\boldsymbol{\mathcal{O}}$ has full column rank. For the function to be injective, the kernel of $\boldsymbol{\mathcal{O}}$ must have dimension 0. That is, it must be composed of only the null vector. Then, writing the system of equations (4.45), if the only solution is $\boldsymbol{x} = \left[\boldsymbol{x}_1^\top,\ \boldsymbol{x}_2^\top,\ \boldsymbol{x}_3^\top\right]^\top = \boldsymbol{0}$, then $\boldsymbol{\mathcal{O}}$ has full column rank.

$$\boldsymbol{\mathcal{O}}\boldsymbol{x} = \begin{bmatrix} \boldsymbol{G}' & \boldsymbol{0} & \boldsymbol{S} \\ \boldsymbol{G}'\boldsymbol{\Psi}^* & \boldsymbol{G}'\boldsymbol{L} & \boldsymbol{0} \\ \boldsymbol{G}'(\boldsymbol{\Psi}^*)^2 & \boldsymbol{G}'(\boldsymbol{\Psi}^* + \boldsymbol{I})\boldsymbol{L} & \boldsymbol{0} \\ \vdots & \vdots & \vdots \\ \boldsymbol{G}'(\boldsymbol{\Psi}^*)^{n-1} & \boldsymbol{G}'\left((\boldsymbol{\Psi}^*)^{n-2} + ... + \boldsymbol{I}\right)\boldsymbol{L} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \boldsymbol{x}_3 \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \tag{4.45}$$

From this point on, the single and multiple baseline cases can be discriminated.

Single Baseline

Having only one baseline $\boldsymbol{l}_{12}$ tracking $m+1$ satellites, $\boldsymbol{G}'$ is written as

$$\boldsymbol{G}' = \left[\mathbf{1}_2 \otimes \left[\boldsymbol{\Upsilon} \boldsymbol{J}_{\boldsymbol{q}}(\boldsymbol{q}_k, \boldsymbol{l}_{12})\right]\right]\Big|_{\boldsymbol{x}_k=\tilde{\boldsymbol{x}}_k}, \quad \boldsymbol{G}' \in \mathbb{R}^{2m\times 4}. \tag{4.46}$$

Replacing the expressions for $\boldsymbol{G}'$ and $\boldsymbol{S}$, the first line of equations from the system (4.45) reads

$$\boldsymbol{G}'\boldsymbol{x}_1 + \boldsymbol{S}\boldsymbol{x}_3 = \begin{bmatrix} \boldsymbol{\Upsilon} \boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\boldsymbol{x}_1 + \mathbf{0}_m \boldsymbol{x}_3 \\ \boldsymbol{\Upsilon} \boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\boldsymbol{x}_1 + \lambda \boldsymbol{I}_m \boldsymbol{x}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}. \tag{4.47}$$

Naturally, this system corresponds to solving the observations equation of the linearized system (4.40) equal to 0. Defining $\boldsymbol{x}_1 \equiv \Delta\boldsymbol{q}_k$ and $\boldsymbol{x}_3 \equiv \Delta\boldsymbol{n}_k$ for convenience, (4.47) can be rewritten as

$$\Delta\boldsymbol{z}_k = \begin{bmatrix} \boldsymbol{\Upsilon} \boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\Delta\boldsymbol{q}_k + \mathbf{0}_m \Delta\boldsymbol{n}_k \\ \boldsymbol{\Upsilon} \boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\Delta\boldsymbol{q}_k + \lambda \boldsymbol{I}_m \Delta\boldsymbol{n}_k \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}. \tag{4.48}$$

As defined before for $\Delta\boldsymbol{x}_k$, also $\Delta\boldsymbol{q}_k = \boldsymbol{q}_k - \tilde{\boldsymbol{q}}_k$ and $\Delta\boldsymbol{z}_k = \boldsymbol{z}_k - \tilde{\boldsymbol{z}}_k$. Concerning $\boldsymbol{x}_3$, it is clear from (4.47) that the only solution is $\boldsymbol{x}_3 = \mathbf{0}$ since $\lambda \neq 0$. The goal is then to find the admissible perturbations from the operation point $\Delta\boldsymbol{q}_k$ that make the measurements variation null. First, consider the scenario where there is a clear line of sight to at least four non-planar satellites, so that $\boldsymbol{\Upsilon}$ is full rank . Also note that in a local sense, the LOS vectors vary very slowly and can be considered approximately constant. Then, taking into account the physical meaning of the observations, it is simple to realize that if the variation in attitude from $\tilde{\boldsymbol{q}}_k$ to $\boldsymbol{q}_k$ ($\Delta\boldsymbol{q}_k$) corresponds to a rotation around an axis collinear to the baseline $\boldsymbol{l}_{12}$, the coordinates of the baseline do not change, which implies $\Delta\boldsymbol{z}_k = \mathbf{0}$. We can then write the solution of (4.47) as

$$\begin{cases} \boldsymbol{x}_1 = \Delta\boldsymbol{q}_k = \tilde{\boldsymbol{q}}_k \circ \boldsymbol{q}\{\alpha\boldsymbol{l}_{12}\} - \tilde{\boldsymbol{q}}_k, \quad \forall\alpha \in \mathbb{R} \\ \boldsymbol{x}_3 = \Delta\boldsymbol{n}_k = \mathbf{0} \end{cases}, \tag{4.49}$$

where $\boldsymbol{q}\{\alpha\boldsymbol{l}_{12}\}$ denotes the unit quaternion that represents a rotation around the axis defined by $\boldsymbol{l}_{12}$ of magnitude $\alpha\|\boldsymbol{l}_{12}\|$, according to (2.4). An analytical verification of this solution can be found in Appendix A.

To verify if this is a solution of the whole system, the second line of equations in (4.45) must be evaluated. It reads

$$\boldsymbol{G}'\boldsymbol{\Psi}^*\boldsymbol{x}_1 + \boldsymbol{G}'\boldsymbol{L}\boldsymbol{x}_2 = \mathbf{0}. \tag{4.50}$$

We can once again for convenience define $\boldsymbol{x}_1 \equiv \Delta\boldsymbol{q}_k$, corresponding to the solution in (4.49). Then, the product

$$\boldsymbol{\Psi}^*\boldsymbol{x}_1 = \boldsymbol{\Psi}^*\Delta\boldsymbol{q}_k = \boldsymbol{\Psi}^*\left(\tilde{\boldsymbol{q}}_k \circ \boldsymbol{q}\{\alpha\boldsymbol{l}_{12}\} - \tilde{\boldsymbol{q}}_k\right) \tag{4.51}$$

can be recognized as the quaternion kinematics equation applied to both $\tilde{\boldsymbol{q}}_k \circ \boldsymbol{q}\{\alpha\boldsymbol{l}_{12}\}$ and $\tilde{\boldsymbol{q}}_k$. That is, the attitude given by these quaternions is changed according to the angular velocity of the body used to compute $\boldsymbol{\Psi}^*$. Matrix $\boldsymbol{\Psi}^*$ depends on the operation point of the linearization. If we consider that the linearization is done around a trajectory where the body is rotating around $\boldsymbol{l}_{12}$, or is static, then $\Delta \boldsymbol{q}' = \boldsymbol{\Psi}^* \Delta \boldsymbol{q}_k$ gives a quaternion difference $\Delta \boldsymbol{q}'$ that still corresponds to a difference in rotation around $\boldsymbol{l}_{12}$. In this case, $\boldsymbol{G}' \boldsymbol{\Psi}^* \boldsymbol{x}_1 = \mathbf{0}$. The same reasoning can be applied to the remaining lines of the system, since $(\boldsymbol{\Psi}^*)^k \boldsymbol{x}_1$, under these assumptions, are simply sucessive rotations of $\boldsymbol{x}_1$ around the same axis. Then, a solution of (4.50) can be found by simply setting $\boldsymbol{x}_2 = \mathbf{0}$, giving the conclusion of non-observability for the attitude.

Solving for a non-trivial solution $\boldsymbol{x}_2$ is more problematic due to the complex structure of matrix $\boldsymbol{L}$. The particular case where the body is static simplifies the problem and is now discussed. It has been shown that if the body is static, then $\boldsymbol{G}' \boldsymbol{\Psi}^* \boldsymbol{x}_1 = \mathbf{0}$. Replacing in (4.50) and noting the expression for $\boldsymbol{G}'$ considering one baseline, the equation to solve becomes

$$\boldsymbol{\Upsilon} \boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12}) \boldsymbol{L} \boldsymbol{x}_2 = \mathbf{0}. \tag{4.52}$$

The static assumption can be used to simplify the expression for $\boldsymbol{L}$. Given its definition in (4.30), the result is

$$\lim_{\|\boldsymbol{\delta} \bar{\boldsymbol{\vartheta}}_{k+1,k}\| \to 0} \boldsymbol{L}(\tilde{\boldsymbol{q}}_k, \tilde{\boldsymbol{b}}_k) = -\frac{\Delta t}{2} \boldsymbol{\Gamma}(\tilde{\boldsymbol{q}}_k), \tag{4.53}$$

with matrix $\boldsymbol{\Gamma}$ defined in (3.4). The equation to solve becomes

$$-\frac{\Delta t}{2} \boldsymbol{\Upsilon} \boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12}) \boldsymbol{\Gamma}(\tilde{\boldsymbol{q}}_k) \boldsymbol{x}_2 = \mathbf{0}. \tag{4.54}$$

With further manipulation of this expression, (4.54) simplifies to

$$\boldsymbol{\Upsilon} \boldsymbol{R}\left(\tilde{\boldsymbol{q}}_k\right) [\boldsymbol{l}_{12} \times] \boldsymbol{x}_2 = 0, \tag{4.55}$$

where $\boldsymbol{R}\left(\tilde{\boldsymbol{q}}_k\right)$ denotes the rotation matrix parameterized by the quaternion $\tilde{\boldsymbol{q}}_k$ as in (2.5). Since $\boldsymbol{\Upsilon}$ is full rank granted that there are at least four non-planar satellites in sight, the only solution to $\boldsymbol{\Upsilon} \boldsymbol{r} = \mathbf{0}$, for $\boldsymbol{r} \in \mathbb{R}^3$, is $\boldsymbol{r} = \mathbf{0}$. As such, we have

$$\boldsymbol{R}\left(\tilde{\boldsymbol{q}}_k\right) [\boldsymbol{l}_{12} \times] \boldsymbol{x}_2 = 0, \tag{4.56}$$

from which trivially follows the solution

$$\boldsymbol{x}_2 = c_1 \boldsymbol{l}_{12}, \quad \forall c_1 \in \mathbb{R}. \tag{4.57}$$

It is then concluded from the linearized model that, in a single baseline scenario, if the body is rotating around the axis defined by the baseline or is static, the attitude is not observable. Furthermore, the corresponding kernel has one degree of freedom defined by the direction of the baseline. For the particular case where the body is static, the bias is also not observable and the corresponding kernel is of dimension 1.

For the case where the linearization is done around a trajectory where the rotation axis does not coincide with the baseline, the complete equation in (4.50) would need to be solved analytically, which will be left for a future work.

Multiple Baseline

If there are three or more non-collinear antennas placed on the body, the analysis of the linearized model is simpler. While in the following the case of two baselines is described, note that it could easily be expanded to three or more baselines.

For two baselines, $\boldsymbol{l}_{12}$ and $\boldsymbol{l}_{13}$ tracking $m+1$ satellites, the expression for $\boldsymbol{G'}$ is

$$\boldsymbol{G'} = \left.\begin{bmatrix} \boldsymbol{1}_2 \otimes \begin{bmatrix} \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\boldsymbol{q}_k, \boldsymbol{l}_{12}) \\ \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\boldsymbol{q}_k, \boldsymbol{l}_{13}) \end{bmatrix} \end{bmatrix}\right|_{\boldsymbol{x}_k = \tilde{\boldsymbol{x}}_k}, \quad \boldsymbol{G'} \in \mathbb{R}^{4m \times 4}. \tag{4.58}$$

The first line of equations of system (4.45) now reads

$$\boldsymbol{G'}\boldsymbol{x}_1 + \boldsymbol{S}\boldsymbol{x}_3 = \begin{bmatrix} \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_{\boldsymbol{k}}, \boldsymbol{l}_{12})\boldsymbol{x}_1 \\ \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_{\boldsymbol{k}}, \boldsymbol{l}_{13})\boldsymbol{x}_1 \\ \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_{\boldsymbol{k}}, \boldsymbol{l}_{12})\boldsymbol{x}_1 \\ \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_{\boldsymbol{k}}, \boldsymbol{l}_{13})\boldsymbol{x}_1 \end{bmatrix} + \begin{bmatrix} \boldsymbol{0}_{2m}\boldsymbol{x}_3 \\ \lambda \boldsymbol{I}_{2m}\boldsymbol{x}_3 \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix}. \tag{4.59}$$

This system can once again be likened to the observations equation, as done in (4.48), but using two baselines. Rearraging the equations, (4.59) can be written as

$$\begin{bmatrix} \Delta \boldsymbol{z}_k^{12} \\ \Delta \boldsymbol{z}_k^{13} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\Delta \boldsymbol{q}_k + \boldsymbol{0}_m \Delta \boldsymbol{n}_k \\ \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\Delta \boldsymbol{q}_k + \lambda \boldsymbol{I}_m \Delta \boldsymbol{n}_k \\ \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{13})\Delta \boldsymbol{q}_k + \boldsymbol{0}_m \Delta \boldsymbol{n}_k \\ \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{13})\Delta \boldsymbol{q}_k + \lambda \boldsymbol{I}_m \Delta \boldsymbol{n}_k \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix}, \tag{4.60}$$

where $\Delta \boldsymbol{z}_k^{12}$ is the variation in observations referring to baseline $\boldsymbol{l}_{12}$ and $\Delta \boldsymbol{z}_k^{13}$ is relative to $\boldsymbol{l}_{13}$. As before, it follows that $\Delta \boldsymbol{n}_k = \boldsymbol{x}_3 = \boldsymbol{0}$. Now, the same rationale can be applied regarding $\Delta \boldsymbol{q}_k$. If this difference in attitudes corresponds to a rotation around $\boldsymbol{l}_{12}$, then $\Delta \boldsymbol{z}_k^{12} = \boldsymbol{0}$. However, if $\boldsymbol{l}_{12}$ and $\boldsymbol{l}_{13}$ are not collinear, then the same $\Delta \boldsymbol{q}_k$ will entail $\Delta \boldsymbol{z}_k^{13} \neq \boldsymbol{0}$, since the coordinates of the baseline $\boldsymbol{l}_{13}$ in the reference frame will change. The analysis is similar if $\Delta \boldsymbol{q}_k$ is instead defined as a rotation around $\boldsymbol{l}_{13}$. Therefore, for non-collinear baselines, the only solution is $\Delta \boldsymbol{q}_k = \boldsymbol{x}_1 = \boldsymbol{0}$. It can be immediately concluded that, without imposing any restrictions on the trajectory of the linearization, the attitude is observable.

To solve for $\boldsymbol{x}_2$, the second line of equations of (4.45) is analysed. Since $\boldsymbol{x}_1 = \boldsymbol{0}$, it reduces to

$$\begin{bmatrix} \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\boldsymbol{L}\boldsymbol{x}_2 \\ \boldsymbol{\Upsilon}\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{13})\boldsymbol{L}\boldsymbol{x}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix}. \tag{4.61}$$

The analysis will again be simplified to the particular case where the body is static. Replacing the expression for $\boldsymbol{L}$ in (4.53) and making the same simplifications as before, the system to solve becomes

$$\begin{bmatrix} \boldsymbol{\Upsilon} \boldsymbol{R}\left(\tilde{\boldsymbol{q}}_k\right)[\boldsymbol{l}_{12}\times]\boldsymbol{x}_2 = 0 \\ \boldsymbol{\Upsilon} \boldsymbol{R}\left(\tilde{\boldsymbol{q}}_k\right)[\boldsymbol{l}_{13}\times]\boldsymbol{x}_2 = 0 \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix}. \tag{4.62}$$

Assuming that $\boldsymbol{\Upsilon}$ is full rank, the solution consists of finding $\boldsymbol{x}_2$ such that, simultaneously, $[\boldsymbol{l}_{12}\times]\boldsymbol{x}_2 = 0$ and $[\boldsymbol{l}_{13}\times]\boldsymbol{x}_2 = 0$. If $\boldsymbol{l}_{12}$ and $\boldsymbol{l}_{13}$ are not collinear, the only solution is naturally $\boldsymbol{x}_2 = \boldsymbol{0}$.

The conclusions from analysing the observability of the linearized system in a multiple baseline scenario are that the attitude is always observable. Furthermore, in the particular case where the body is static, it is seen that the bias is also observable. While the general case is not solved analytically, intuitively it is to be expected that since the bias is observable when static, it will also be observable in a dynamic scenario as the measurements of the system provide more information.

4.5 Integer Ambiguity Determination

The goal of the integer ambiguity determination algorithm is to solve problem (4.10). By finding the exact integer ambiguities, the accuracy of the attitude solution is greatly enhanced, as shown in [8], where the least-squares approach is used for determining the float solution. The EKF used in the present work is expected to be more precise compared to the float solution computed with the least-squares approach, since it uses the gyroscope measurements and is a recursive method. However, determining the exact set of ambiguities still enhances the accuracy of the solution, as will be shown later on.

The method used to solve (4.10) is called the MC-LAMBDA method. It is built upon the simpler LAMBDA method [12] and so this method is briefly introduced first.

4.5.1 LAMBDA

The Least-squares AMBiguity Decorrelation Adjustment (LAMBDA) method is used to solve the minimization problem defined in (4.63) in an efficient way, where $\hat{\boldsymbol{N}}$ is the float solution of DD ambiguities.

$$\check{\boldsymbol{N}}^{\mathcal{U}} = \underset{\boldsymbol{N} \in \mathbb{Z}^{m \times n}}{\arg\min} \; \|vec(\hat{\boldsymbol{N}} - \boldsymbol{N})\|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}} \tag{4.63}$$

This is equivalent to solving the more complex problem (4.10) without the orthonormality constraint on $\boldsymbol{R}$. Being an ILS problem, an ambiguity search space can be defined as

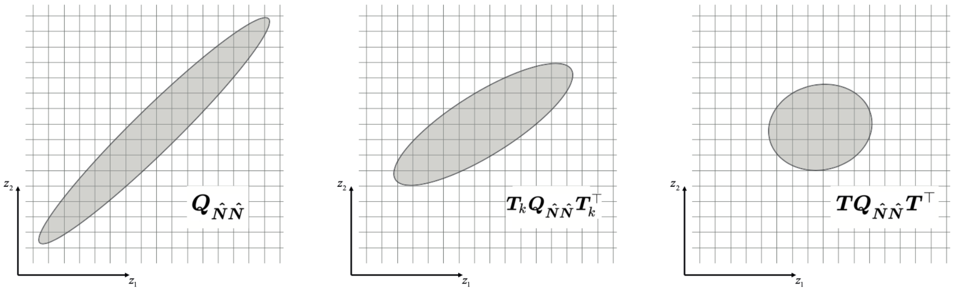
$$\Omega^{\mathcal{U}}(\chi^2) = \{\boldsymbol{N} \in \mathbb{Z}^{m \times n} \mid \|vec(\hat{\boldsymbol{N}} - \boldsymbol{N})\|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}} \leq \chi^2\}. \tag{4.64}$$

Solving (4.63) consists of finding the minimizer of the cost function in the ambiguity search space. As such, the choice of χ^2 is critical since it sets the size of the search space. Too many candidates $\boldsymbol{N}$ hinder the search, while it must be guaranteed that the search space contains at least one matrix of integer ambiguities. Is has been shown that computing χ^2 by rounding the nearest solution is a valid option [12]. That is, $\chi^2 = \|vec(\hat{\boldsymbol{N}} - [\hat{\boldsymbol{N}}])\|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}}$, where the notation $[(\cdot)]$ denotes the rounding of each entry to its nearest integer.

The shape of the search space is governed by the covariance matrix $\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}$. Since the float ambiguities are typically heavily correlated [28], the search space, geometrically an hyperellipsoid, is highly elongated. This fact causes halting during the search process described later on, which decreases its efficiency. Therefore, the first step of the LAMBDA method consists of the decorrelation of the search space. A transformation $\boldsymbol{T}$ is applied to the ambiguities with the goal of decorrelating matrix $\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}$. Since the transformation should keep the integerness of the ambiguities, the decorrelation is not perfect as both $\boldsymbol{T}$ and $\boldsymbol{T}^{-1}$ must have integer entries. The transformed ambiguities are defined as

$$\boldsymbol{Z} = \boldsymbol{T}\boldsymbol{N}, \quad \hat{\boldsymbol{Z}} = \boldsymbol{T}\hat{\boldsymbol{N}}, \quad \boldsymbol{Q}_{\hat{\boldsymbol{Z}}\hat{\boldsymbol{Z}}} = \boldsymbol{T}\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}. \tag{4.65}$$

Computing the integer transformation $\boldsymbol{T}$ is done using iterative transformations $\boldsymbol{T}_i$ until the covariance matrix cannot be further decorrelated. The process is described in [28], and the overall transformation is then calculated as $\boldsymbol{T} = \prod_i \boldsymbol{T}_i$. A representation of the search spaces in a 2D case during the iterative process can be seen in Figure 4.2. As shown, the search space becomes rounder as the covariance matrix is decorrelated.



(a) Original ellipse, defined by $\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}$.

(b) Intermediate decorrelated ellipse, defined by $\boldsymbol{T}_k\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}\boldsymbol{T}_k^\top$

(c) Final decorrelated ellipse, defined by $\boldsymbol{T}\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}\boldsymbol{T}^\top$

Figure 4.2: Representation of example initial, intermediate and decorrelated search space defined by the covariance matrix of the DD ambiguities.[8]

If the covariance matrix were diagonal, the search process would be very simple as the solution of the minimization in (4.63) would be obtained by simply rounding each entry of the float solution to its nearest integer. This entails that some kind of diagonalization can be done to improve the efficiency of the search. Using a $\boldsymbol{LDL}^\top$ diagonalization of $\boldsymbol{Q}_{\hat{\boldsymbol{Z}}\hat{\boldsymbol{Z}}}$, the quadratic form in (4.63) using the transformed ambiguities can be written as the sum of squares:

$$\|vec(\hat{\boldsymbol{Z}} - \boldsymbol{Z})\|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}} = \|vec(\hat{\boldsymbol{Z}} - \boldsymbol{Z})\|^2_{\boldsymbol{LDL}^\top} = \sum_{i=1}^{mn} \frac{\left(\hat{z}_{i|I} - z_i\right)^2}{\sigma^2_{\hat{z}_{i|I}}} \leq \chi^2, \tag{4.66}$$

where the scalars $\hat{z}_{i|I},\ i = 1,..,mn$ are the conditional least-squares float ambiguities [28]. These correspond to the least-squares estimates of ambiguity z_i, given the fixing of the previous ambiguities up until $i-1$. As such, the notation $\hat{z}_{i|I}$ stands for $\hat{z}_{i|1,\ldots,(i-1)}$. Their corresponding conditional variances

$\sigma^2_{\hat{z}_{i|I}}$ coincide with the diagonal elements of matrix $\boldsymbol{D}$ of the diagonalization.

The sum of squares structure in (4.66) allows writing nm bounds for the individual ambiguities z_i (4.67), so that they can be determined (searched for) in a sequential way. The consequence of the decorrelation step in (4.65) is that this depth-first search is halted as few times as possible. For details on the implementation of the decorrelation, diagonalization and search, see e.g. [28, 38].

$$\begin{cases} (\hat{z}_1 - z_1)^2 \leq \sigma^2_{\hat{z}_1}\chi^2 \\ \left(\hat{z}_{2|1} - z_2\right)^2 \leq \sigma^2_{\hat{z}_{2|1}} \left[\chi^2 - (\hat{z}_1 - z_1)^2 / \sigma^2_{\hat{z}_1}\right] \\ \quad \vdots \\ \left(\hat{z}_{mn|MN} - z_{mn}\right)^2 \leq \sigma^2_{\hat{z}_{mn|MN}} \left[\chi^2 - \sum_{j=1}^{mn-1} \left(\hat{z}_{j|J} - z_j\right)^2 / \sigma^2_{\hat{z}_{j|J}}\right] \end{cases} \tag{4.67}$$

The search process was primarily developed to find the minimizer of the search space, $\check{\boldsymbol{N}}^{\mathcal{U}}$, in an efficient way. However, it can also obviously be used to enumerate all the integer matrices belonging to $\Omega^{\mathcal{U}}$. Lastly, the solution in terms of transformed ambiguities $\boldsymbol{Z}$ is brought back to the original space using the inverse transformation $\boldsymbol{N} = \boldsymbol{T}^{-1}\boldsymbol{Z}$.

4.5.2 MC-LAMBDA

In this case, the minimization problem to be solved is the one described in (4.10). The corresponding cost function $C(\boldsymbol{N})$ is defined as

$$C(\boldsymbol{N}) = \|vec(\hat{\boldsymbol{N}} - \boldsymbol{N})\|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}} + \|vec(\hat{\boldsymbol{R}}(\boldsymbol{N}) - \check{\boldsymbol{R}}(\boldsymbol{N}))\|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}}, \tag{4.68}$$

with $\check{\boldsymbol{R}}(\boldsymbol{N})$ the solution of the nonlinear minimization problem in (4.11). Being a minimization problem with integer constraints, the solution must again be found by performing a search on the ambiguities' search space defined in (4.69).

$$\Omega^{\mathcal{C}}(\chi^2) = \{\boldsymbol{N} \in \mathbb{Z}^{m\times n} \mid C(\boldsymbol{N}) \leq \chi^2\} \tag{4.69}$$

The volume of this search space is more sensible to the choice of χ^2 when compared to $\Omega^{\mathcal{U}}$, the search space of the LAMBDA problem, since $\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}$ is characterized by entries smaller in magnitude than those of $\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}$. For the same candidate $\boldsymbol{N}$, the cost function of the MC-LAMBDA problem takes larger values. Therefore, choosing a χ^2 which guarantees that $\Omega^{\mathcal{C}}$ is not empty is not simple, and might produce a large number of candidates to evaluate. This presents a problem, computationally speaking, since evaluating the cost function $C(\boldsymbol{N})$ a large number of times makes the algorithm unsuitable for real-time applications, mostly due to the computation of $\check{\boldsymbol{R}}(\boldsymbol{N})$ which, as shown later, is done iteratively. A solution to this problem is given in [8] and consists of introducing simpler bounding functions:

$$C_1(\boldsymbol{N}) \leq C(\boldsymbol{N}) \leq C_2(\boldsymbol{N}) \tag{4.70}$$

These functions should present a compromise between being as tight as possible to $C(\boldsymbol{N})$, while proving to be easier to evaluate so as to decrease the computational cost. Several options for these bounds are presented in the literature, and the one used in this work is based on the eigenvalues of $\boldsymbol{Q}^{-1}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}$. This bound makes use of the relations $\lambda_m\|(\cdot)\|^2_{\boldsymbol{I}} \leq \|(\cdot)\|^2_{\boldsymbol{Q}} \leq \lambda_M\|(\cdot)\|^2_{\boldsymbol{I}}$, with λ_m and λ_M being the smallest and largest eigenvalues of $\boldsymbol{Q}^{-1}$, respectively. After some manipulation, the first inequality can be used to write bound $C_1(\boldsymbol{N})$ as

$$C_1(\boldsymbol{N}) = \|vec(\hat{\boldsymbol{N}} - \boldsymbol{N})\|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}} + \lambda_m \sum_{i=1}^{3}(\|\hat{\boldsymbol{r}}_i(\boldsymbol{N})\| - 1)^2, \tag{4.71}$$

where $\hat{\boldsymbol{r}}_i(\boldsymbol{N})$ are the column vectors of $\hat{\boldsymbol{R}}(\boldsymbol{N})$. Along with this bound, the respective search space Ω_1 is defined in (4.72), similarly to $\Omega^{\mathcal{C}}$.

$$\Omega_1(\chi^2) = \{\boldsymbol{N} \in \mathbb{Z}^{m\times n} \mid C_1(\boldsymbol{N} \leq \chi^2\} \tag{4.72}$$

Note the relation between the ambiguities search spaces defined so far. It is easy to conclude that, for a given χ^2, $\Omega_1(\chi^2)$ is contained in $\Omega^{(U}(\chi^2)$, that is, $\Omega^{\mathcal{U}}(\chi^2) \supseteq \Omega_1(\chi^2)$. In turn, due to the inequalities in (4.70), the search space $\Omega^{\mathcal{C}}(\chi^2)$ is contained in $\Omega_1(\chi^2)$.

Two efficient search schemes are proposed [8] , taking advantage of the bounding functions: the *Expansion* approach, using the lower bound $C_1(\boldsymbol{N})$, and the *Search and Shrink* approach making use of $C_2(\boldsymbol{N})$. In this work, the Expansion approach is used.

First, the cost χ^2 is set, as in the LAMBDA case, for example by rounding the float solution and computing $\chi^2 = \|vec(\hat{\boldsymbol{N}} - [\hat{\boldsymbol{N}}])\|^2_{\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}}$. Then, all integer matrices in $\Omega_1(\chi^2)$ are enumerated. Since $\Omega_1(\chi^2)$ is contained in $\Omega^{\mathcal{U}}(\chi^2)$, this is done by first enumerating all $\boldsymbol{N}$ in $\Omega^{\mathcal{U}}(\chi^2)$ and then, for each matrix, evaluating $C_1(\boldsymbol{N})$. If $\Omega_1(\chi^2)$ turns out to be non-empty, then $C(\boldsymbol{N})$ is evaluated for each candidate in $\Omega_1(\chi^2)$ in order to enumerate $\Omega^{\mathcal{C}}(\chi^2)$. If this search space is non-empty, then $\check{\boldsymbol{N}}$ is immediately found as the minimizer of $C(\boldsymbol{N})$ for the candidates in $\Omega^{\mathcal{C}}(\chi^2)$. However, if initially it is found that $\Omega_1(\chi^2)$ is empty, or if after enumerating the $\boldsymbol{N}$ in $\Omega_1(\chi^2)$ and evaluating $C(\boldsymbol{N})$, $\Omega^{\mathcal{C}}(\chi^2)$ turns out empty, χ^2 is increased and the whole process is repeated iteratively until $\check{\boldsymbol{N}}$ is found. The increase in χ^2 is critical: a small increment might imply many iterations before a non-empty set is found, while large increments might lead to unnecessarily big search spaces that take too long to evaluate. Experimental trials using different increments suggest that doubling χ^2 in each iteration is a good option [29], and so this is the increment strategy chosen in this work.

The efficiency of the Expansion approach depends on the tightness of the bounds, as mentioned before. In this particular case, if the lowest eigenvalue of $\boldsymbol{Q}^{-1}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}$ is many orders of magnitude lower than the other eigenvalues, then $C_1(\boldsymbol{N})$ is not close enough to $C(\boldsymbol{N})$. This implies that for a value χ^2 that makes $\Omega^{\mathcal{C}}(\chi^2)$ non-empty, the corresponding search space $\Omega_1(\chi^2)$ will include a high number of candidates. Since each candidate of $\Omega_1(\chi^2)$ must be evaluated before arriving at $\Omega^{\mathcal{C}}(\chi^2)$, the search becomes less efficient.

An overview of the implementation and reasoning behind the MC-LAMBDA algorithm has been given.

However, some issues remain to be tackled. The expressions for the conditional solution $\hat{\boldsymbol{R}}(\boldsymbol{N})$ and its respective covariance matrix $\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}$ were written in Section 4.1, where the float solution is obtained using the least-squares estimates. However, in the present work, the float solutions are computed using an EKF in which the attitude is parameterized using the quaternion. Therefore, new expressions for $\hat{\boldsymbol{R}}(\boldsymbol{N})$ and $\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}$ must be derived in order to use the MC-LAMBDA algorithm. The second issue is finding the solution $\breve{\boldsymbol{R}}(\boldsymbol{N}))$ of the minimization problem written in (4.11). These problems are addressed in the following.

Obtaining the conditional solution

The state vector estimated by the EKF previously described includes the attitude quaternion and the DD ambiguities. Since no constraints are imposed on the normality of the quaternion or the integerness of the DD ambiguities, these estimates correspond to float solutions. In other words, the EKF computes the float solution $\hat{\boldsymbol{q}}$ and $\hat{\boldsymbol{N}}$ as part of its updated state vector. Also, from the EKF's updated covariance matrix, matrices $\boldsymbol{Q}_{\hat{\boldsymbol{q}}\hat{\boldsymbol{q}}}$, $\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}$ and $\boldsymbol{Q}_{\hat{\boldsymbol{q}}\hat{\boldsymbol{N}}}$ can be extracted.

There are several approximations that can be done in order to compute the conditional solution $\hat{\boldsymbol{R}}(\boldsymbol{N})$ in the present case, where the float solution is obtained using an EKF instead of a least-squares solution.

Firstly, the expressions of the least-squares case (4.8) and (4.9) can be used by transforming the float solution $\hat{\boldsymbol{q}}$ into $\hat{\boldsymbol{R}}$. Furthermore, using the nonlinear functions that define the entries of $\boldsymbol{R}(\boldsymbol{q})$ in (2.5), the covariance matrices $\boldsymbol{Q}_{\hat{\boldsymbol{q}}\hat{\boldsymbol{q}}}$ and $\boldsymbol{Q}_{\hat{\boldsymbol{q}}\hat{\boldsymbol{N}}}$ can be used to write $\boldsymbol{Q}_{\hat{\boldsymbol{R}}\hat{\boldsymbol{R}}}$ and $\boldsymbol{Q}_{\hat{\boldsymbol{R}}\hat{\boldsymbol{N}}}$. Unfortunately, in practice it was seen that this would result in a covariance matrix $\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}$ which is not positive definite. The high precision of the EKF ambiguity estimates causes the product $\boldsymbol{Q}_{\hat{\boldsymbol{R}}\hat{\boldsymbol{N}}}\boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}^{-1}$ to have very large entries, making this approach unfeasible.

Another option is to compute the conditional solution $\hat{\boldsymbol{R}}(\boldsymbol{N})$ by solving its minimization problem (4.4) using the usual WLS equations. The downside of this approach is that it only works when using two or more non-collinear baselines, as only then it is possible to estimate the full attitude of the body in an epoch-by-epoch basis. Using the EKF, it might be possible to estimate the full attitude under certain conditions using a single baseline. To take advantage of this, the conditional solution must necessarily be computed using the float solution, described in the following.

To describe how the EKF estimates lead to $\hat{\boldsymbol{R}}(\boldsymbol{N})$ and $\boldsymbol{Q}_{\hat{\boldsymbol{R}}(\boldsymbol{N})\hat{\boldsymbol{R}}(\boldsymbol{N})}$, consider first the nonlinear least-squares problem (4.73). For simplicity of notation, the definitions $\boldsymbol{y} \equiv vec(\boldsymbol{Y})$ and $\boldsymbol{n} \equiv vec(\boldsymbol{N})$ will be applied in this context.

$$\langle \hat{\boldsymbol{n}}, \hat{\boldsymbol{q}} \rangle = \underset{\boldsymbol{n}\in\mathbb{R}^{mn}, \boldsymbol{q}\in\mathbb{R}^4}{\arg\min} \|\boldsymbol{y} - \boldsymbol{S}\boldsymbol{n} - \boldsymbol{g}(\boldsymbol{q})\|^2_{\boldsymbol{Q}_{\boldsymbol{YY}}} \tag{4.73}$$

This equation corresponds to the minimization problem using the GNSS-based attitude model (4.1), where the rotation of the baselines in the body frame is written using the quaternion product instead of the rotation matrix, as in the nonlinear observation model (4.24). In other words, (4.73) defines a float solution where the attitude is parameterized using the quaternion. Note that the shape of $\boldsymbol{S}$ has been modified to fit with vector $\boldsymbol{n}$.

For a general nonlinear model defined as $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\beta})$, where $\boldsymbol{x}$ and $\boldsymbol{y}$ are input and output data, respectively, and $\boldsymbol{\beta} \in \mathbb{R}^p$ are unknowns to be estimated, the minimization problem

$$\hat{\boldsymbol{\beta}} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \|\boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\beta})\|^2_{\boldsymbol{Q}} \tag{4.74}$$

can be solved using, for example, the Gauss-Newton method. The Gauss-Newton is a method used to solve nonlinear least-squares problems that iteratively finds an estimate of the unknown parameters, minimizes the squared norm error. After providing an initial estimate $\boldsymbol{\beta}^0$, each step is defined by

$$\left(\boldsymbol{J}^\top \boldsymbol{Q}^{-1} \boldsymbol{J}\right) \Delta\boldsymbol{\beta} = \boldsymbol{J}^\top \boldsymbol{Q}^{-1} \Delta\boldsymbol{y}, \tag{4.75}$$

where $\Delta\boldsymbol{\beta} = \boldsymbol{\beta}^{s+1} - \boldsymbol{\beta}^s$, $\boldsymbol{J}$ is the Jacobian matrix of $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\beta})$ in relation to $\boldsymbol{\beta}$ and $\Delta\boldsymbol{y} = \boldsymbol{y} - \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\beta}^s)$.

Concerning the nonlinear least-squares problem in (4.73), the unknown parameters are $\boldsymbol{\beta} = \left[\boldsymbol{q}^\top, \boldsymbol{n}^\top\right]^\top$. As such, let $\boldsymbol{J}_{\boldsymbol{q}^s, \boldsymbol{n}^s} = \begin{bmatrix}\boldsymbol{G}' & \boldsymbol{S}\end{bmatrix}$ denote the Jacobian matrix of the nonlinear observation model implicit in (4.73), evaluated for the estimates at iteration s, $\boldsymbol{q}^s$ and $\boldsymbol{n}^s$. Then, considering only the first iteration of the Gauss-Newton method applied to this problem, using (4.75) we have

$$\begin{bmatrix}\boldsymbol{G}'^\top \\ \boldsymbol{S}^\top\end{bmatrix} \boldsymbol{Q}^{-1}_{\boldsymbol{YY}} \begin{bmatrix}\boldsymbol{G}' & \boldsymbol{S}\end{bmatrix} \begin{bmatrix}\Delta\boldsymbol{q}^1 \\ \Delta\boldsymbol{n}^1\end{bmatrix} = \begin{bmatrix}\boldsymbol{G}'^\top \\ \boldsymbol{S}^\top\end{bmatrix} \boldsymbol{Q}^{-1}_{\boldsymbol{YY}} \Big(\boldsymbol{y} - \left(\boldsymbol{S}\boldsymbol{n}^0 - \boldsymbol{g}(\boldsymbol{q}^0)\right)\Big), \tag{4.76}$$

where $\Delta\boldsymbol{q}^1 = \boldsymbol{q}^1 - \boldsymbol{q}^0$, with $\boldsymbol{q}^0$ being the initial guess for the quaternion. Vector $\Delta\boldsymbol{n}^1$ is defined in similar fashion. While this problem is not a linear least-squares, if the initial guess is considered deterministic and close to the optimal solution such that the linearization is a good local approximation of the nonlinear system, it will be assumed that one may define a "pseudo" covariance matrix relative to the estimates of the first iteration.This matrix is defined in (4.77).

$$\begin{bmatrix}\tilde{\boldsymbol{Q}}_{\boldsymbol{q}^1\boldsymbol{q}^1} & \tilde{\boldsymbol{Q}}_{\boldsymbol{q}^1\boldsymbol{n}^1} \\ \tilde{\boldsymbol{Q}}_{\boldsymbol{n}^1\boldsymbol{q}^1} & \tilde{\boldsymbol{Q}}_{\boldsymbol{n}^1\boldsymbol{n}^1}\end{bmatrix} = \left(\begin{bmatrix}\boldsymbol{G}'^\top \\ \boldsymbol{S}^\top\end{bmatrix} \boldsymbol{Q}^{-1}_{\boldsymbol{YY}} \begin{bmatrix}\boldsymbol{G}' & \boldsymbol{S}\end{bmatrix}\right)^{-1} \tag{4.77}$$

In this way, the estimates $\boldsymbol{q}^1$ and $\boldsymbol{n}^1$ are considered the float solutions using a nonlinear least-squares formulation, along with their respective "pseudo" covariance matrices which are assumed to be an acceptable approximation.

If the integer ambiguities $\boldsymbol{n}$ are known, then a conditional solution for the quaternion can be defined similarly to the conditional rotation matrix $\hat{\boldsymbol{R}}(\boldsymbol{N})$ in the least-squares approach detailed in Section 4.1. The conditional quaternion $\hat{\boldsymbol{q}}(\boldsymbol{n})$ is defined in (4.78).

$$\hat{\boldsymbol{q}}(\boldsymbol{n}) = \arg \min_{\boldsymbol{q} \in \mathbb{R}^4} \|\boldsymbol{y} - \boldsymbol{S}\boldsymbol{n} - \boldsymbol{g}(\boldsymbol{q})\|^2_{\boldsymbol{Q}_{\boldsymbol{YY}}} . \tag{4.78}$$

Keeping in mind that the goal of this derivation is to compute the conditional solution using the float solutions, the nonlinear least-squares problem (4.78) is solved like the previous least-squares, using the first iteration of the Gauss-Newton method, which in this case gives

$$\boldsymbol{G'}^\top \boldsymbol{Q}_{\boldsymbol{YY}}^{-1} \boldsymbol{G'} \Delta \boldsymbol{q}(\boldsymbol{n})^1 = \boldsymbol{G'}^\top \boldsymbol{Q}_{\boldsymbol{YY}}^{-1} \Big(\boldsymbol{y} - \big(\boldsymbol{S}\boldsymbol{n} - \boldsymbol{g}(\boldsymbol{q}(\boldsymbol{n})^0) \big) \Big), \tag{4.79}$$

where as before $\Delta \boldsymbol{q}(\boldsymbol{n})^1 = \boldsymbol{q}(\boldsymbol{n})^1 - \boldsymbol{q}(\boldsymbol{n})^0$, with $\boldsymbol{q}(\boldsymbol{n})^0$ being the initial guess of the conditional solution.

To link the float and conditional solutions of these nonlinear least-square problems, it is assumed that the first guesses used to solve both problems, namely $\boldsymbol{q}^0$ and $\boldsymbol{q}(\boldsymbol{n})^0$, are the same. Thus, also $\boldsymbol{g}(\boldsymbol{q}^0) = \boldsymbol{g}(\boldsymbol{q}(\boldsymbol{n})^0)$. From the first iteration of the Gauss-Newton algorithm in (4.76), one can write

$$\boldsymbol{G'}^\top \boldsymbol{Q}_{\boldsymbol{YY}}^{-1} \boldsymbol{y} = \boldsymbol{G'}^\top \boldsymbol{Q}_{\boldsymbol{YY}}^{-1} \begin{bmatrix} \boldsymbol{G'} & \boldsymbol{S} \end{bmatrix} \begin{bmatrix} \Delta \boldsymbol{q}^1 \\ \Delta \boldsymbol{n}^1 \end{bmatrix} + \boldsymbol{G'}^\top \boldsymbol{Q}_{\boldsymbol{YY}}^{-1} \left(\boldsymbol{S}\boldsymbol{n}^0 - \boldsymbol{g}(\boldsymbol{q}^0) \right). \tag{4.80}$$

Replacing expression (4.80) in (4.79) and simplifying for $\boldsymbol{q}(\boldsymbol{n})^1$, this conditional quaternion is written as in (4.81).

$$\boldsymbol{q}(\boldsymbol{n})^1 = \boldsymbol{q}^1 + \boldsymbol{G'}\boldsymbol{S} \left(\boldsymbol{n}^1 - \boldsymbol{n} \right) \tag{4.81}$$

As such, the result of the first iteration of the conditional quaternion problem is written as a function of the float solution, where only the first iteration was considered as well.

Using the expressions for the inverse of a matrix partitioned into four blocks, as are the matrices in (4.77), it is simple to arrive at the result $\boldsymbol{G'}\boldsymbol{S} = -\tilde{\boldsymbol{Q}}_{\boldsymbol{q}^1\boldsymbol{n}^1} \tilde{\boldsymbol{Q}}_{\boldsymbol{n}^1\boldsymbol{n}^1}^{-1}$. If the float solution is computed instead using an EKF formulation, as is the case in the present work, then the float solutions $\boldsymbol{q}^1$ and $\boldsymbol{n}^1$ can be replaced by $\hat{\boldsymbol{q}}$ and $vec(\hat{\boldsymbol{N}})$, respectively, since the Kalman Filter is a least-squares estimator that is updated recursively. Furthermore, the "pseudo" covariance matrices are replaced by the covariance matrices of the EKF estimates. Then, rewriting $\boldsymbol{n} = vec(\boldsymbol{N})$, the expression used to compute the conditional solution $\boldsymbol{q}(\boldsymbol{N})$ is given in (4.82). Its covariance matrix $\boldsymbol{Q}_{\boldsymbol{q}(\boldsymbol{N})\boldsymbol{q}(\boldsymbol{N})}$ is calculated as in (4.83), using the variance propagation law.

$$\boldsymbol{q}(\boldsymbol{N}) = \hat{\boldsymbol{q}} - \boldsymbol{Q}_{\hat{\boldsymbol{q}}\hat{\boldsymbol{N}}} \boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}^{-1} \Big(vec(\hat{\boldsymbol{N}} - \boldsymbol{N}) \Big) \tag{4.82}$$

$$\boldsymbol{Q}_{\boldsymbol{q}(\boldsymbol{N})\boldsymbol{q}(\boldsymbol{N})} = \boldsymbol{Q}_{\hat{\boldsymbol{q}}\hat{\boldsymbol{q}}} - \boldsymbol{Q}_{\hat{\boldsymbol{q}}\hat{\boldsymbol{N}}} \boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{N}}}^{-1} \boldsymbol{Q}_{\hat{\boldsymbol{N}}\hat{\boldsymbol{q}}} \tag{4.83}$$

These expressions are obtained from gross approximations that were deemed necessary in order to be able to link the float and conditional solutions. However, it was verified experimentally that good results are achieved, as shown later. Then, $\boldsymbol{R}(\boldsymbol{N})$ and its respective covariance matrix is computed from $\boldsymbol{q}(\boldsymbol{N})$ and $\boldsymbol{Q}_{\boldsymbol{q}(\boldsymbol{N})\boldsymbol{q}(\boldsymbol{N})}$ using the quaternion parameterization of the rotation matrix (2.5). The derivation of $\boldsymbol{Q}_{\boldsymbol{q}(\boldsymbol{N})\boldsymbol{q}(\boldsymbol{N})}$ is included in Appendix B.1.

Solving the nonlinear least-squares

Evaluation of the cost function $C(\boldsymbol{N})$ in (4.68) for a given candidate matrix of integer ambiguities $\boldsymbol{N}$ requires computing $\breve{\boldsymbol{R}}(\boldsymbol{N})$. The problem solved in order to find this orthonormal matrix is of the type

$$\breve{\boldsymbol{R}} = \underset{\boldsymbol{R} \in SO(3)}{\arg\min} \ \| vec(\hat{\boldsymbol{R}} - \boldsymbol{R}) \|_{\boldsymbol{Q}}^2. \tag{4.84}$$

As mentioned before, it is often the case that the covariance matrix $\boldsymbol{Q}$ is not diagonal. Then, this problem must be solved using nonlinear estimation techniques. Several different methods are presented and compared in [29]. Different attitude parametrizations of the rotation matrix can be used, resulting in higher or lower computational loads according to the parameters that need to be estimated. An Euler angle parameterization implies that only three parameters need to be estimated, but it involves the evaluation of trigonometric functions. Writing the rotation matrix as a function of the quaternion is shown to be a good option and is the one used in this work.

The constrained minimization is then written as

$$\check{\boldsymbol{q}} = \arg \min_{\boldsymbol{q}\in\mathbb{R}^4,\ \|\boldsymbol{q}\|=1} \|vec(\hat{\boldsymbol{R}} - \boldsymbol{R}(\boldsymbol{q}))\|^2_{\boldsymbol{Q}}, \tag{4.85}$$

where the normality constraint on $\boldsymbol{q}$ guarantees that $\boldsymbol{R}(\boldsymbol{q}) \in SO(3)$. Having a single constraint on the parameters to determine, the estimation problem is solved using the Lagrangian function defined in (4.86). [1]

$$\mathcal{L}(\boldsymbol{q}, \lambda) = \left(vec(\hat{\boldsymbol{R}} - \boldsymbol{R}(\boldsymbol{q}))\right)^\top \boldsymbol{Q}^{-1} vec(\hat{\boldsymbol{R}} - \boldsymbol{R}(\boldsymbol{q})) - \lambda(\boldsymbol{q}^\top \boldsymbol{q} - 1) \tag{4.86}$$

The goal is to find the minimizer $\check{\boldsymbol{q}}$ of the function defined in (4.86), which is done by determining the stationary points of $\mathcal{L}(\boldsymbol{q}, \lambda)$. Computing the gradient of the Lagrangian function, the stationary points are the solution of the nonlinear system (4.87).

$$\begin{cases} -\frac{1}{2}\nabla_{\boldsymbol{q}}\mathcal{L}(\boldsymbol{q}, \lambda) = \boldsymbol{J}^\top_{\boldsymbol{R}(\boldsymbol{q})} \boldsymbol{Q}^{-1} vec(\hat{\boldsymbol{R}} - \boldsymbol{R}(\boldsymbol{q})) + \lambda \boldsymbol{q} = \boldsymbol{0} \\ -\nabla_\lambda \mathcal{L}(\boldsymbol{q}, \lambda) = \boldsymbol{q}^\top \boldsymbol{q} - 1 = 0 \end{cases} \tag{4.87}$$

To solve this nonlinear system, both the Newton-Raphson method and the Levenberg–Marquardt algorithm were implemented. Let the 5 nonlinear functions in (4.87) be lumped in vector $\boldsymbol{g}(\gamma)$, where γ contains the quaternion and Lagrange multiplier, $\gamma = \left[\boldsymbol{q}^\top, \lambda\right]^\top$. After picking an initial estimate γ^0, the Newton-Raphson method consists of using the expressions in (4.88) to iteratively improve the estimate.

$$\begin{aligned} \gamma^{s+1} &= \gamma^s + \delta\gamma^s, \\ \delta\gamma^s &= -\boldsymbol{J}^{-1}_{\boldsymbol{g},\gamma^s} \boldsymbol{g}(\gamma^s) \end{aligned} \tag{4.88}$$

Matrix $\boldsymbol{J}_{\boldsymbol{g},\gamma^s}$ is the Jacobian matrix of $\boldsymbol{g}(\gamma)$ evaluated at state γ^s and corresponds to the Hessian matrix of the Lagrangian function $\mathcal{L}(\boldsymbol{q}, \lambda)$. Its explicit derivation for the problem at hand is found in Appendix B.2. If it is close to a singular matrix, the inverted Hessian may be numerically unstable, causing the solution to diverge. This problem can be tackled by adding a positive definite matrix $\boldsymbol{M}_s$ so that $\boldsymbol{J}_{\boldsymbol{g},\gamma^s} + \boldsymbol{M}_s$ becomes positive definite. The Levenberg–Marquardt algorithm uses this correction, where $\boldsymbol{M}_s$ can be adjusted in each iteration as seen fit. However, picking the right correction $\boldsymbol{M}_s$ is not trivial. Furthermore, it was seen experimentally that, for this particular nonlinear problem, the Newton-Raphson method gives acceptable results. Therefore, the Newton-Raphson algorithm was chosen to solve this problem.

---

[1] In this context, λ denotes a Lagrange multiplier and not the wavelength of the GNSS carrier

4.6 Solution Fixing

The integer ambiguity matrices $\boldsymbol{N}$ corresponding to the solution of the MC-LAMBDA algorithm in each time step are initially saved during a certain period of time, as represented in the diagram of Section 4.2. After enough solutions are gathered, taking into account potential changes in the observed satellites or loss of lock, the fixed set of integer ambiguities is chosen as the most "popular" MC-LAMBDA solution. Note that if the observed satellites or the reference satellite changes, the set of ambiguities must be transformed accordingly. If loss of lock occurs for a certain satellite, the corresponding new DD ambiguity can be estimated using the current estimated attitude and the carrier phase observations, according to the expressions of the GNSS-based attitude model.

After this process, both the fixed ambiguity solution $\check{\boldsymbol{N}}$ and the accurate conditional quaternion $\boldsymbol{q}(\check{\boldsymbol{N}})$ can be used to improve the float solutions of the next time steps. This is done, for example, by applying a second correction step to the EKF where the fixed solutions will behave as measurements fed to the filter. The usual expressions apply, where the new state estimate and error covariance matrix are build upon the float solution EKF updated state and covariance $\hat{\boldsymbol{x}}_{k|k}$ and $\boldsymbol{P}_{k|k}$ as

$$\begin{aligned}
\boldsymbol{K'}_k &= \boldsymbol{H'}^\top \left(\boldsymbol{H'}\boldsymbol{P}_{k|k}\boldsymbol{H'}^\top + \boldsymbol{R'}\right)^{-1} \\
\hat{\boldsymbol{x}}'_{k|k} &= \hat{\boldsymbol{x}}_{k|k} + \boldsymbol{K'}_k \left(\boldsymbol{z'}_k - \boldsymbol{H'}\hat{\boldsymbol{x}}_{k|k}\right) \\
\boldsymbol{P'}_{k|k} &= \left(\boldsymbol{I} - \boldsymbol{K'}_k\boldsymbol{H'}\right)\boldsymbol{P}_{k|k}\left(\boldsymbol{I} - \boldsymbol{K'}_k\boldsymbol{H'}\right)^\top + \boldsymbol{K'}_k\boldsymbol{R'}\boldsymbol{K'}_k^\top,
\end{aligned} \tag{4.89}$$

where the residual $\boldsymbol{z'}_k - \boldsymbol{H'}\hat{\boldsymbol{x}}_{k|k}$ is written as

$$\boldsymbol{z'}_k - \boldsymbol{H'}\hat{\boldsymbol{x}}_{k|k} = \begin{bmatrix} \boldsymbol{q}(\check{\boldsymbol{N}}) \\ \check{\boldsymbol{N}} \end{bmatrix} - \begin{bmatrix} \boldsymbol{I}_4 & 0 & 0 \\ 0 & 0 & \boldsymbol{I}_{mn} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{q}}_{k|k} \\ \hat{\boldsymbol{b}}_{k|k} \\ \hat{\boldsymbol{n}}_{k|k} \end{bmatrix} \tag{4.90}$$

and $\boldsymbol{R'}$ is the covariance matrix of the measurements $\boldsymbol{z'}_k$. An expression for the covariance matrix of the conditional solution, $\boldsymbol{Q}_{\boldsymbol{q}(\boldsymbol{N})\boldsymbol{q}(\boldsymbol{N})}$ has been deduced in the previous Section. However, it is chosen to tune $\boldsymbol{R'}$ manually so as to obtain the best compromise between improving the accuracy of the float solution, while still allowing the next step of the EKF to change the state estimates, if needed. This updated solution defined by $\hat{\boldsymbol{x}}'_{k|k}$ and $\boldsymbol{P'}_{k|k}$ is then used in the EKF propagation step of the next time step, closing the loop.

Note that since the fixed ambiguities are computed using the float solution provided by the EKF, the two update steps are not independent. Therefore, the observations of this second update step of the filter are not uncorrelated with the pseudorange and carrier phase observations used to compute the float solution. Nonetheless, since this correlation is complex to quantify due to the nature of the constrained ILS problem, the filter will be used as described.

After the integer ambiguities are fixed, in each time step, the solution of the MC-LAMBDA is fed to this second update step of the EKF. It is expected that since the correct integer ambiguities have been found, the solution of the integer ambiguity determination algorithm will not change.

Chapter 5

Implementation and Data Acquisition

In order to verify and assess the performance of the attitude estimation solution proposed in the previous chapter, a software system was designed and implemented.

The system is initially meant to post-process the GNSS and gyroscope data. This framework allows for faster development of the software and easier debug, should a problem arise during the processing of the data. The software is developed with real-time applications in mind and can easily be modified to suit these requirements in the future. Further details about the data processing and system inputs and outputs are given in section 5.1.

Due to hardware purchase restrictions, owning to the Covid-19 pandemic, experimental tests were done using only two pairs of GNSS antennas and receivers, which is critical when trying to estimate the full attitude of the body. As such, a simple simulation environment was developed, providing GNSS observations across any number of receivers that allows simulating different body kinematics, as explained in section 5.2.

Finally, in section 5.3, the hardware used for the experimental testing of the software is described and a representation of the test vehicle used for acquiring GNSS and IMU data during the test trials is given.

5.1 Software Design and Overall Structure

The software development was done in the MATLAB® environment. This software is suitable for computational algorithms intensive on matrix operations, as is the present case. Note also that at this point of development, the processing time and memory efficiency are not the main priority.

An overview of the main building blocks constituting the developed software is given in Figure 5.1.

Most of the individual steps represented in this flowchart have already been described as they pertain to the proposed attitude estimation method. Apart from these, the pre-processing of the observations collected by the GNSS receivers is highlighted.

The receivers are configured to provide the raw data in the Receiver INdependent Exchange Format (RINEX) format, which is one of the standard formats for this type of application and is meant to provide

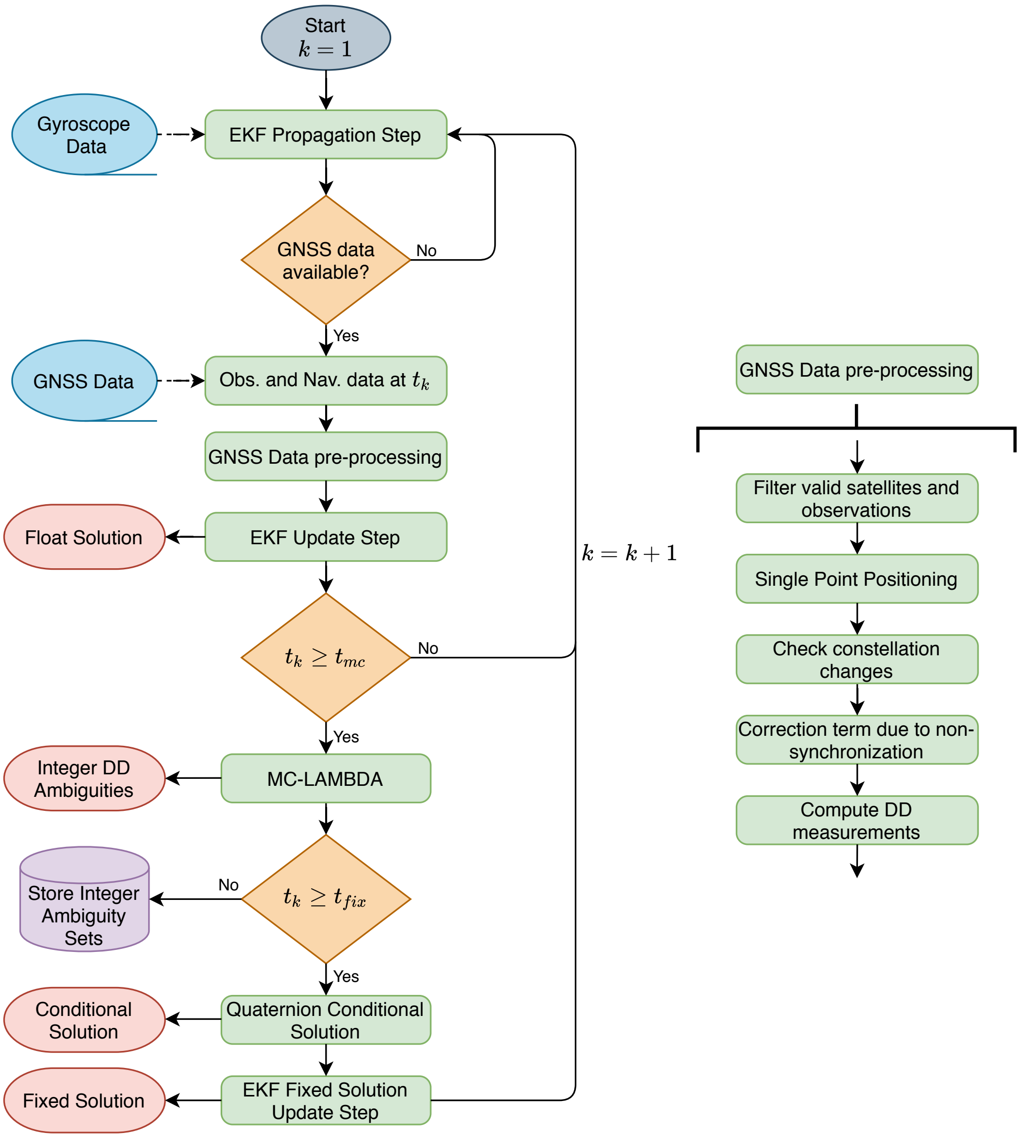


Figure 5.1: Detailed flowchart of the algorithm used to process the GNSS and gyroscope data.

a common framework, independent of brand or cost of the GNSS receiver (as the name entails). This format provides the required observables such as the pseudorange and carrier-phase measurements, along with the navigation message containing the ephemeris data. For more on this standard, see [39].

The first block of the pre-processing layer shown illustrated on the right part of the flowchart deals with selecting the observed satellites common to every receiver which provide valid data. Then, a simple Single Point Positioning algorithm is used to compute the approximate position of the reference receiver. The position is used to form the LOS vectors and choose the reference satellite, based on the elevation angle. To improve the accuracy of this algorithm, a tropospheric delay model is implemented, based on [27], and navigation data parameters are used to compute the satellite clock delay.

Having the list of observed satellites and reference satellite for the present epoch, these are compared with the ones of the previous measurements, in the *Check constellation changes* block. The algorithm is designed to use as much observation data as possible, and so if new satellites are in view, the size of the vector of DD ambiguity estimates is increased accordingly. The new estimates are either initialized with no *a priori* knowledge, or a first guess is obtained using the carrier-phase observations model and the attitude estimate. Furthermore, if the reference satellite changes, the DD ambiguities in $\hat{\boldsymbol{x}}_{k|k}$ and respective terms in the error covariance matrix must be changed accordingly to guarantee a smooth transition. All the expressions used to deal with changes in observed satellites or reference satellite are included in Appendix C.

The experimental setup used does not allow for synchronization in real time between receivers. While it is expected that the sampling of the GNSS signals is done at the same time due to the very short baseline length, it is confirmed experimentally that the receivers are not synchronized, with the difference between the two receiver clocks (corrected by the clock offsets) reaching $1\,\mathrm{ms}$ in some cases. This time difference introduces an error in the model of DD measurements, which is modelled in the next pre-processing block, and corrected in the vector of DD observations that is fed to the correction step of the EKF. The model and assumptions used for this correction are written in Appendix D.

The remaining part of the algorithm, relative to the attitude estimation and integer ambiguity resolution, mostly follows the steps described in the previous chapter.

Finally, note that due to the linearization implicit in the EKF, some caution when initializing the filter must be taken into account. In this work, the initialization of the estimated attitude and DD ambiguities at t_0 is done using a gross approximation which may be far from the true state of the system. To solve this issue, the covariance matrix $\boldsymbol{R}_k$ is tuned so that in the beginning, the filter has lower sensitivity to the carrier-phase measurements. It was seen experimentally that this adjustment allows the filter to converge to the correct solution even if the initial guess is far from optimal, although with lower accuracy. After some time, $\boldsymbol{R}_k$ is adjusted according to the noise model of the measurements, increasing the bandwidth of the filter relative to the carrier-phase measurements and therefore the accuracy of the solution. This process of adjusting the $\boldsymbol{R}_k$ matrix can be seen clearly in the results presented in the next Chapter. It is also important to know that the integer ambiguities are determined from the float solution only after it is considered that the filter has converged close to the true state. This is defined by the manually tuned parameter t_{mc}, as seen in the flowchart.

5.2 Simulation Setup

The implemented algorithm receives as input GNSS observations from the GNSS receivers connected to the antennas and angular velocity measurements from a 3-axis rate gyroscope. The software is designed so that it does not distinguish between simulation or experimental data. Consequently, the developed simulation script must also generate sample RINEX files and angular velocity measurements.

First, the real attitude of the body in relation to the local ENU frame in each time instant is simulated. The exact discrete time attitude kinematics in (4.15) are used, receiving as input the simulated exact angular velocity. The simulation allows choosing a static configuration or a dynamic scenario where any kind of rotation can be simulated. The discrete-time gyro model (4.18) is used to simulate the noisy gyroscope measurements, where the parameters σ_v and σ_u can be tuned to increase or decrease the angular random walk (ARW) and rate random walk (RRW).

To arrive at the GNSS measurements, the stochastic models introduced in section 2.3 are used. The expressions for the pseudorange and carrier phase measurements in (2.11) and (2.12) include ionosphere and troposphere delay terms, as well as unmodelled error such as multipath. Instead of considering each individual error source, the following model for the variance of the measurements is used [6]:

$$\begin{cases} \left(\sigma_{P,i}^{k}\right)^2 = 100^2 \left(c_1^2 + \left(\frac{c_2}{\sin \mathrm{el}_k}\right)^2\right) \\ \left(\sigma_{\Psi,i}^{k}\right)^2 = c_1^2 + \left(\frac{c_2}{\sin \mathrm{el}_k}\right)^2 \end{cases} , \tag{5.1}$$

where $\sigma_{P,i}^{k}$ is the standard deviation of the code observations relative to satellite k at receiver i and $\sigma_{\Psi,i}^{k}$ is relative to the carrier phase. The carrier phase variance is modelled as being four orders of magnitude smaller than the corresponding pseudorange variance, and both are written in units of m^2. According to this model, the noise parameters depend only on the elevation angle el_k of satellite k, which is assumed the same for every receiver. Constants c_1 and c_2 are tuned according to the scenario one wishes to simulate. It is suggested in [6] that using $c_1 = c_2 = 2\,\mathrm{mm}$ provides realistic noise levels.

To compute the elevation angle as well as the distance term in the observation models, ρ_i^k, the satellite's position is required. The positions are calculated using sample navigation data collected during a real experiment. In this simplified simulation, both the satellites and the Earth's surface are considered static. Furthermore, the receiver and satellite clock offsets are not simulated. As such, the only simulated source of error is the realization of a white-noise like process with the variances in (5.1), which is added to the distance term.

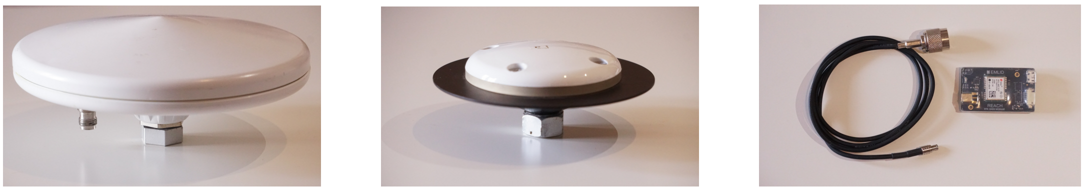
Lastly, the integer ambiguities are generated using a pseudo random number generator and rounding to the nearest integer. The range of possible integers is simulated such that the resulting DD ambiguities are of a similar order of magnitude to those determined experimentally.

5.3 Experimental Setup

The equipment used to collect data throughout the static and dynamic experiments is enumerated in the following. The configuration of baseline and gyroscope used to perform the actual experiments is also depicted.

5.3.1 Hardware Overview

As mentioned before, the experimental testing of the algorithm is restricted by the available hardware. Only two GPS antennas are used, as shown in Figure 5.2. Some relevant electrical specifications are presented in Table 5.1 [40, 41]. The AT575-75 antenna is chosen as the reference antenna to perform the double difference computations. Note that it would be optimal if the antennas were of the same brand and model since the noise parameters relative to the measurements, in the algorithm, are assumed the same for the two antennas. Also, since the measured and estimated coordinates of the baselines are relative to the antenna phase centers, the phase center offset specified in the datasheet of the antennas, given in Table 5.1, has to be taken into account.


(a) AT575-75 antenna by AeroAntenna Technology.

(b) 53G15A antenna by Antcom Corporation.

(c) Reach Module by Emlid.

Figure 5.2: GNSS antennas and receiver hardware used in the experimental trials.

Table 5.1: Electrical specifications of the GPS antennas.

Antenna Part Number (P/N)	AT575-75	53G15A
Frequency (MHz)	1575.42 ± 6	1575.42 ± 12
Impedance (Ohm)	50	50
Antenna Gain (dBic):		
@ 90° (Zenith)	4.0	5.4
@ 30°	-1.0	1.8
@ 20°	-2.5	0.4
@ 10°	-4.5	-1.3
Z Phase Center Offset (cm)	4.8	1.3

One of the two identical GNSS receivers used is shown on the right, in Figure 5.2c. These are low-cost Reach Module receivers by Emlid, typically used for real-time kinematic (RTK) positioning applications. However, they can be configured to provide raw data measurements in the RINEX format using the Reach interface. Its general specifications are available in [42]. While the number of channels is not specified, it is seen experimentally that it can track every satellite constellation simultaneously, including of course the GPS satellites.

As for the gyroscope, the only available solution was using a smartphone's IMU, which is equipped with a multitude of sensors, such as a 3-axis accelerometer, gyroscope and magnetometer. The raw angular velocity data was recorded and downloaded using the Sensor Lab application [43]. While smartphones' sensors naturally are not suitable for precise applications, the main maneuvers done in the dynamic tests are clearly seen in the angular velocity data, as shown in the next chapter. However, using a dedicated, more precise rate gyroscope would obviously enhance the prediction of the body orientation.

5.3.2 Testing Platform

Prior to setting up the antennas and gyro configuration for either the static or dynamic tests, ground planes are placed under the antennas, as recommended in standard antenna placement guidelines. The thin metal pieces shown in Figure 5.3 reduce multipath effects, therefore providing shielding and improving signal reception.

In every experiment, the antennas are connected to the GNSS receivers using coaxial cables. The power is supplied by a laptop which is connected to the Micro-USB port of the receiver. Lastly, the smartphone used as a gyroscope is fixed to the support connecting the two antennas, rotating with the

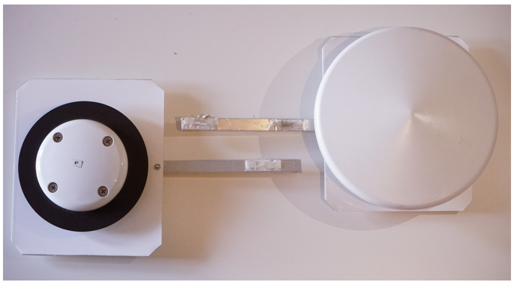


Figure 5.3: GPS antennas mounted with ground planes.

baseline and therefore the body frame.

The setup used in the static trails is shown on the left in Figure 5.4a. The thin beam connecting the antennas is set on a tripod, used to further reduce multipath effects, while also enabling a clearer view above the horizon. The electronic devices, namely the receivers and the laptop, are placed as far as possible from the antennas to lessen interference due to any radio frequency (RF) noise generated from these devices.

The dynamic trials were done by placing the antennas, along with the smartphone, on the roof of a car, as shown in Figure 5.4b. This very simple setup allows placing the antennas a certain height above the ground, while the ground plates minimize the multipath interferences reflected on the car's surface. One downside of this setup is the low range of motion in the pitch and roll axes during the experiments, since this depends on the shape of the terrain. A workaround to this issue is manually rotating the baseline during the experiment, as described in the test trials.

(a) Antenna configuration used in the static trials.

(b) Antenna configuration used in the dynamic trials.

(c) Gyroscope (smartphone) configuration in respect to the baseline.

Figure 5.4: Setup of the equipment required to obtain the GPS and IMU observables.

Chapter 6

Trials Description and Results

In this Chapter, the proposed solution for solving the problem of attitude estimation using GNSS observations as well as IMU measurements is validated by performing several trials.

As mentioned in the previous Chapter, due to a shortage of available equipment, experimental data is acquired using only two GNSS receivers. Furthermore, evaluating the estimation error of the proposed solution is only possible when there is a trustworthy reference of the body's attitude, which would imply using specialized equipment or a different, very accurate attitude estimation technique. Using the developed simulation environment, these issues are somewhat avoided since the true attitude of the body is known and any dynamic scenario, as well as number of baselines, can be generated. The results obtained using simulated data are presented in Section 6.1.

The end goal of the development of any algorithm used for navigation is its implementation and testing in a real-life scenario. Furthermore, GNSS applications have long been developed, implemented and tested in realistic scenarios. As such, it is deemed fit that the algorithm be validated using experimental data, even if the range of different trials that can be performed is limited. The results of the performed experiments, using the setup described in the previous Chapter, are given in Section 6.2.

6.1 Simulation Results

In order to evaluate the performance of the developed algorithm under different scenarios, several trials are simulated and analyzed in the following.

Whether in the implemented simulation environment or processing algorithm, there are some parameters that must be tuned. An overview of these parameters, as well as the values assigned to them throughout the different trials, is given in Section 6.1.1.

Then, a parametric analysis of the performance of the implemented integer ambiguity determination framework, based on the MC-LAMBDA method, is given in the next Section. It is shown how the method performs under more and less trying scenarios, which is crucial when wishing to use the highly accurate carrier-phase measurements.

In what remains, the actual attitude estimates are presented in different trials where the platform can be either static or dynamic. Each trial showcases both the performance and limitations of the algorithm under different scenarios, where special attention is given to the single baseline observability results presented as part of the dynamic trials.

6.1.1 Simulation and Processing Parameters

The implemented simulation environment provides sample GNSS observation and navigation files, as well as angular velocity measurements. The values assigned to parameters common to every analyzed simulated trial are first enumerated in the following.

The noise levels of the simulated gyroscope measurements depend on parameters σ_v and σ_u, according to the discrete-time stochastic model in (4.18). These were tuned so that the noise power spectral density (PSD) of the simulated measurements is similar to that of the experimental measurements, obtained using a smartphone. Note that the variance of the simulated white noise affecting the bias evolution in time, the RRW, is slightly increased in order to better show the evolution of the gyro bias estimate. The values of σ_v used throughout all simulations, for each axis, are $0.1921 \times 10^{-3}\,\mathrm{rad \cdot s^{-\frac{1}{2}}}$, $0.2200 \times 10^{-3}\,\mathrm{rad \cdot s^{-\frac{1}{2}}}$ and $0.1975 \times 10^{-3}\,\mathrm{rad \cdot s^{-\frac{1}{2}}}$. The corresponding values for σ_u, determining the RRW, are $0.0696 \times 10^{-3}\,\mathrm{rad \cdot s^{-\frac{3}{2}}}$, $0.0556 \times 10^{-3}\,\mathrm{rad \cdot s^{-\frac{3}{2}}}$ and $0.0353 \times 10^{-3}\,\mathrm{rad \cdot s^{-\frac{3}{2}}}$.

Concerning the GNSS observations, there are several parameters that must be fixed according to the procedure and models used to simulate the observables, which was detailed in Section 5.2. The variance of the code and carrier-phase measurements is seen to depend on the parameters c_1 and c_2, apart from the elevation angle of the satellites. In the simulated static and dynamic trials detailed in the following, the values used are $c_1 = c_2 = 3\,\mathrm{mm}$, which corresponds to a more trying scenario than the value of $2\,\mathrm{mm}$ suggested in the literature. The position of the reference antennas is set to the geodetic coordinates of Sátão, a town in Viseu, Portugal, given by the latitude, longitude and height above mean sea level $\phi = 40.753\,031\,°$, $\lambda = -7.715\,333\,°$ and $h_1 = 652\,\mathrm{m}$. Lastly, the satellite positions are computed using the navigation files collected at UTC time $18 : 30 : 00$ on $2020/09/16$, where 7 GPS satellites are observed.

The position of the remaining antenna are simulated along the direction of axis X and Y of the body frame, up to a maximum of two baselines. That is, in the single baseline case, the coordinates of the baseline in the body frame are $\boldsymbol{l}_{i1} = \|\boldsymbol{l}_{ik}\| \, [1, 0, 0]^\top$, and having multiple baselines, a second baseline $\boldsymbol{l}_{i2} = \|\boldsymbol{l}_{ik}\| \, [0, 1, 0]^\top$ is added. All baselines are equal in length, $\|\boldsymbol{l}_{ik}\|$, which is fixed at $\|\boldsymbol{l}_{ik}\| = 1.3\,\mathrm{m}$ in the static and dynamic trials.

To compute the baseline coordinates in the local frame along time, the initial attitude is set at the arbitrary Euler angles $\varphi = 10\,°$, $\theta = -25\,°$ and $\psi = 40\,°$. Lastly, the sampling frequencies of the simulated gyroscope measurements and GNSS observations are chosen as $20\,\mathrm{Hz}$ and $5\,\mathrm{Hz}$, respectively, matching those of the experimental data.

The remaining parameters are relative to the processing of the measurements. Concerning the EKF, the state and respective covariance matrix must be initialized. The attitude is initialized by considering that the body frame is aligned with the reference frame, while the bias and DD ambiguities estimates are also initialized at 0. The initial estimation error covariance matrix is defined as a diagonal matrix. The quaternion estimate's error covariance matrix is set equal to the identity matrix, while the variance of each gyro bias estimate has a value of $2\,\text{rad}^2/\text{s}^2$. The initial variance of each DD ambiguity estimate is set to 1×10^4, since its magnitude is initially unknown.

The continuous-time process noise matrix $\boldsymbol{Q}$, relative to the gyroscope measurements, is defined according to the σ_v and σ_u noise parameters written above, the only difference being that the RRW is set two orders of magnitude lower. Matrix $\boldsymbol{R}_k$, as mentioned in Section 5.1, is defined differently according to the processing time. The used expressions for the observations' variance are written in (6.1), where $t_{\boldsymbol{R}_{chg}}$ is the time at which the filter bandwidth relative to the carrier phase measurements is narrowed, increasing the accuracy of the estimation. Then, $\boldsymbol{R}_k$ follows according to (3.13).

$$\begin{aligned} &\begin{cases} \left(\sigma^k_{P,i}\right)^2 = 100^2\left(c_1^2 + \left(\frac{c_2}{\sin \text{el}_k}\right)^2\right) \\ \left(\sigma^k_{\Psi,i}\right)^2 = 5 \times 10^{-1}\left(\sigma^k_{P,i}\right)^2 \end{cases} , \quad \text{if} \ \ t < t_{\boldsymbol{R}_{chg}} \\ &\begin{cases} \left(\sigma^k_{P,i}\right)^2 = 100^2\left(c_1^2 + \left(\frac{c_2}{\sin \text{el}_k}\right)^2\right) \\ \left(\sigma^k_{\Psi,i}\right)^2 = 5 \times 10^{-3}\left(\sigma^k_{P,i}\right)^2 \end{cases} , \quad \text{if} \ \ t \geq t_{\boldsymbol{R}_{chg}} \end{aligned} \tag{6.1}$$

In every trial, the transition time used is manually adjusted to $t_{\boldsymbol{R}_{chg}} = 50\,\text{s}$. This concludes the setup of the EKF float solution, from which follows the part of the algorithm dealing with the integer ambiguities.

As shown in the flowchart of the overall algorithm in Section 5.1, the integer ambiguity determination method and fixing of the respective ambiguities occur only after reaching the processing times t_{mc} and t_{fix}, respectively. Through trial and error, it was considered that at $t_{mc} = 100\,\text{s}$, the EKF estimates can be assumed to have converged near to the true values. Then, the integer ambiguities storing period before fixing is set to last $50\,\text{s}$, which leads to $t_{fix} = 150\,\text{s}$.

Lastly, the covariance matrix $\boldsymbol{R'}$ used in the 2nd update step of the EKF, during the Solution Fixing step, was manually tuned to the identity matrix of appropriate size multiplied by 1×10^{-6}. In the following, the simulation results obtained using these parameters are presented and discussed.

6.1.2 Integer Ambiguity Determination Success Rate

First off, the success rate of the implemented MC-LAMBDA method, when integrated with the EKF float solution, is evaluated by performing Monte Carlo simulations under different scenarios. The simulated scenarios are initially detailed, followed by the success rate results.

Trial Description

Since this trial is concerned only with the performance of the integer ambiguity determination algorithm, during the processing runs the fixing of the ambiguities is not performed. That is, the time at which fixing occurs, t_{fix}, is set greater than the total time of the simulated data. As such, the sets of integer ambiguities evaluated correspond to the stored integer ambiguities represented in the flowchart of Figure

5.1, which are computed starting at t_{mc}, until the end of the processing. If the ambiguities were fixed, the output of the MC-LAMBDA from that point on would very likely be equal to the set of fixed ambiguities, providing no further information about the success rate of the MC-LAMBDA method.

The success rate is defined as the percentage of epochs satisfying $t > t_{mc}$ in which the output of the MC-LAMBDA method is equal to the correct set of integer ambiguities. It is expected that the success rate varies according to how noisy the observations are or how many satellites are being tracked, for example. With this in mind, different baseline lengths, observation noises and number of observed satellites are simulated in order to see how these affect the success rate of the chosen method. For each combination of the parameters, several Monte Carlo simulations are done, from which the success rates are computed. In every simulation, a single baseline is used.

Success Rates

The simulated scenarios in terms of baseline length, pseudorange noise and number of satellites used are represented in Table 6.1. For each combination, the resulting success rate is presented. Note that as the pseudorange noise varies, so does the carrier phase noise, since the parameters that are directly changed are c_1 and c_2 of model (5.1). Furthermore, the indicated code measurements noise corresponds to that of the observations of the satellite with highest elevation angle. Therefore, it represents the minimum value of the code measurements noise of all satellites, as the noise increases for lower elevation angles. Also, for simulations with less than 7 satellites, the discarded satellites are those with higher elevation angle, which further increases the overall noise of the measurements.

Table 6.1: Integer Ambiguity Determination success rates of the EKF+MC-LAMBDA integration under different scenarios.

Baseline length $[\mathrm{m}]$	1			5			10		
Code noise $\sigma^k_{P,i}$ $[\mathrm{cm}]$	50	40	30	50	40	30	50	40	30
$\#$ Satellites				**Success rates** $(\%)$:					
5	83.96	94.75	97.30	96.85	97.23	96.81	95.41	98.37	98.29
6	93.08	98.92	98.56	98.31	98.15	97.97	98.40	97.87	99.06
7	95.05	97.34	98.77	98.18	98.97	98.55	98.42	98.53	97.67

There is a clear increase in the success rates of the ambiguity determination as the measurements noise decreases or the number of tracked satellites is higher, which can be seen in the simulation results corresponding to a unitary baseline length. It is natural that having less noisy observations leads to a more accurate float solution, which is the starting point of the search process done in the MC-LAMBDA algorithm, increasing its success rate. Note that a code noise of 50 cm is quite high when compared to a baseline length of 1 m, and even in this case a high success rate of 83.96% is achieved.

As more satellites are available, more information is fed to the filter, improving the attitude estimates along with the accuracy of the integer ambiguity candidates.

As for the baseline length, it is expected that shorter baselines hinder correct integer ambiguity determination. This effect is not only due to having less precise attitude estimates, but also since the observation model used in the EKF is highly nonlinear for very short baselines. As seen in Table 6.1, the correct set of ambiguities is determined more often when simulating a baseline of 5 m, compared to 1 m.

Overall, since the float solution is obtained using a recursive filter, the estimated float DD ambiguities are already very accurate, although not exact, improving the results of the MC-LAMBDA method. If more baselines were used, it is expected that the success rate increases since the method uses the knowledge of the relative geometry of the baselines, implicit in the orthogonality of the conditional rotation matrix. This being said, even in a single baseline scenario the results are likely to be better than those obtained using, for example, the simpler LAMDBA method, since the MC-LAMBDA also takes the known baseline length into account. Also, note that the simulated code and carrier-phase noises are considerably high, since a code noise of 30 cm reasonably approximates a real-life scenario, which also confirms the good performance of the integer ambiguity determination algorithm. Given these results, it is very likely that the integer ambiguity set chosen for the fixing is the correct one, since it only needs to occur at least 50% of the time during the ambiguity storing period.

6.1.3 Static Trial

In the following, the attitude estimate solution as well as the relevant intermediate solutions of the developed algorithm are presented and analyzed. First, an overview of the simulated static trial used to generate the required data is given.

Trial Description

Since in this trial the body is considered static, the simulated measured angular velocity consists only of noise. The 3-axis angular velocity measurements and simulated gyroscope bias are represented in Figure 6.1. The parameters related to the evolution of the bias were clearly exaggerated in order to better assess the performance of the filter, as mentioned previously.

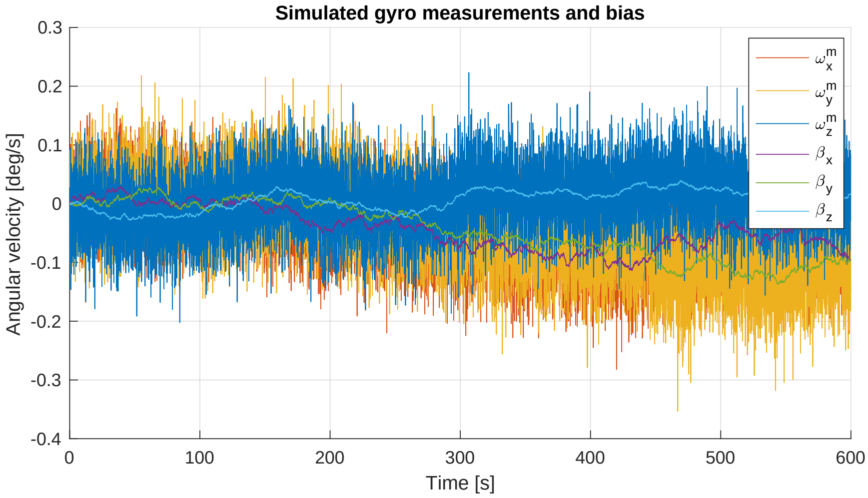


Figure 6.1: Simulated angular velocity measurements for the static trial.

It is expected that in a static scenario, a single baseline is not enough to estimate the full attitude of the body, nor the gyroscope bias. Therefore, two baselines are simulated, as seen in the results that follow. Note that all parameters not mentioned here are set to the values defined in Section 6.1.1.

Multiple Baseline Scenario Results

In order to better show and evaluate the characteristics of each intermediate and final attitude solution, two different processing approaches are used. The first, hereby called *EKF+MC-LAMBDA (w/o Fix)*, corresponds to the results of the MC-LAMBDA solution without fixing the integer ambiguities, as done in the previous section. As such, the attitude estimates shown are the conditional solutions at each epoch for $t > t_{mc} = 100\,\mathrm{s}$. The other solution, *EKF+MC-LAMBDA Fixed* is the more accurate fixed solution, where a starting time of $t_{fix} = 150\,\mathrm{s}$ is used. In this case, the shown solutions are part of the updated state of the EKF after the 2nd Update Step corresponding to the Solution Fixing. Using the simulated data from the static trial having two baselines, the results in Figure 6.2 are obtained.

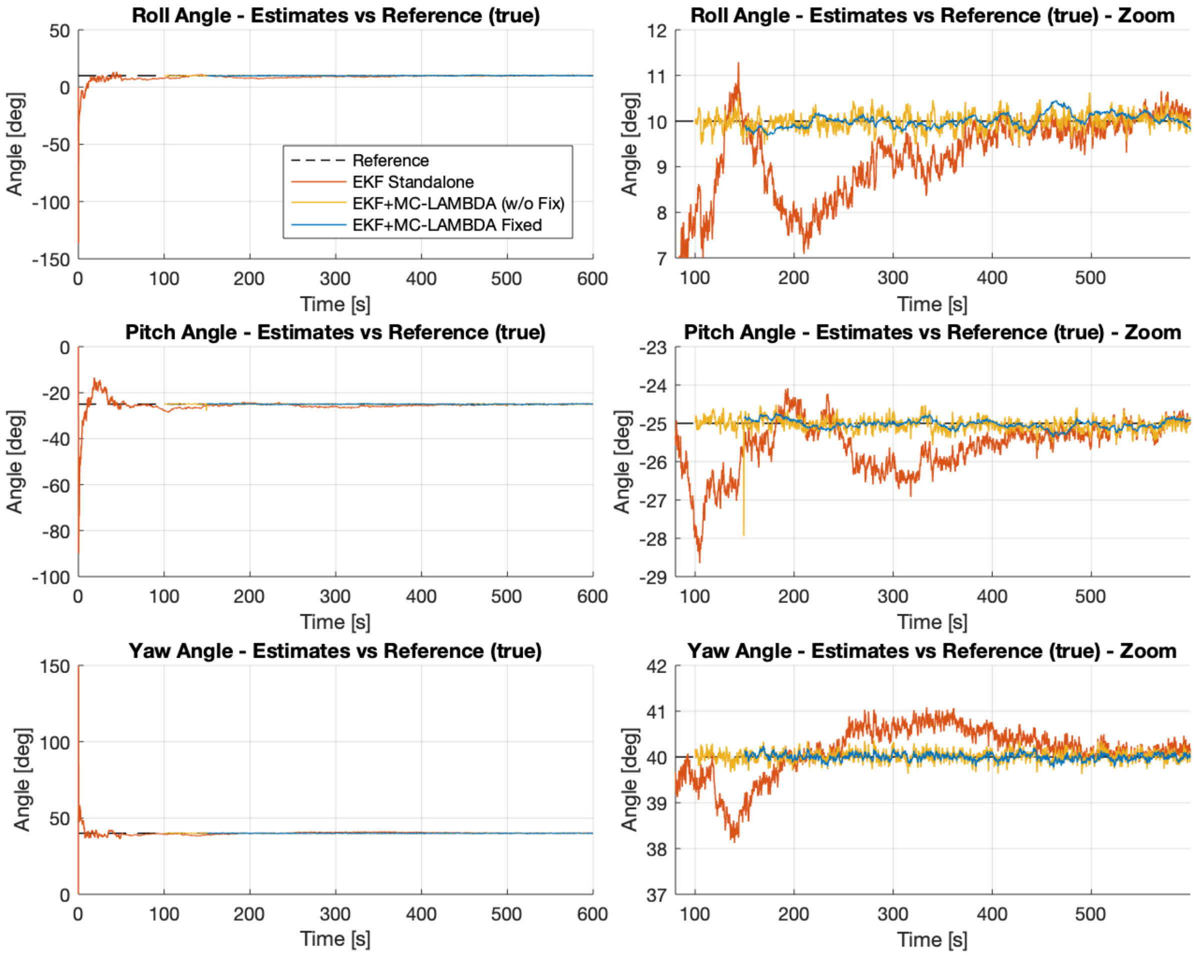


Figure 6.2: Static trial attitude estimates, including intermediate solutions.

The *EKF Standalone* solution corresponds to the EKF float solution without fixing the ambiguities. It is included in the plots in order to compare the accuracy of this intermediate solution with the conditional and fixed solutions. The estimation error mean value and standard deviation relative to each of the

Table 6.2: Performance metrics of the estimated attitude solutions for the static trial, evaluated for $t > t_{fix}$.

Euler Angle	Solution	Error Mean $[\mathrm{deg}]$	Error Std. $[\mathrm{deg}]$
Roll φ	EKF Standalone	0.673	0.756
	EKF+MC-LAMBDA w/o Fix	0.003	0.181
	EKF+MC-LAMBDA Fixed	-0.009	0.147
Pitch θ	EKF Standalone	0.435	0.502
	EKF+MC-LAMBDA w/o Fix	0.038	0.156
	EKF+MC-LAMBDA Fixed	0.021	0.111
Yaw ψ	EKF Standalone	-0.275	0.396
	EKF+MC-LAMBDA w/o Fix	-0.016	0.115
	EKF+MC-LAMBDA Fixed	-0.001	0.072

shown attitude solutions are summarized in Table 6.2.

The plots on the left of Figure 6.2 clearly show that the float solution, after around $50\,\mathrm{s}$, converges to the true attitude, even though the filter's initial estimated Euler angles are off by 10 to 40 degrees. It is also seen that at $t = 50\,\mathrm{s}$, the EKF solution becomes more accurate due to the tuning of matrix $\boldsymbol{R}_k$, as previously described. From the close up of the estimates on the right of Figure 6.2, it is noted that both the conditional solution without fixing and the fixed solution are more accurate than the float solution. Of the two, the fixed solution is overall closer to the true value, as seen in Table 6.2. Since, in this case, the correct set of integer ambiguities is fixed at $t = t_{fix} = 150\,\mathrm{s}$, in the following epochs the accuracy is improved. This result validates the chosen approach of fixing the integer ambiguities in a 2nd EKF Update Step, since the accuracy of the attitude estimation is improved.

It is also important to note the high accuracy of the float solution, which is characterized by sub degree standard deviation. This is possible due to using a recursive filter, where the estimates are improved over time as new measurements are received. The accuracy of the float solution using the typical LS approach would be lower since it is an epoch-by-epoch method. Furthermore, the used approach successfully integrates the gyroscope measurements, further improving the accuracy of the estimate.

The quantities that remain to be discussed are the 3-axis gyroscope bias. On the left of Figure 6.3, the *EKF Standalone* and fixed solutions are presented, alongside the simulated true values. On the right, a close up view of the estimates is shown. Once again, Table 6.3 summarizes the error metrics of the considered solutions.

Having two baselines, the bias estimates converge to the simulated values, as seen in the plots. Being able to estimate the gyroscope bias in this scenario is a result which had already been hinted at in the observability analysis done before. Having a good estimate of the gyro bias, the true angular velocity's mean value is estimated accurately, improving the predicted state estimate of the filter.

The bias estimate when fixing the ambiguities is slighty more accurate, as seen in Table 6.3. In the model of the nominal system, there is no direct relation between the gyro bias and the integer ambiguities. However, the MC-LAMBDA method improves the quality of the attitude estimates. Since

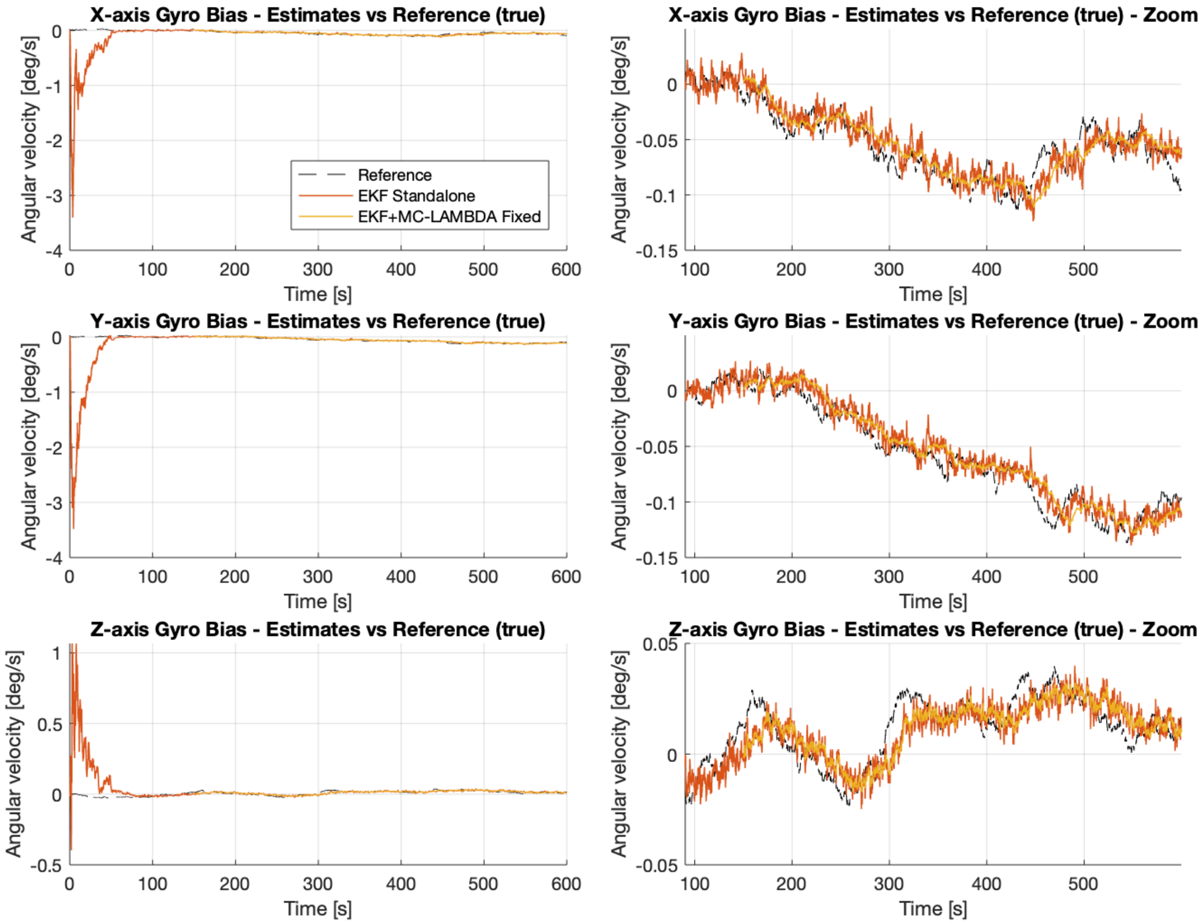


Figure 6.3: EKF without fixing and fixed solutions' bias estimates during the simulated static trial.

Table 6.3: Performance metrics of the estimated gyro bias for the static trial, evaluated for $t > t_{fix}$.

Gyro bias	Solution	Error Mean $[\mathrm{rad/s}]$	Error Std. $[\mathrm{rad/s}]$
X-axis β_x	EKF Standalone	-0.0032	0.0146
	EKF+MC-LAMBDA Fixed	-0.0031	0.0139
Y-axis β_y	EKF Standalone	-0.0048	0.0117
	EKF+MC-LAMBDA Fixed	-0.0044	0.0119
Z-axis β_z	EKF Standalone	0.0007	0.0094
	EKF+MC-LAMBDA Fixed	0.0006	0.0089

the predicted state at a given epoch depends of the fixed attitude solution of the previous epoch, the structure of the filter makes it so that the bias estimate becomes more precise.

6.1.4 Dynamic Trials

Scenarios in which the body frame is rotating with respect to the local frame were also simulated. In the following, the simulated trials are briefly described and then the corresponding results are presented.

Trials Description

Several different dynamic trials can be simulated since both the number of baselines and their coordinates, as well as the angular velocity of the body can be varied freely. In this work, two different scenarios are simulated, one using two baselines and other where a single baseline configuration is simulated.

A dynamic trial using two baselines allows comparing the performance of the developed algorithm in a dynamic scenario as opposed to the static trial previously presented. The simulated baseline coordinates in this case, as well as all other simulation parameters were described in Section 6.1.1. The measured angular velocity, from which the simulated dynamics are fully described, is given in Figure 6.4. The resulting pseudorange and carrier-phase observations are used to estimate the attitude of the body, shown in the next Section.

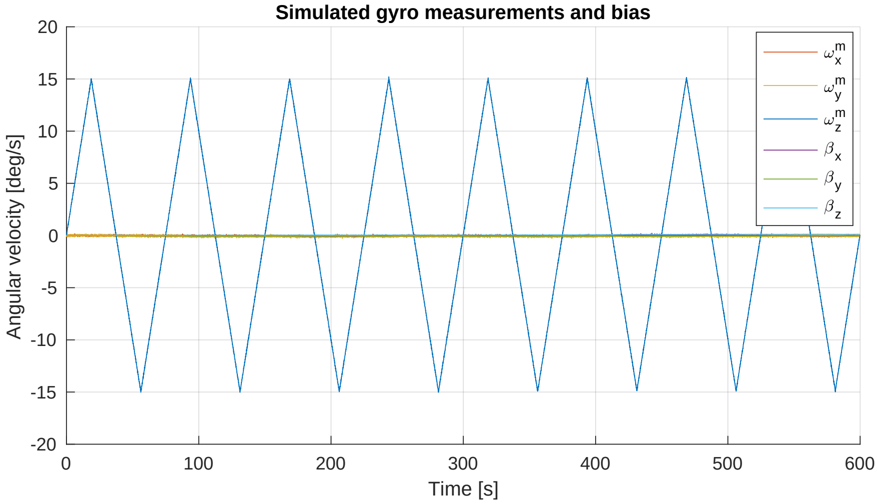


Figure 6.4: Simulated angular velocity measurements for the dynamic trials.

A single baseline case is simulated in order to experimentally check the observability of the filter in these conditions. It was deduced, using a linearized model, that if the body is static, a single baseline is not enough in order to estimate the full attitude and gyro bias. Since the dynamic case could not be studied analytically, it will be seen experimentally if the state can be estimated. The simulated body motion in this trial is exactly the same as the dynamic multiple baseline scenario, represented in Figure 6.4.

Multiple Baseline Scenario

Similarly to the simulated static trial results, the estimated attitude and corresponding error metrics are given in Figure 6.5 and Table 6.4.

First off, it is noted from the left plots of Figure 6.5 that, when compared to the previous static trial, the *EKF Standalone* solution converges more quickly to the true estimate. Naturally, since there is a greater diversity in the measurements fed to the filter due to the body motion, it has more information in order to estimate the state.

The increased excitation of the filter also results in a more accurate float estimate when compared

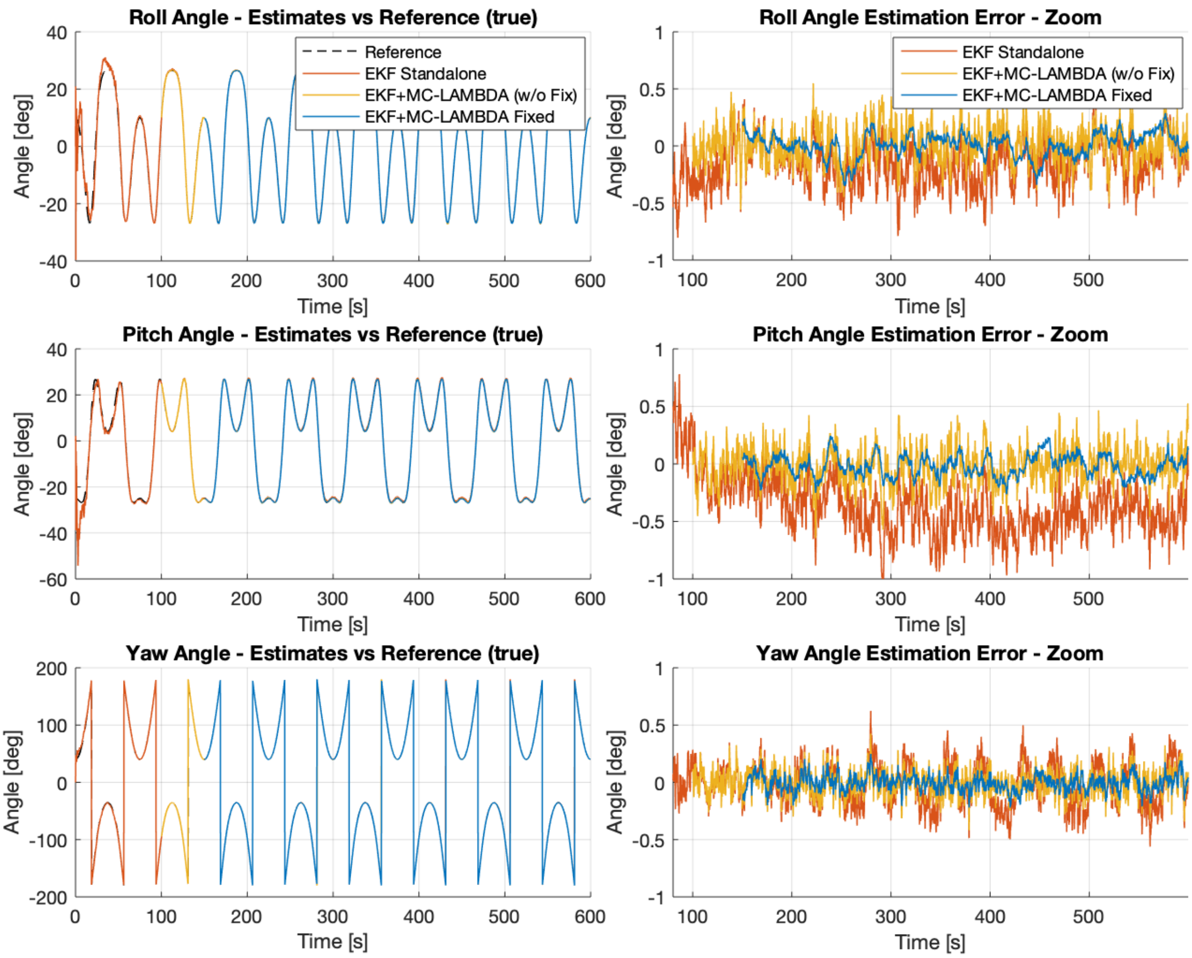


Figure 6.5: Dynamic trial, multiple baseline attitude estimates, including intermediate solutions.

Table 6.4: Performance metrics of the estimated attitude solutions for the multiple baseline dynamic trial, evaluated for $t > t_{fix}$.

Euler Angle	Solution	Error Mean $[\mathrm{deg}]$	Error Std. $[\mathrm{deg}]$
Roll φ	EKF Standalone	-0.149	0.193
	EKF+MC-LAMBDA w/o Fix	0.013	0.162
	EKF+MC-LAMBDA Fixed	0.010	0.104
Pitch θ	EKF Standalone	-0.422	0.201
	EKF+MC-LAMBDA w/o Fix	-0.016	0.160
	EKF+MC-LAMBDA Fixed	-0.018	0.094
Yaw ψ	EKF Standalone	-0.023	0.186
	EKF+MC-LAMBDA w/o Fix	-0.005	0.108
	EKF+MC-LAMBDA Fixed	-0.006	0.071

to the static case, as expected. In this case, the standard deviation of each of the Euler angles corresponding to the output of the EKF, without fixing the ambiguities, is of around $0.2°$. As in the static case, the fixed solution is the more accurate out of the three solutions, which can be concluded from Table 6.4 and the close up of the Euler angles' estimation error on the right of Figure 6.5. The improvement of the attitude estimate when fixing the integer ambiguities, as opposed to simply taking the output of the

MC-LAMBDA method, is again apparent.

The estimation results of the gyroscope bias are shown in Figure 6.6. Even when subject to angular velocities of up to $15\,\mathrm{deg/s}$, the high accuracy of the attitude solution allows the correct estimation of the bias. This said, the oscillatory pattern that is most noticeable on the X-axis gyro bias estimate of the *EKF Standalone* solution, in the zoomed plot, is clearly due to the triangular shape of the simulated angular velocity. It is therefore concluded that high angular velocities can worsen the accuracy of the gyroscope bias estimate. Overall, both the float and fixed estimates represented in the plots follow the true simulated gyro bias.

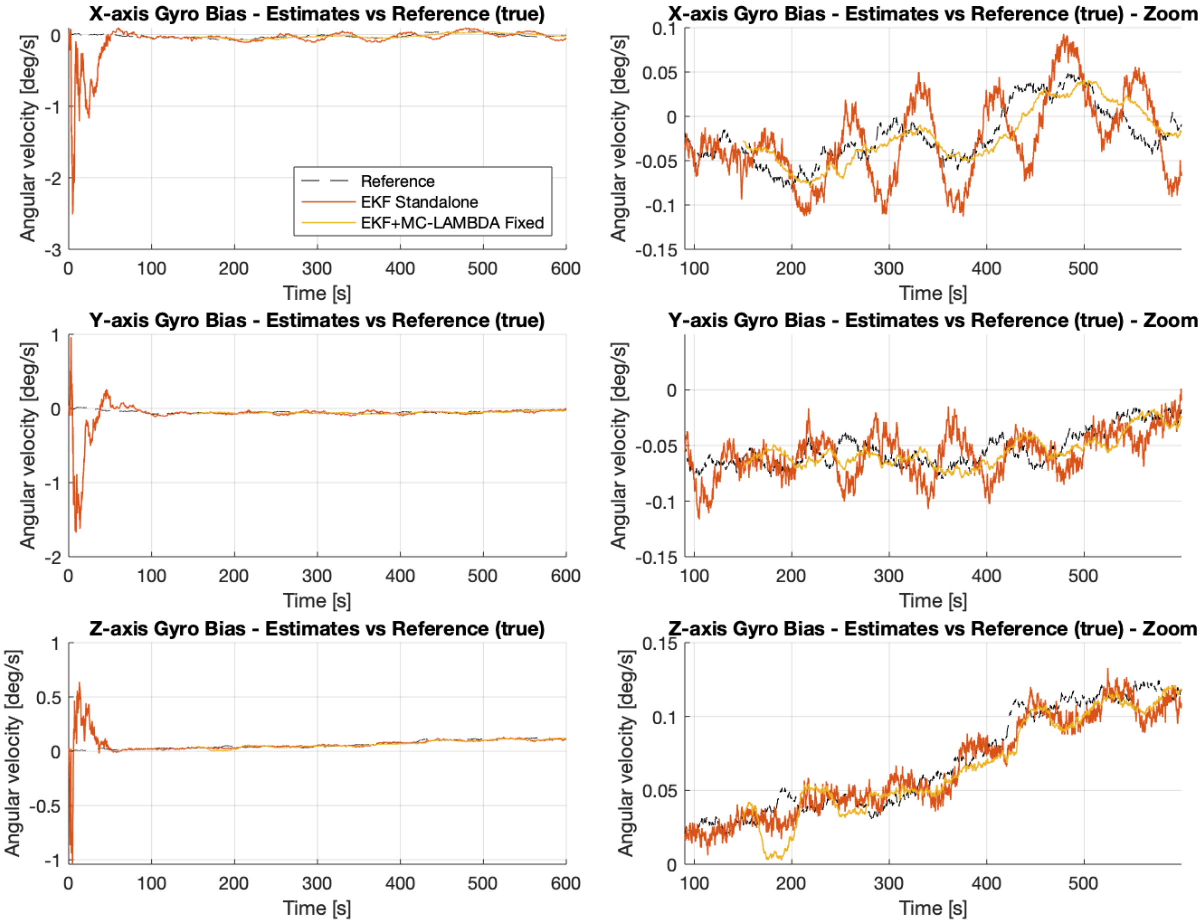


Figure 6.6: Dynamic trial, multiple baseline EKF without fixing and fixed solutions' bias estimates.

Single Baseline Observability

The results shown and analyzed in this Section correspond to the state estimates obtained using a single baseline. The experimental trials discussed later were also performed using only two GPS antennas, and so these simulation results are expected to provide useful insights about what to expect of the experimental results. The resulting attitude estimates and related metrics are presented in Figure 6.7 and Table 6.5.

The most important result to remark is the successful estimation of the full attitude using a single baseline, as seen in the Figure's plots. Even if the convergence of the *EKF Standalone* solution is slower

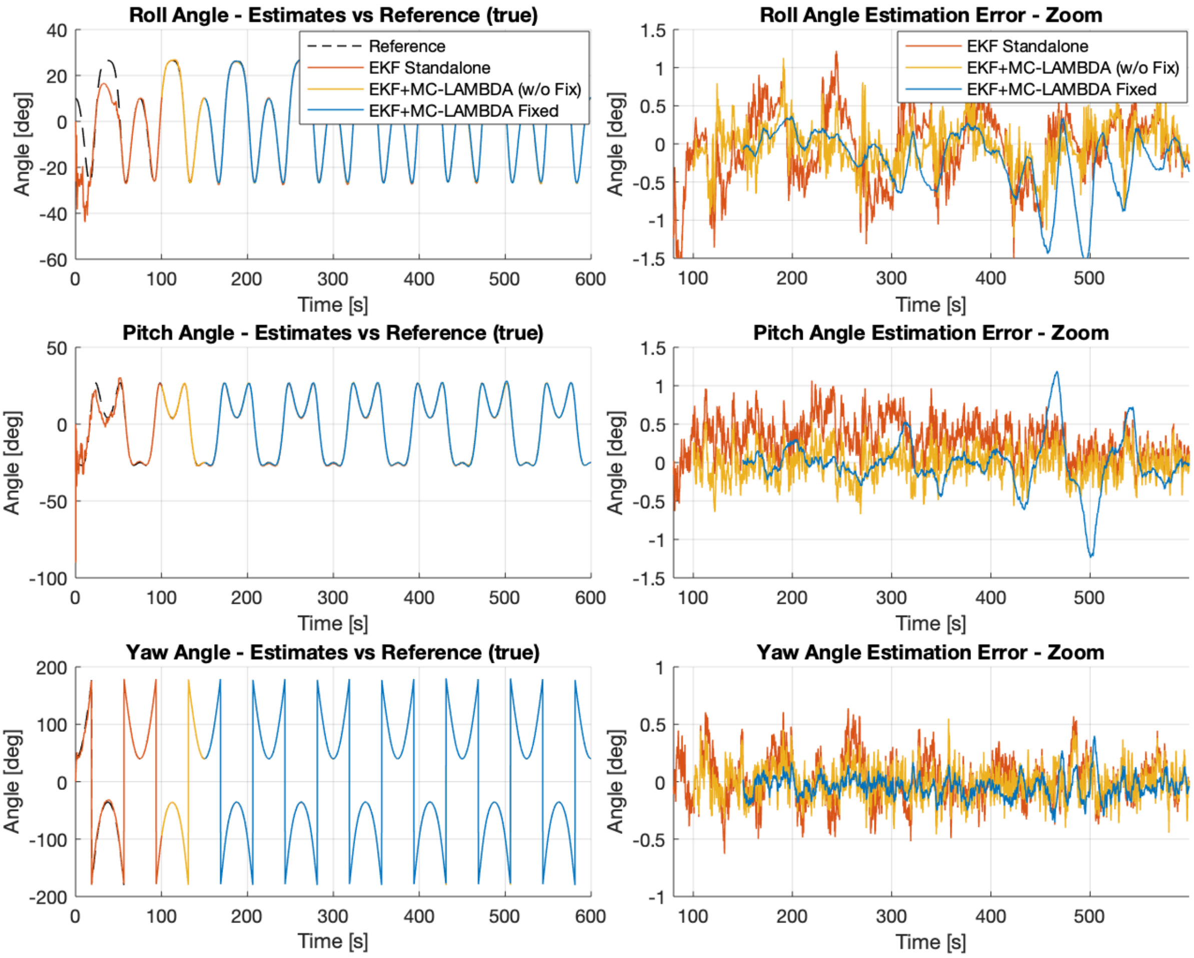


Figure 6.7: Dynamic trial, single baseline attitude estimates, including intermediate solutions.

Table 6.5: Performance metrics of the estimated attitude solutions for the single baseline dynamic trial, evaluated for $t > t_{fix}$.

Euler Angle	Solution	Error Mean $[\text{deg}]$	Error Std. $[\text{deg}]$
Roll φ	EKF Standalone	0.141	0.447
	EKF+MC-LAMBDA w/o Fix	-0.023	0.289
	EKF+MC-LAMBDA Fixed	-0.220	0.404
Pitch θ	EKF Standalone	0.309	0.254
	EKF+MC-LAMBDA w/o Fix	-0.016	0.195
	EKF+MC-LAMBDA Fixed	-0.015	0.336
Yaw ψ	EKF Standalone	0.007	0.193
	EKF+MC-LAMBDA w/o Fix	-0.010	0.141
	EKF+MC-LAMBDA Fixed	-0.044	0.095

when compared to the multiple baseline case since less information is available, after convergence the filter is able to reach sub degree accuracy.

These results further validate the developed quaternion conditional solution, derived in Section 4.5.2. Recall that the *EKF+MC-LAMBDA (w/o Fix)* attitude solution corresponds to this conditional solution, and that it is also used to compute the fixed solution. The original conditional solution based on the LS approach, described in section 4.1, requires having at least three baselines to fully determine the attitude. However, the deduced expressions allow the approximate computation of the attitude conditional solution using the EKF float solution and for any number of baselines, therefore taking advantage of the characteristics of the recursive filter. This quaternion conditional solution is what allows using the MC-LAMBDA method for integer ambiguity resolution in a single baseline case.

It is seen on the zoomed plots of the attitude estimation error on the right of Figure 6.7, as well as in the corresponding Table, that the roll and pitch fixed solutions are the worse estimates. The opposite was true in the multiple baseline results shown previously. The error in the fixed solution is explained by analyzing the gyro bias estimates, shown in Figure 6.8. The fixing process lowers the covariance matrix entries of the filter, since it consists of a 2nd Update Step. As such, the filter becomes less sensitive to the angular velocity measurements. Since in a single baseline scenario the GNSS observations are fewer in number, the gyro bias estimate are worsened, except for the bias relative to the Z-axis since it is the axis around which the body rotates, exciting the filter. Due to the error of the gyro bias in the X and Y axis, the output of the EKF propagation step is less accurate, therefore affecting the final updated solution. As the body rotation becomes more diverse, it is expected that the fixed gyro bias estimates improve.

Apart from the errors in the fixed solution, it is important to note that, in the *EKF Standalone* solution, the gyro bias is correctly estimated, similarly to the body attitude. These simulation results suggest that even using a single baseline, if the motion of the body is *sufficiently* varied, the full attitude and gyro bias are observable. It is also seen by trial and error that as the body's angular velocity is higher, the estimates become more accurate.

6.2 Experimental Results

Using the hardware and experimental setup presented in the previous Chapter, static and dynamic trials were performed.

The collected measurements are substantially different from the simulated data since simulating more realistic observations would involve very complex models, which are outside of the scope of this thesis. Due to this, the algorithm parameters to tune were assigned different values when processing the experimental data, discussed in Section 6.2.1.

In the Section that follows, a static trial and the corresponding results are briefly presented and discussed. A static trial using only two single-frequency receivers is the most demanding scenario which can be tested. Considering also the type of gyroscope at hand, the resulting attitude estimates are not optimal.

Lastly, the results of a dynamic trial are presented in Section 6.2.3. Using the collected data from the experiment, the performance of the proposed solution in a real-life scenario is evaluated. Furthermore, this trial is meant to replicate the observability conditions shown in the dynamic, single baseline simulated trial.

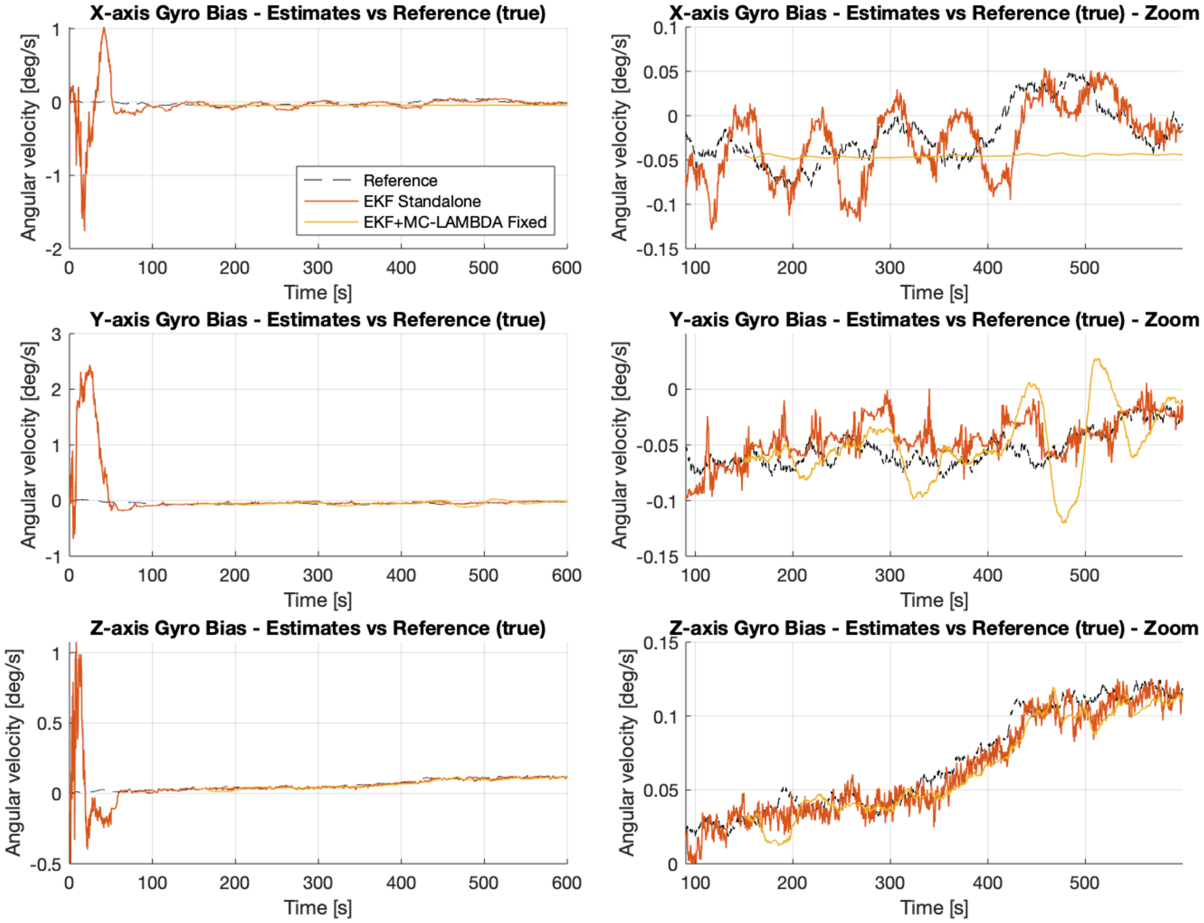


Figure 6.8: Dynamic trial, single baseline EKF without fixing and fixed solutions' bias estimates.

6.2.1 Algorithm Parameters

The tuning of the filter parameters and overall flow of the algorithm using experimental data resulted in different values from those used in the processing of simulated data. The experimental data contains errors not modelled directly in the simulation, meaning that the parameters used in one scenario might not be optimal in the other.

First off, in both experimental trials, the used baseline has a baseline length of $1.95\,\mathrm{m}$. In the body frame, the direction of the baseline is set along the positive X axis. This information is used in order to define the initial attitude of the body in relation to the ENU frame, and to be able to interpret the attitude estimate results. As shown previously in Figure 5.4c, the gyroscope is then set along the positive X axis of the body frame, facing up.

Concerning the filter parameters, the $\boldsymbol{Q}$ and $\boldsymbol{R}_k$ matrices used are the same as the ones defined in Section 6.1.1. The $\boldsymbol{R}_k$ is also set to change at $t = t_{\boldsymbol{R}_{chg}}$. However, in the experimental trials a value of $t_{\boldsymbol{R}_{chg}} = 100\,\mathrm{s}$ is used, allowing more time for the initial convergence of the filter.

The parameters defining the starting times of the MC-LAMBDA method and the Solution Fixing are set differently for the static and dynamic trial. As there is lower filter excitation in the static trial, more time is given for the converge of the float solution, as shown in the results that follow. In this case, it is set $t_{mc} = 550\,\mathrm{s}$ and $t_{fix} = 600\,\mathrm{s}$, corresponding to a $50\,\mathrm{s}$ interval of storing candidate sets of integer ambiguities.

In the processing of the dynamic trial, these times are set at $t_{mc} = 200\,\text{s}$ and $t_{fix} = 240\,\text{s}$, where the time window for storing ambiguities is smaller. These parameters are very different between static and dynamic trial since, again, when only one baseline is used, the information given by the GNSS observations in the static case is less varied. In any case, for real life applications, the times set for the static trial would obviously also give good results in a dynamic scenario.

Finally, note that the Reach Module receivers were set to provide GNSS observables at a rate of $5\,\text{Hz}$, while the angular velocity measurements are sampled at a frequency of $20\,\text{Hz}$.

6.2.2 Static Trial

In the following, a brief description of the setup of the trial is given, and the corresponding estimated attitude is presented and discussed.

Trial Description

The hardware and setup used for this trial was described in Section 5.3. The collected data corresponds to the GPS observables and navigation message beginning at UTC time $18:52:00$ of $2020/09/16$. The approximate location of the antennas is given by a latitude of $\phi = 40.7221\,°$ and longitude $\lambda = -7.7264\,°$. Taking into account this epoch and location, note that only observations from satellites whose elevation angle is higher than $15\,°$ are used in the processing.

Knowing the approximate attitude of the body is crucial in order to interpret the results. Referencing the setup baseline orientation with a map of the site, it is seen that the yaw angle of the body is approximately equal to $30\,°$. As seen in Figure 5.4a, both the roll and pitch angles should be very close to $0\,°$ since the ground surface is flat. Lastly, the measured angular velocity which characterizes the trial is shown in Figure 6.9. While the measurements' mean value is close to $0\,\text{deg/s}$, the low quality of the smartphone's sensors is evident by the magnitude of the noise.

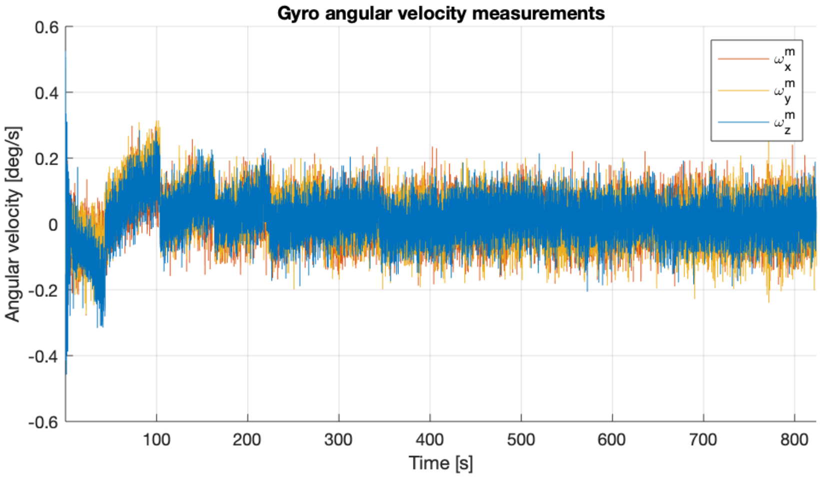


Figure 6.9: Measured angular velocity during the experimental static trial.

Results

The attitude estimation results obtained by processing the GPS observations and gyroscope measurements collected during the static trial, using the parameters described previously, are given in Figure 6.10. Since the float, conditional and fixed solutions have already been compared in detail in the simulation results, from here on only the fixed solution is shown. Note that as the fixing of the solution occurs only at $t = t_{fix} = 600\,\mathrm{s}$, the estimates shown before this epoch correspond to the EKF float solution, in order to evaluate the evolution of the filter. The same is true for all results that follow.

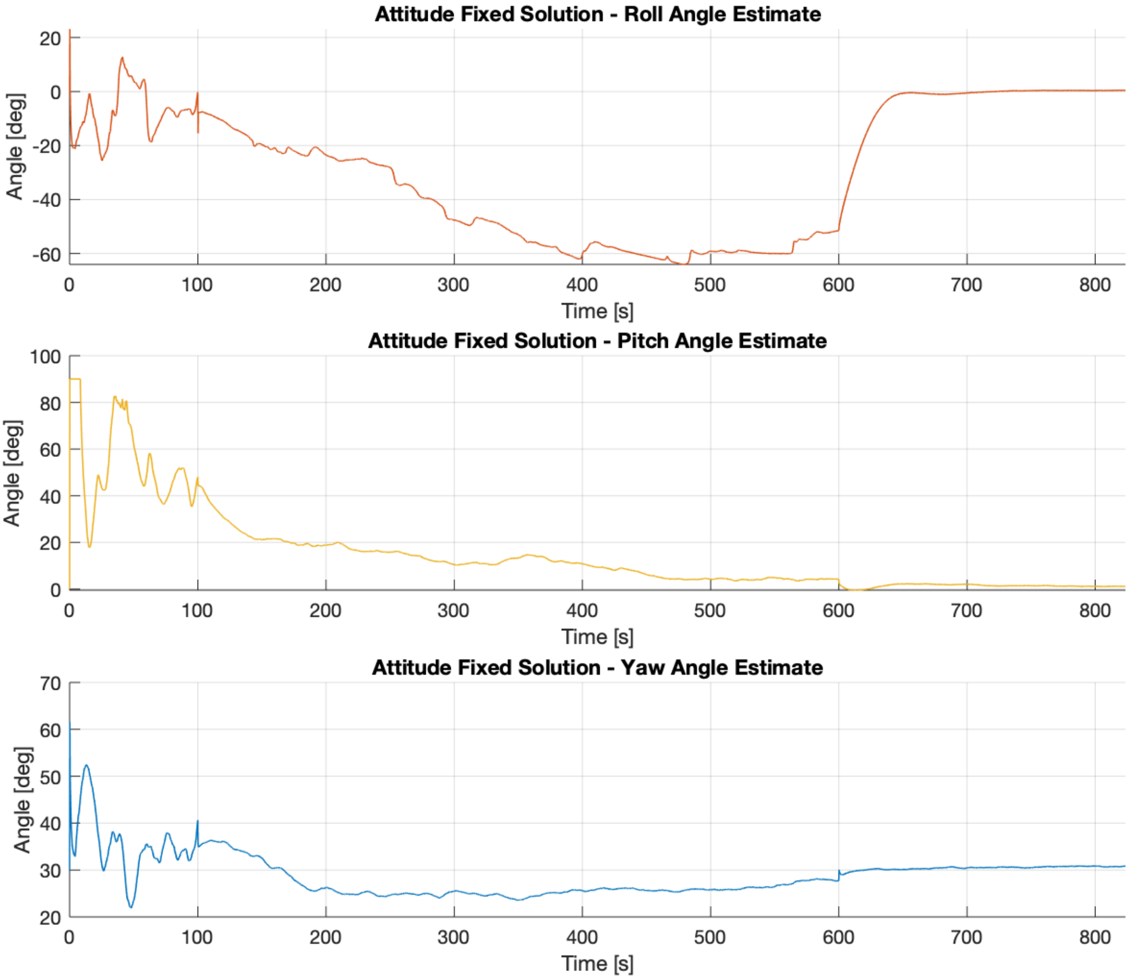


Figure 6.10: Fixed solutions' estimated Euler angles during the static experiment.

When comparing the evolution of the Euler angles in this Figure with the simulation results, it is clear that the convergence rate of the filter is much slower, only stabilizing after around $500\,\mathrm{s}$. At $t = t_{\boldsymbol{R}_{chg}} = 100\,\mathrm{s}$, the solution becomes more stable as the filter's bandwidth relative to the carrier-phase measurements is increased. Overall, the pitch and yaw angles converge to the approximate reference angles of $0\,°$ and $30\,°$, respectively. The roll angle is not observable, since up to $t = 600\,\mathrm{s}$ its estimate diverges to around $-60\,°$, which is clearly incorrect. This result is expected, since it was shown using a linearized model that in a static, single baseline scenario, the full attitude is not observable. The one degree of freedom corresponds to a rotation around the baseline, which in this case coincides exactly with the roll angle, due to having defined the baseline in the body frame along the X axis.

After fixing occurs, the small error in the pitch and yaw estimates is corrected and the estimates become less noisy since the integer ambiguities are known. In fact, it is seen that, after fixing, the pitch and yaw estimates standard deviations are of $0.65°$ and $0.36°$, respectively. Although the mean error can't be known precisely, these estimates are clearly close to the true values and their standard deviations rival those shown in the simulation results.

The slow convergence of the estimates is also related to the low number of observed satellites over time, which is shown in the top plot of Figure 6.11. During the whole experiment, a maximum of 7 GPS satellites are used to compute the float solution. If signals from other constellations were also processed, the convergence rate and accuracy of the solution would surely improve. The same would happen if more baselines were available.

The bottom plot in Figure 6.11 represents the epochs in which there occurred a change in the reference satellite selected by the algorithm. As shown , at around $t = 500\,\text{s}$, there is a change. Since there isn't any discontinuity in the Euler angles estimates at this epoch, the expressions used to deal with a change in reference satellite, deduced in Appendix D, are correctly implemented.

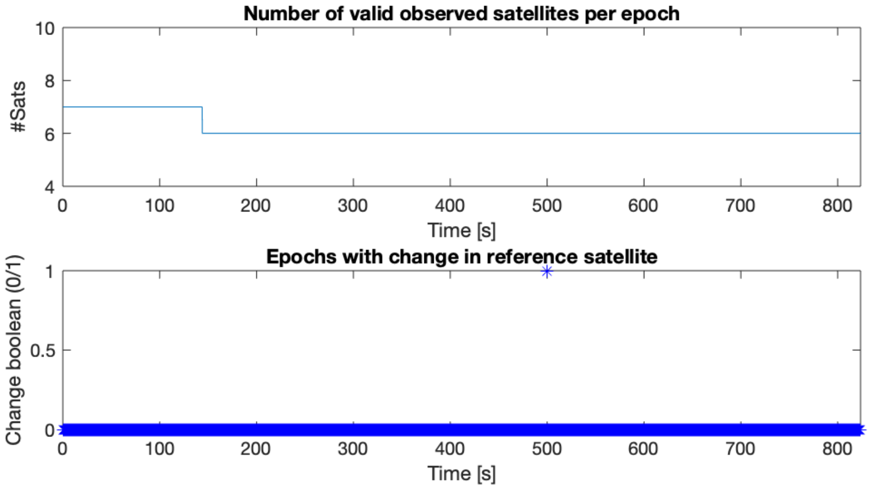


Figure 6.11: Changes in the GPS satellites in view during the static experiment.

Lastly, the estimated gyroscope bias are plotted in Figure 6.12. Similarly to the attitude estimates, the X-axis gyroscope bias appears not to be observable due to having a single baseline placed along the X-axis of the body frame. Note that the sharp drop in the X-axis bias estimate after fixing occurs is what makes the roll estimate in Figure 6.10 converge to a value close to $0°$. This once again shows that in this trial, the roll estimate is not directly influenced by the GNSS observations, due to the baseline configuration.

In the next Section, the results obtained using the proposed solution to process data collected during a dynamic real-life scenario are presented.

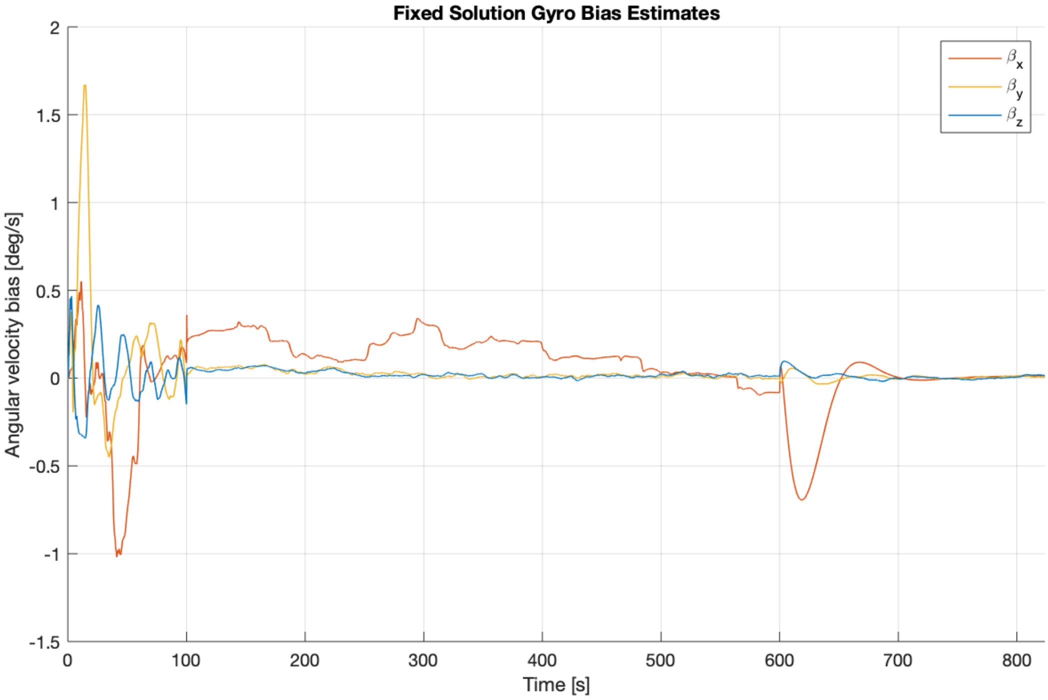


Figure 6.12: Fixed solutions' estimated gyroscope bias during the static experiment.

6.2.3 Dynamic Trial

The trajectory done during the dynamic experiment is described next, along with the body frame's angular motion. The maneuvers and sections of the experiment where the baseline's attitude is manually changed are highlighted. Lastly, the attitude estimates output by the algorithm are presented.

Trial Description

The dynamic trial was performed by placing the GPS antennas and gyroscope on top of a moving car, as was shown in Figure 5.4b. The experiment takes place at the same date and location as the static trial discussed previously. The baseline length is set at $1.95\,\mathrm{m}$ and the initial yaw angle in this case is approximately equal to $-60\,°$, derived from the starting position of the car seen in Figure 5.4b. Note that the baseline, aligned with the body frame's X axis, is placed perpendicular to the car.

In order to have a broad reference of what to expect from the processing results, the trajectory of the platform during the experiment is represented in Figure 6.13. By driving the car along this trajectory, only the yaw angle can be changed easily. Thus, to increase the excitation of the filter and introduce more varied angular motion to the experiment, the baseline is manually rotated around its axis at some epochs. These epochs are marked in the trajectory by a star close to the location where this happens. The experiments' start and finish points are highlighted using a red and yellow rectangle, respectively.

Lastly, the measured angular velocity matching this trajectory is shown in Figure 6.14. Note the noisy measurements when the car is initially traveling in a straight line, due to its vibration and the rough terrain. After around $520\,\mathrm{s}$, the car is stopped and the engine is turned off, transitioning into a completely static scenario.

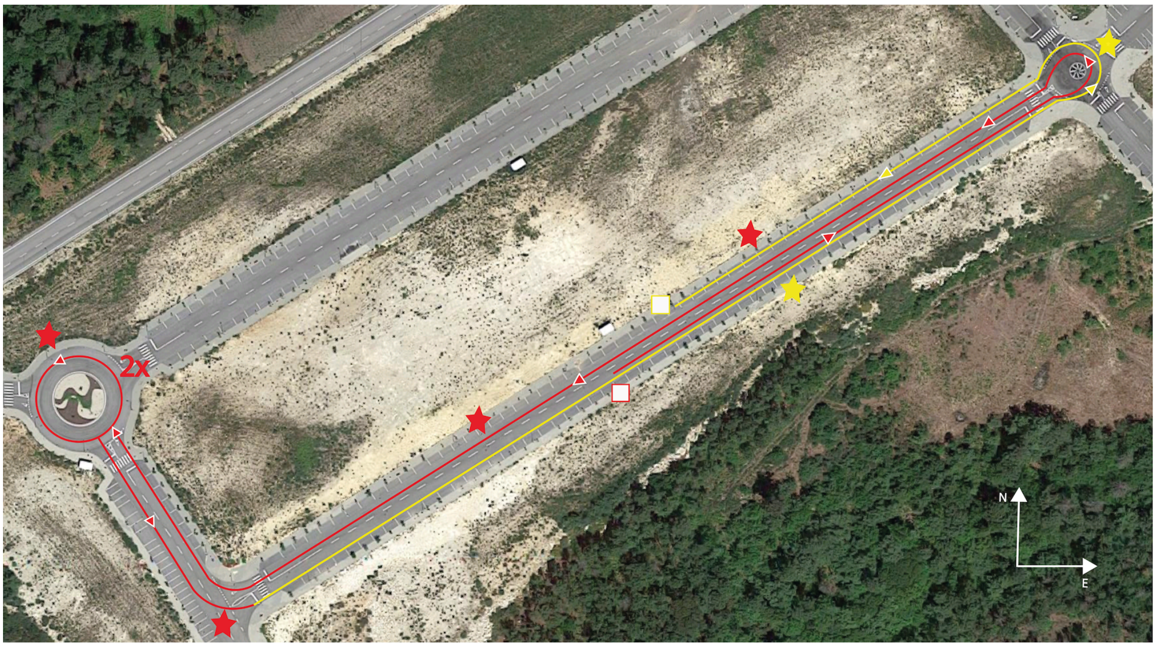


Figure 6.13: Trajectory and key points describing the experimental dynamic trial.

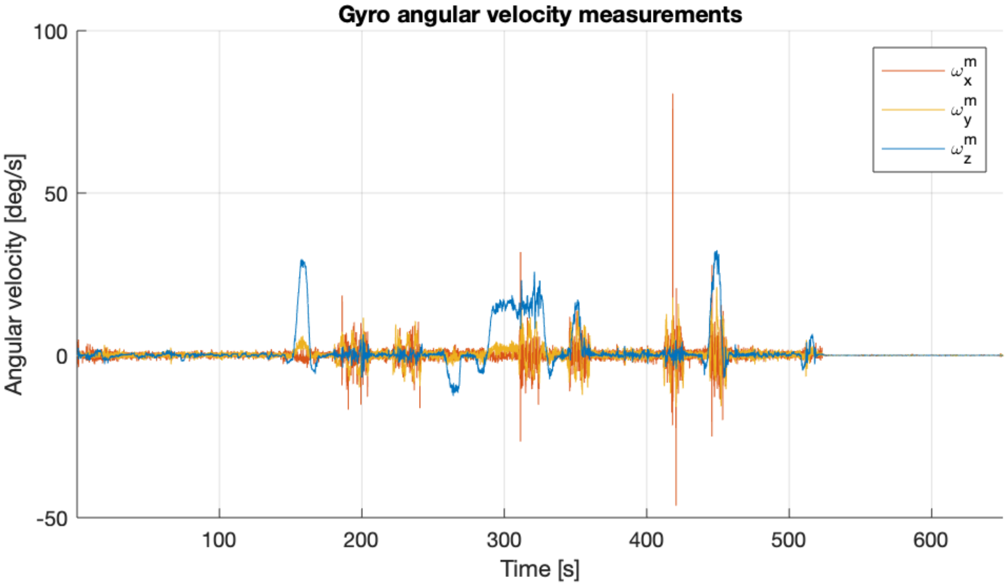


Figure 6.14: Measured angular velocity during the experimental dynamic trial.

Results

The proposed solution should be able to clearly show every turn and nudge of the baseline represented in the trajectory of Figure 6.13. The Euler angles corresponding to the fixed solution are shown in Figure 6.15.

After $t = t_{fix} = 100\,\mathrm{s}$, the filter's estimate starts converging to the correct attitude, as the yaw angle estimate approaches $-60\,°$. Note that the convergence rate in this case is much quicker when compared to the static trial due to the angular motion. After convergence, each turn and roundabout done during the trajectory of the vehicle is clearly seen in the $90\,°$ and $180\,°$ variations of the estimated heading. Concerning the pitch estimate, it is close to $0\,°$, as expected, since the terrain's inclination is

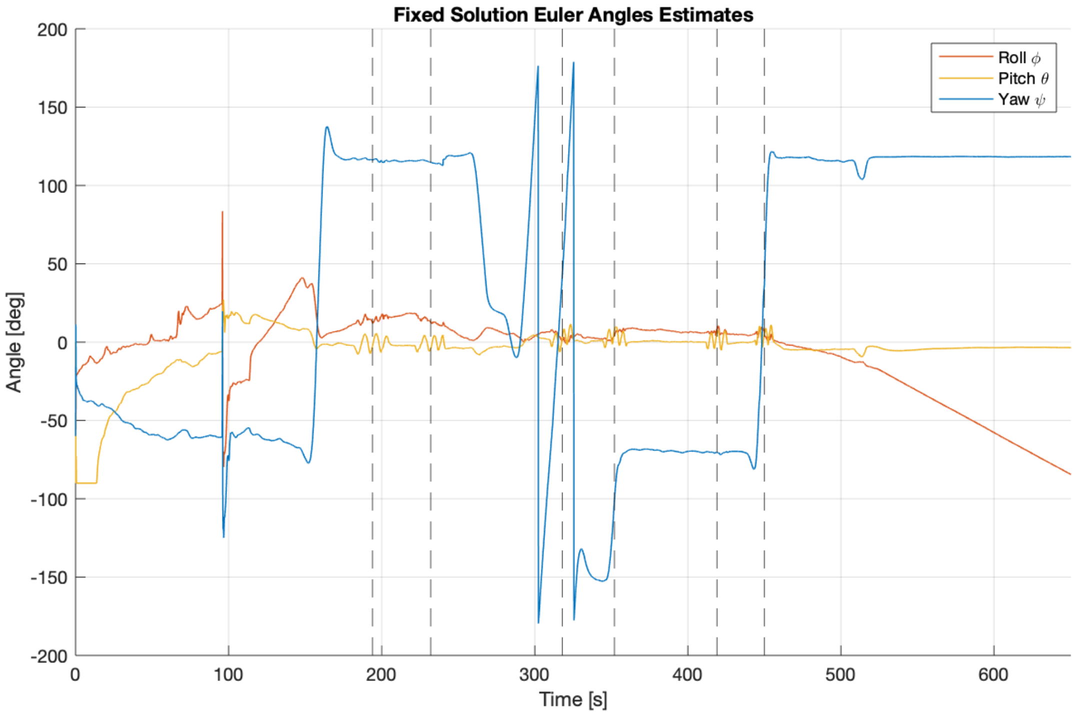


Figure 6.15: Fixed solutions' estimated Euler angles during the dynamic experiment.

very low. The pitch oscillations marked by the horizontal dashed lines, between the epochs at around $200\,\mathrm{s}$ and $450\,\mathrm{s}$, correspond exactly to the manual rotations of the baseline marked with star symbols in the Figure of the trajectory. Overall, the pitch and yaw angles are estimated correctly, as expected, since the baseline coincides with the X-axis of the body frame.

Looking at the roll angle estimate, it can be concluded that the additional angular motion of the body causes the estimate not to diverge during the dynamic part of the experiment. Since the baseline is rotated around both the Y and Z axis, the filter is able to keep the roll estimate close to $0\,°$, although there is some error. In fact, after the body is static, at around $t = 520\,\mathrm{s}$, the roll estimate starts to diverge, as expected.

The gyroscope bias estimates in Figure 6.16 can be analyzed is a similar way. During the dynamic part of the trial, the filter has some difficulty with precisely estimating the bias, as was the case during the simulated dynamic trials. Even more so since, as seen in Figure 6.14, the angular velocities's magnitude during the experiment can reach around $30\,\mathrm{deg/s}$. Once again, since a single baseline is used, the X-axis bias estimate is not correct, being responsible for the deviation of the roll angle estimate when the body is not rotating. This can be seen clearly after the vehicle comes to a full stop at the end of the experiment. In this situation, the gyro bias is not observable and its integration during the propagation step of the filter causes the divergence of the roll angle.

In a concluding note, it is seen that in a dynamic scenario the roll angle does not diverge as markedly as in the static case. Also, the extra excitation of the filter causes the convergence of the estimates to be quicker. Comparing the Euler angles pitch and yaw results to the vehicle trajectory, one can conclude that the proposed solution correctly estimates these angles. In order to assess the performance quantitatively, a trustworthy attitude reference would be needed, which is left for a future work.

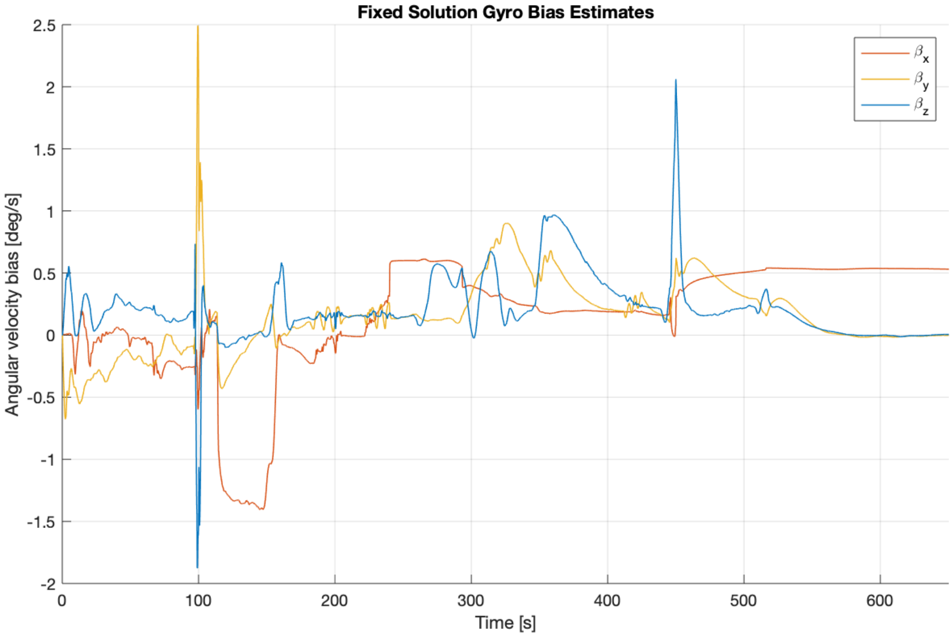


Figure 6.16: Fixed solutions' estimated gyroscope bias during the dynamic experiment.

Chapter 7

Conclusions

Throughout this document, a novel solution for GNSS-based attitude estimation was detailed. In this formulation, the float solution is obtained recursively using an EKF fed by angular velocity and carrier phase measurements, which is then integrated with the effective MC-LAMBDA method. All steps needed for reproduction of this solution were unambiguously described, as well as the reasoning behind it.

While the experimental validation of the proposed solution falls short of evaluating its performance using multiple baselines, as well as presenting error metrics, the results clearly shown that the pitch and yaw angles of the vehicle are estimated in a real-life scenario. Quantitative evaluation of the performance of this solution was done using simulated data, which allowed drawing conclusions about the success rate of the integer ambiguities determination and the attitude estimation accuracy.

In the following, the accomplishments of the this research are enumerated. While the proposed solution provides some insight into new ways of exploring the GNSS observations and existing solutions for attitude estimation, a complete study obviously could not have been done in this short time span. Therefore, some aspects of this solution with room for improvement, as well as future work and analysis to be done on this project are discussed next.

7.1 Contributions

Overall, the objectives that this thesis sets out to accomplish have been met. The integration of the gyroscope measurements in the kinematics model underlying the EKF allowed a precise estimation of the attitude. Namely, concerning the float solution, precisions of $0.76°$, 0.50° and 0.40° for the roll, pitch and yaw angles, respectively, were achieved in the most demanding static scenario.

Furthermore, preliminary conclusions about the observability of the filter concerning the attitude and gyro bias estimation were achieved by analyzing the observability properties of a linearized system model. It was concluded analytically that in a single baseline scenario, if the body is static, there is not enough information to estimate the full attitude and 3-axis gyro bias. Doing the same analysis for the case of multiple baselines, the state is found to be observable. Then, by simulating a dynamic scenario using only two antennas, it was seen how it is likely possible for the system to be observable under certain excitation conditions. However, these must be found analytically in a future research.

Using the developed integer ambiguity determination framework, the overall solution made use of the precise carrier phase measurements. To do so, a way of linking the filter's attitude float solution to the quaternion condicional solution given the correct set of integer ambiguities was derived. It was seen how the use of these two methods implies success rates of at least 83.96%, even in a single baseline scenario when only 5 GPS satellites are used and for a simulated code noise and baseline length of $50\,\mathrm{cm}$ and $1\,\mathrm{m}$, respectively.

The proposed solution was validated using both simulated and experimental data. Both types of trials allowed to conclude that the proposed method of computing the fixed solution further increases the accuracy of the float and conditional solutions. Using simulated data, it was shown how the precision of the fixed solution can reach around $0.1\,°$ in a dynamic scenario, being slightly lower if the body is still. Lastly, the very simple experimental setup used suggests that this navigation framework is suitable for a multitude of real-life applications such as medium-sized UAVs or autonomous vehicles, usually already equipped with GNSS receivers and an IMU.

7.2 Future Work

The attitude estimation solution that was presented still has large room for improvement and further validation. Applications where a single baseline is used are of great interest, and require a more complete study of the observability properties of the designed filter which is part of the solution.

Further experimental validation should be done, using more GNSS antennas and a better performing gyroscope. Apart from testing the algorithm in post-processing, it is crucial that the algorithm may also be used in real-time, since this is the case for the usual attitude estimation systems. To do so, in the future the software must be slightly adapted to receive measurements over time. More importantly, its time complexity needs to be lowered depending on the sampling frequency of the equipment. Specifically, making the software more time efficient might involve choosing a tighter bound C_1 in (4.70) or an alternative way of solving the nonlinear minimization problem in (4.84).

Prior to fixing the integer ambiguities, it was proposed that there be a time period where sets of ambiguities are stored. Other integer ambiguity determination methods try to validate the set of integer ambiguities determined at each epoch using statistical tests based on approximate probability density functions. In a future study, the same approach could be done for the MC-LAMBDA method, providing a measure of how likely it is that a set of integer ambiguities is correct. Furthermore, this work does not go into detail on the mechanisms to detect and deal with disturbances such as cycle slips or loss of lock. Research on these methods is crucial before using this solution in real-life applications.

Finally, the techniques applied to improve the precision of precise positioning can often be used in this context as well. In the future, it could be studied how the integration of precise ephemeris, clock corrections and others provided by fixed base stations affect the quality of the attitude estimates. Expanding the solution to include observations from other constellations apart from the GPS would no doubt also prove very interesting.

Bibliography

[1] W. F. Blanchard. Air navigation systems chapter 4. hyperbolic airborne radio navigation aids – a navigator's view of their history and development. *Journal of Navigation*, 44(3):285–315, 1991. doi: 10.1017/S0373463300010092.

[2] B. W. Parkinson, T. Stansell, R. Beard, and K. Gromov. A history of satellite navigation. *Navigation*, 42(1):109–164. doi: https://doi.org/10.1002/j.2161-4296.1995.tb02333.x.

[3] G. Strang and K. Borre. *Linear Algebra, Geodesy, and GPS*. Wellesley-Cambridge Press, 1997. ISBN 9780961408862. URL `https://books.google.pt/books?id=MjNwWUY8jx4C`.

[4] M. L. Psiaki. Batch algorithm for global-positioning-system attitude determination and integer ambiguity resolution. *Journal of Guidance, Control, and Dynamics*, 29(5):1070–1079, 2006. doi: 10.2514/1.18351.

[5] M. Campo-Cossio, A. Puras, R. Arnau, and D. Bolado. Real-time attitude determination system based on GPS carrier phase measurements and aided by low-cost inertial sensors for high dynamic applications, 2009.

[6] C. Eling, L. Klingbeil, and H. Kuhlmann. Real-time single-frequency GPS/MEMS-IMU attitude determination of lightweight UAVs. *Sensors*, 15:26212–26235, 10 2015. doi: 10.3390/s151026212.

[7] D. Medina, V. Centrone, R. Ziebold, and J. García. Attitude determination via GNSS carrier phase and inertial aiding. In *ION GNSS+ 2019*, September 2019. URL `https://elib.dlr.de/129046/`.

[8] G. Giorgi and P. Teunissen. *GNSS Carrier Phase-Based Attitude Determination*. February 2012. ISBN 978-953-51-0150-5. doi: 10.5772/38381.

[9] P. Axelrad and L. M. Ward. Spacecraft attitude estimation using the global positioning system - methodology and results for RADCAL. *Journal of Guidance, Control, and Dynamics*, 19(6):1201–1209, 1996. doi: 10.2514/3.21772.

[10] D. Willi and M. Rothacher. GNSS attitude determination with non-synchronized receivers and short baselines onboard a spacecraft. *GPS Solutions*, 21(4):1605–1617, 2017. ISSN 1521-1886. doi: 10.1007/s10291-017-0639-0. URL `https://doi.org/10.1007/s10291-017-0639-0`.

[11] G. Giorgi, P. Teunissen, and T. Gourlay. Instantaneous global navigation satellite system (gnss)-based attitude determination for maritime applications. *Oceanic Engineering, IEEE Journal of*, 37: 348–362, 07 2012. doi: 10.1109/JOE.2012.2191996.

[12] P. Teunissen. The least-squares ambiguity decorrelation adjustment: A method for fast GPS integer ambiguity estimation. *Journal of Geodesy*, 70:65–82, 11 1995. doi: 10.1007/BF00863419.

[13] P. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31 (1):1–10, 1966. doi: 10.1007/BF02289451.

[14] M. Shuster. Approximate algorithms for fast optimal attitude computation. 02 1978. doi: 10.2514/6.1978-1249.

[15] L. Markley. Attitude determination from vector observations: A fast optimal matrix algorithm. *Journal of the Astronautical Sciences*, 41, 07 1993.

[16] P. Teunissen. A general multivariate formulation of the multi-antenna GNSS attitude determination problem. *Artificial Satellites*, 42:97–111, 01 2007. doi: 10.2478/v10018-008-0002-3.

[17] C. Park and P. Teunissen. A baseline constrained LAMBDA method for an integer ambiguity resolution of GNSS attitude determination systems. *Journal of Institute of Control, Robotics and Systems*, 14:587–594, 06 2008. doi: 10.5302/J.ICROS.2008.14.6.587.

[18] W. Bing, S. Lifen, X. Guorui, D. Yu, and Q. Guobin. Comparison of attitude determination approaches using multiple global positioning system (GPS) antennas. *Geodesy and Geodynamics*, 4(1):16 – 22, 2013. ISSN 1674-9847. doi: https://doi.org/10.3724/SP.J.1246.2013.01016. URL `http://www.sciencedirect.com/science/article/pii/S1674984715300574`.

[19] J. Wertz. *Spacecraft Attitude Determination and Control*. Astrophysics and Space Science Library. Springer Netherlands, 1978. ISBN 9789027709592. URL `https://books.google.pt/books?id=GtzzpUN8VEoC`.

[20] J. Solà. Quaternion kinematics for the error-state kalman filter. *CoRR*, abs/1711.02508, 2017. URL `http://arxiv.org/abs/1711.02508`.

[21] Image taken from this source. URL `https://commons.wikimedia.org/wiki/File:ECEF_ENU_Longitude_Latitude_right-hand-rule.svg`.

[22] GPS Enterprise Space & Missile Systems Center (SMC) - LAAFB. NAVSTAR GPS Space Segment/User Segment L1C Interfaces, May 2020.

[23] N. National Coordination Office for Space-Based Positioning and Timing. Official U.S. government information about the Global Positioning System (GPS) and related topics, Space Segment. URL `https://www.gps.gov/systems/gps/space/`.

[24] D. Doberstein. *Fundamentals of GPS receivers: A hardware approach*. 01 2012. ISBN 978-1-4614-0408-8. doi: 10.1007/978-1-4614-0409-5.

[25] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins. *Global Positioning System: Theory and Practice*. Springer-Verlag, 1997. ISBN 9783211828397. URL `https://books.google.pt/books?id=tIZTAAAAMAAJ`.

[26] B. Parkinson and J. Spilker. *Global Positioning System: Theory and Applications*. Number vol. 1 in Ciencia militar y naval. American Institute of Aeronautics & Astronautics, 1996. ISBN 9781600864193. URL `https://books.google.pt/books?id=lvI1a5J_4ewC`.

[27] C. Goad and L. Goodman. A modified Hopfield tropospheric correction model. Presented at the American Geophysical Union Fall Annual Meeting, San Francisco, CA, 1974.

[28] A. Kleusberg and P. Teunissen. *GPS for Geodesy*. GPS for Geodesy. Springer, 1996. ISBN 9783540607854. URL `https://books.google.pt/books?id=yg1NAQAAIAAJ`.

[29] G. Giorgi. *GNSS Carrier Phase-based Attitude Determination. Estimation and Applications*. PhD thesis, Delft University of Technology, 2011.

[30] J. L. Farrell, J. Stuelpnagel, R. H. Wessner, J. R. Velman, and J. Brook. A least squares estimate of satellite attitude (Grace Wahba). *Siam Review*, 8:384–386, 1966.

[31] L. Markley and D. Mortari. Quaternion attitude estimation using vector observations. *Journal of the Astronautical Sciences*, 48, 04 2000. doi: 10.1007/BF03546284.

[32] S. A. Whitmore and L. Hughes. United States Patent - Closed-form Integrator for the Quaternion (Euler Angle) Kinematics Equations, 2000.

[33] F. L. Markley and J. L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. Published jointly by Microcosm Press and Springer, 2014.

[34] A. Gelb and T. A. S. Corporation. *Applied Optimal Estimation*. The MIT Press, 1974. ISBN 0262570483.

[35] Y. Song and J. W. Grizzle. The extended kalman filter as a local asymptotic observer for nonlinear discrete-time systems. In *1992 American Control Conference*, pages 3365–3369, 1992. doi: 10.23919/ACC.1992.4792775.

[36] R. Hermann and A. Krener. Nonlinear controllability and observability. *IEEE Transactions on Automatic Control*, 22(5):728–740, 1977. doi: 10.1109/TAC.1977.1101601.

[37] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis. Analysis and improvement of the consistency of extended kalman filter based slam. In *2008 IEEE International Conference on Robotics and Automation*, pages 473–479, 2008. doi: 10.1109/ROBOT.2008.4543252.

[38] P. de Jonge and C. Tiberius. *The LAMBDA Method for Integer Ambiguity Estimation: Implementation Aspects*. Publications of the Delft Geodetic Computing Centre. Verlag der Delft Univers. of Technolog., 1996. URL `https://books.google.pt/books?id=-y4QvwEACAAJ`.

[39] International GNSS Service (IGS), RINEX Working Group and Radio Technical Commission for Maritime Services Special Committee 104 (RTCM-SC104). (2015). The Receiver Independent Exchange Format (Version 3.03) documentation. Available at:. URL `ftp://igs.org/pub/data/format/rinex303.pdf`.

[40] Technical drawing and specifications of the AT575-75 GPS antennas. Consulted at 5th December, 2020. Available at:, . URL `https://www.aeroantenna.com/PDF/AT575-75K.pdf`.

[41] Technical drawing and specifications of the 53G15A GPS antenna, manufactured by Antcom Corporation. Consulted at 5th December, 2020, . URL `https://manualzz.com/doc/34666472/active-l1-gps-antenna--p-n--3g15a--xx-x-c-also-know`.

[42] Specification of the Reach Module RTK GNSS receiver, manufactured by Emlid. Consulted at 5th December, 2020. URL `https://docs.emlid.com/reach/specs/`.

[43] LP Ellis. (2019). Sensor Lab (Version 1.3) [Mobile application software]. Retrieved from https://play.google.com/store.

[44] G. Hendeby and F. Gustafsson. On nonlinear transformations of gaussian distributions, January 2003.

Appendix A

Analytical validation of the system solution in the observability analysis

In Section 4.4.3, the solution of system (A.1) was given based on the physical description of the measurements.

$$\begin{bmatrix} \boldsymbol{\Upsilon} \boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12}) \Delta \boldsymbol{q}_k + \boldsymbol{0}_m \Delta \boldsymbol{n}_k \\ \boldsymbol{\Upsilon} \boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12}) \Delta \boldsymbol{q}_k + \lambda \boldsymbol{I}_m \Delta \boldsymbol{n}_k \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix} \tag{A.1}$$

In the following, it is confirmed analytically that a quaternion difference representing a rotation around $\boldsymbol{l}_{12}$ is a solution of this system of equations, as written in (A.2).

$$\begin{cases} \Delta \boldsymbol{q}_k = \tilde{\boldsymbol{q}}_k \circ \boldsymbol{q}\{\alpha \boldsymbol{l}_{12}\} - \tilde{\boldsymbol{q}}_k, \quad \forall \alpha \in \mathbb{R} \\ \Delta \boldsymbol{n}_k = \boldsymbol{0} \end{cases} \tag{A.2}$$

First, note that $\Delta \boldsymbol{n}_k = \boldsymbol{0}$ is a trivial solution. Furthermore, if $\boldsymbol{\Upsilon}$ is full rank, then the equation to solve becomes simply

$$\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12}) \Delta \boldsymbol{q}_k = \boldsymbol{0}. \tag{A.3}$$

Then, we wish to prove that $\Delta \boldsymbol{q}_k = \tilde{\boldsymbol{q}}_k \circ \boldsymbol{q}\{\alpha \boldsymbol{l}_{12}\} - \tilde{\boldsymbol{q}}_k, \quad \forall \alpha \in \mathbb{R}$ is a solution of this equation. An arbitrarily small rotation around the axis defined by $\boldsymbol{l}_{12}$ can be defined using (2.4) as

$$\boldsymbol{q}\{\alpha \boldsymbol{l}_{12}\} \approx \begin{bmatrix} 1 \\ \frac{1}{2} \alpha \boldsymbol{l}_{12} \end{bmatrix}, \tag{A.4}$$

for a small α. Then, the product $\tilde{\boldsymbol{q}}_k \circ \boldsymbol{q}\{\alpha \boldsymbol{l}_{12}\}$ results in

$$\tilde{\boldsymbol{q}}_k \circ \boldsymbol{q}\{\alpha \boldsymbol{l}_{12}\} \approx \begin{bmatrix} \tilde{q}_w - \frac{1}{2} \alpha \tilde{\boldsymbol{q}}_v^\top \boldsymbol{l}_{12} \\ \frac{1}{2} \alpha \tilde{q}_w \boldsymbol{l}_{12} + \tilde{\boldsymbol{q}}_v + \frac{1}{2} \alpha \tilde{\boldsymbol{q}}_v \times \boldsymbol{l}_{12}. \end{bmatrix} \tag{A.5}$$

The solution $\tilde{\boldsymbol{q}}_k \circ \boldsymbol{q}\{\alpha \boldsymbol{l}_{12}\} - \tilde{\boldsymbol{q}}_k$ is then simplified to

$$\Delta \boldsymbol{q}_k = \tilde{\boldsymbol{q}}_k \circ \boldsymbol{q}\{\alpha \boldsymbol{l}_{12}\} - \tilde{\boldsymbol{q}}_k \approx \frac{1}{2}\alpha \begin{bmatrix} -\tilde{\boldsymbol{q}}_v^\top \boldsymbol{l}_{12} \\ \tilde{q}_w \boldsymbol{l}_{12} + \tilde{\boldsymbol{q}}_v \times \boldsymbol{l}_{12} \end{bmatrix}. \tag{A.6}$$

Then, the product $\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\Delta \boldsymbol{q}_k$ is written as

$$\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\Delta \boldsymbol{q}_k \approx \alpha \left[\tilde{q}_w \boldsymbol{l}_{12} + \tilde{\boldsymbol{q}}_v \times \boldsymbol{l}_{12} \quad \tilde{\boldsymbol{q}}_v^\top \boldsymbol{l}_{12} \boldsymbol{I}_3 - \boldsymbol{l}_{12}\tilde{\boldsymbol{q}}_v^\top + \tilde{\boldsymbol{q}}_v \boldsymbol{l}_{12}^\top - \tilde{q}_w [\boldsymbol{l}_{12} \times]\right] \begin{bmatrix} -\tilde{\boldsymbol{q}}_v^\top \boldsymbol{l}_{12} \\ \tilde{q}_w \boldsymbol{l}_{12} + \tilde{\boldsymbol{q}}_v \times \boldsymbol{l}_{12} \end{bmatrix}. \tag{A.7}$$

Performing this matrix product and simplifying, one obtains

$$\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\Delta \boldsymbol{q}_k \approx \alpha \left[-\tilde{q}_w \boldsymbol{l}_{12} \tilde{\boldsymbol{q}}_v^\top \boldsymbol{l}_{12} + \tilde{q}_w \tilde{\boldsymbol{q}}_v \boldsymbol{l}_{12}^\top \boldsymbol{l}_{12} - \tilde{q}_w \boldsymbol{l}_{12} \times (\tilde{\boldsymbol{q}}_v \times \boldsymbol{l}_{12})\right]. \tag{A.8}$$

Finally, noting the cross-product property $\boldsymbol{a} \times \boldsymbol{b} \times \boldsymbol{c} = (\boldsymbol{a} \cdot \boldsymbol{c})\boldsymbol{b} - (\boldsymbol{a} \cdot \boldsymbol{b})\boldsymbol{c}$, we trivially obtain $\boldsymbol{J}_{\boldsymbol{q}}(\tilde{\boldsymbol{q}}_k, \boldsymbol{l}_{12})\Delta \boldsymbol{q}_k = \boldsymbol{0}$. Therefore, it has been confirmed analytically that (A.2) is a solution of the system of equations (A.1).

Appendix B

Derivations

B.1 Quaternion to rotation matrix uncertainty propagation

This Section details the computation of the covariance matrix relative to the entries of the rotation matrix, $\boldsymbol{Q}_{\boldsymbol{R}(\boldsymbol{N})\boldsymbol{R}(\boldsymbol{N})}$, having the covariance matrix of the quaternion $\boldsymbol{Q}_{\boldsymbol{q}(\boldsymbol{N})\boldsymbol{q}(\boldsymbol{N})}$.

Since the entries of a rotation matrix $\boldsymbol{R}$ can be written as a function of the unit quaternion $\boldsymbol{q}$, matrix $\boldsymbol{Q}_{\boldsymbol{RR}}$ is obtained from $\boldsymbol{Q}_{\boldsymbol{qq}}$ using the covariance propagation law. The relation between $\boldsymbol{R}$ and $\boldsymbol{q}$ is written in (2.5). As it is nonlinear in $\boldsymbol{q}$, a linearization is done. The following is based on [44].

Assuming that $\boldsymbol{q}$ is Gaussian distributed, with covariance matrix $\boldsymbol{Q}_{\boldsymbol{qq}}$, a first order Taylor approximation of the nonlinear functions relative to the entries of $\boldsymbol{R}(\boldsymbol{q})$ implies writting $\boldsymbol{Q}_{\boldsymbol{RR}}$ as

$$\boldsymbol{Q}_{\boldsymbol{RR}} = \boldsymbol{J}'\boldsymbol{Q}_{\boldsymbol{qq}}\boldsymbol{J}'^{\top}, \tag{B.1}$$

where $\boldsymbol{J}'$ is the Jacobian matrix of the entries of $\boldsymbol{R}(\boldsymbol{q})$ in relation to $\boldsymbol{q}$. Since $\boldsymbol{Q}_{\boldsymbol{qq}} \in \mathbb{R}^{4\times 4}$, the maximum rank of $\boldsymbol{Q}_{\boldsymbol{RR}}$ would be equal to 4, implying a positive semi-definite matrix, in this case.

Therefore, a second order Taylor approximation is used, resulting in $\boldsymbol{Q}_{\boldsymbol{RR}}$ computed as

$$\boldsymbol{Q}_{\boldsymbol{RR}} = \boldsymbol{J}'\boldsymbol{Q}_{\boldsymbol{qq}}\boldsymbol{J}'^{\top} + \frac{1}{2}\left[\operatorname{tr}\left(\boldsymbol{Q}_{\boldsymbol{qq}}\boldsymbol{J}''_{i}\boldsymbol{Q}_{\boldsymbol{qq}}\boldsymbol{J}''_{j}\right)\right]_{ij}. \tag{B.2}$$

In this expression, $\boldsymbol{J}''_{i}$ denotes the derivative $\frac{d}{d\boldsymbol{q}}$ of the i-th line of $\boldsymbol{J}'$, while $[m(i,j)]_{ij}$ represents a matrix, where the entry at position i,j is computed according to expression $m(i,j)$. The notation $\operatorname{tr}(\boldsymbol{M})$ corresponds to the trace of matrix $\boldsymbol{M}$.

Then, according to the expression for $\boldsymbol{R}(\boldsymbol{q})$ in (2.5), $\boldsymbol{J'}$ is written as in (B.3).

$$\boldsymbol{J'} = 2 \begin{bmatrix} 0 & 0 & -2q_y & -2q_z \\ q_z & q_y & q_x & q_w \\ -q_y & q_z & -q_w & q_x \\ -q_z & q_y & q_x & -q_w \\ 0 & -2q_x & 0 & -2q_z \\ q_x & q_w & q_z & q_y \\ q_y & q_z & q_w & q_x \\ -q_x & -q_w & q_z & q_y \\ 0 - 2q_x & -2q_y & 0 & \end{bmatrix} \tag{B.3}$$

Then, as mentioned before, $\boldsymbol{J''}_i$ is simply the Jacobian matrix of line i. As an example, $\boldsymbol{J''}_1$ and $\boldsymbol{J''}_1$ are computed as follows:

$$\boldsymbol{J''}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -4 & 0 \\ 0 & 0 & 0 & -4 \end{bmatrix}, \quad \boldsymbol{J''}_2 = \begin{bmatrix} 0 & 0 & 0 & 2 \\ 0 & 0 & 2 & 0 \\ 0 & 2 & 0 & 0 \\ 2 & 0 & 0 & 0 \end{bmatrix}, \quad \dots \tag{B.4}$$

Using these expressions based on a second order Taylor expansion, $\boldsymbol{Q_{RR}}$ becomes full rank, and is successfully computed from quaternion $\boldsymbol{q}$ and its covariance matrix $\boldsymbol{Q_{qq}}$.

B.2 Hessian matrix of the Lagrangian function for the nonlinear least-squares

This Section presents the computation of the Hessian matrix $\boldsymbol{J}_{\boldsymbol{g},\gamma^s}$, used in the Newton-Raphson method presented in Section 4.5.2, in order to solve the nonlinear least-squares problem.

Apart from some constants, this Hessian matrix $\boldsymbol{J}_{\boldsymbol{g},\gamma^s}$ corresponds to the Jacobian matrix of the function defined in (B.5), in relation to $\boldsymbol{x}_n = \left[\boldsymbol{q}^\top, \lambda\right]^\top$.

$$\boldsymbol{F}(\boldsymbol{x}_n) = \begin{bmatrix} \boldsymbol{J}^\top_{\boldsymbol{R}(\boldsymbol{q})} \boldsymbol{Q}^{-1} vec(\hat{\boldsymbol{R}} - \boldsymbol{R}(\boldsymbol{q})) + \lambda \boldsymbol{q} \\ \boldsymbol{q}^\top \boldsymbol{q} - 1 \end{bmatrix} \tag{B.5}$$

Then, matrix $\boldsymbol{J}_{\boldsymbol{g},\gamma^s}$ is subdivided in

$$\boldsymbol{J}_{\boldsymbol{g},\gamma^s} = \begin{bmatrix} \boldsymbol{J}^1_{\boldsymbol{q}} & \boldsymbol{J}^1_\lambda \\ \boldsymbol{J}^2_{\boldsymbol{q}} & \boldsymbol{J}^2_\lambda \end{bmatrix}, \tag{B.6}$$

where $\boldsymbol{J}^1_{\boldsymbol{q}}$ denotes the Jacobian matrix of the first line of $\boldsymbol{F}(\boldsymbol{x}_n)$ in order to $\boldsymbol{q}$. The remaining blocks are defined in a similar way. As such, the derived expressions for the blocks of the matrix are presented in the following.

$$\boldsymbol{J}_{\boldsymbol{q}}^{1} = \left[\begin{bmatrix} \frac{d}{dq_w}\boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})}^{\top} \\ \frac{d}{dq_x}\boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})}^{\top} \\ \frac{d}{dq_y}\boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})}^{\top} \\ \frac{d}{dq_z}\boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})}^{\top} \end{bmatrix} \boldsymbol{Q}^{-1} vec(\hat{\boldsymbol{R}} - \boldsymbol{R}(\boldsymbol{q})) \right]^{\top} - \boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})}^{\top}\boldsymbol{Q}^{-1}\boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})} + \lambda \boldsymbol{I}_4, \tag{B.7}$$

$$\boldsymbol{J}_{\lambda}^{1} = \boldsymbol{q}, \qquad \boldsymbol{J}_{\boldsymbol{q}}^{2} = 2\boldsymbol{q}^{\top}, \qquad \boldsymbol{J}_{\lambda}^{2} = 0$$

The expressions for $\boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})}$ and its derivatives, such as $\frac{d}{dq_w}\boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})}$, are trivial to obtain. In fact, $\boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})}$ coincides with matrix $\boldsymbol{J'}$ whose expression is written in (B.3) of Appendix B.1. Regarding the derivatives, an example is given next for computing, for example, $\frac{d}{dq_w}\boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})}$:

$$\frac{d}{dq_w}\boldsymbol{J}_{\boldsymbol{R}(\boldsymbol{q})} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 \\ 0 & 0 & -2 & 0 \\ 0 & 0 & 0 & -2 \\ 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{B.8}$$

The remaining derivatives are computed in a similar way, therefore completely describing the method of solving the nonlinear least-squares problem, which is part of the MC-LAMBDA method.

Appendix C

DD correction model due to non-synchronized receivers

The GNSS receivers used to perform the experiments are seen to not be perfectly synchronized. On any given epoch, there is a difference in the sampling time of the GNSS signals between different receivers which can have a magnitude of up to $\approx 1\,\text{ms}$. As such, there is the need to know to what extent this offset impacts the reliability of the considered observation model, in which the receivers are assumed to be synchronized. The following analysis is based on [10].

First, consider the code measurements equation (C.1). The time instant $\bar{t}_i$ is the time at which the pseudorange observation is measured by receiver i. This measured time, $\bar{t}_i$, corrected by the receiver clock offset dt_i, corresponds to t_i.

$$P_i^k(\bar{t}_i) = \rho_i^k + c(dt_i(\bar{t}_i) - dt^k(t_i - \tau_i^k)) + I_i^k + T_i^k + \epsilon_{P,i}^k, \tag{C.1}$$

Remember that ρ_i^k is computed as $\rho_i^k = \left\|\boldsymbol{R}_3(\omega_e \tau_i^k)\boldsymbol{r}^k(t_i - \tau_i^k) - \boldsymbol{r}_i(t_i)\right\|$. Rewriting the range between receiver i and satellite k, ρ_i^k, as a function of the LOS vector, $\boldsymbol{u}_i^k$, we have

$$\boldsymbol{u}_i^k = \frac{\boldsymbol{R}_3(\omega_e \tau_i^k)\boldsymbol{r}^k(t_i - \tau_i^k) - \boldsymbol{r}_i(t_i)}{\rho_i^k}. \tag{C.2}$$

Replacing in (C.1), the pseudorange is written as

$$P_i^k(\bar{t}_i) = {\boldsymbol{u}_i^k}^T\big(\boldsymbol{R}_3(\omega_e \tau_i^k)\boldsymbol{r}^k(t_i - \tau_i^k) - \boldsymbol{r}_i(t_i)\big) + c(dt_i(\bar{t}_i) - dt^k(t_i - \tau_i^k)) + I_i^k + T_i^k + \epsilon_{P,i}^k. \tag{C.3}$$

For a different receiver j, the analogous pseudorange observation is written as (C.4).

$$P_j^k(\bar{t}_j) = {\boldsymbol{u}_j^k}^T\big(\boldsymbol{R}_3(\omega_e \tau_j^k)\boldsymbol{r}^k(t_j - \tau_j^k) - \boldsymbol{r}_j(t_j)\big) + c(dt_j(\bar{t}_j) - dt^k(t_j - \tau_j^k)) + I_j^k + T_j^k + \epsilon_{P,j}^k \tag{C.4}$$

The receivers are considered to be non-synchronized when $t_i \neq t_j$. As said previously, using the hardware at hand, it is seen experimentally that, in some cases, this difference reaches $|t_i - t_j| = 1\,\text{ms}$.

To compute the single differences (SD) between measurements P_i^k and P_j^k, the usual simplifications for short baseline can be made. Firstly, the troposphere and ionosphere delays are considered approximately the same between the two receivers. Furthermore, considering that the satellite clock offset varies slowly in time, that is, $dt^k(t_i - \tau_i^k) \approx dt^k(t_j - \tau_j^k)$, the satellite clock offset is also eliminated when subtracting the measurements. As such, the expression for the non-synchronized SD code measurements is

$$\begin{aligned} P_{ij}^k = {} & {\boldsymbol{u}_i^k}^T \big(\boldsymbol{R}_3(\omega_e \tau_i^k)\boldsymbol{r}^k(t_i - \tau_i^k) - \boldsymbol{r}_i(t_i)\big) - {\boldsymbol{u}_j^k}^T \big(\boldsymbol{R}_3(\omega_e \tau_j^k)\boldsymbol{r}^k(t_j - \tau_j^k) - \boldsymbol{r}_j(t_j)\big) + \\ & + c(dt_i(\bar{t}_i) - dt_j(\bar{t}_j)) + \epsilon_{P,ij}^k. \end{aligned} \tag{C.5}$$

To simplify this expression, some considerations can be made. The average velocity of a satellite belonging to the GPS constellation is of ≈ 4 km/s, while the average altitude is of around 20 000 km. For a time difference of 1 ms, the variation of position is negligible when compared to the receiver to satellite distance. Since $\tau_i^k \approx \frac{\rho_i^k}{c}$, it follows that $\tau_i^k \approx \tau_j^k$. Together with the fact that the baseline is very short when compared to the receiver to satellite range, the LOS vectors can be considered parallel, such that $\boldsymbol{u}_j^k \approx \boldsymbol{u}_i^k = \boldsymbol{u}^k$. The error incurred when doing this approximation is of the order $|({\boldsymbol{u}_j^k}^T - {\boldsymbol{u}_i^k}^T)\boldsymbol{R}_3(\omega_e \tau_j^k)\boldsymbol{r}^k(t_j - \tau_j^k)|$. For a time difference $\Delta t = |t_i - t_j| = 1$ ms and the average velocity and altitude mentioned before, $\|{\boldsymbol{u}_j^k}^T - {\boldsymbol{u}_i^k}^T\| \approx 1 \times 10^{-7}$ m. Given the geometry, this vector is approximately perpendicular to the satellite position and therefore the approximation of parallel LOS vectors is valid. Using this simplification, the SD code measurement are written as

$$\begin{aligned} P_{ij}^k = {} & {\boldsymbol{u}^k}^T \left[\big(\boldsymbol{R}_3(\omega_e \tau_i^k)\boldsymbol{r}^k(t_i - \tau_i^k) - \boldsymbol{r}_i(t_i)\big) - \big(\boldsymbol{R}_3(\omega_e \tau_j^k)\boldsymbol{r}^k(t_j - \tau_j^k) - \boldsymbol{r}_j(t_j)\big)\right] + \\ & + c(dt_i(\bar{t}_i) - dt_j(\bar{t}_j)) + \epsilon_{P,ij}^k. \end{aligned} \tag{C.6}$$

Finally, the term $\boldsymbol{r}_j(t_j) - \boldsymbol{r}_i(t_i)$ must be extrapolated in order to relate the SD measurements to the baseline $\boldsymbol{b}_{ij}$ at time t_i. It will be assumed that the influence of the velocity of the antennas in the ENU frame is neglectible, considering a small time difference between receivers and low linear and angular velocities. Doing so, there is no need to correct the receiver positions, as $\boldsymbol{r}_j(t_j) \approx \boldsymbol{r}_j(t_i)$. Then, noting that $\boldsymbol{r}_j(t_i) - \boldsymbol{r}_i(t_i) = \boldsymbol{b}_{ij}$, (C.6) is written as

$$P_{ij}^k = {\boldsymbol{u}^k}^T \boldsymbol{b}_{ij}(t_i) + E_{ij}^k + c(dt_i(\bar{t}_i) - dt_j(\bar{t}_j)) + \epsilon_{P,ij}^k, \tag{C.7}$$

with E_{ij}^k corresponds to the extrapolation. or correction term, given by

$$E_{ij}^k = {\boldsymbol{u}^k}^T \big(\boldsymbol{R}_3(\omega_e \tau_i^k)\boldsymbol{r}^k(t_i - \tau_i^k) - \boldsymbol{R}_3(\omega_e \tau_j^k)\boldsymbol{r}^k(t_j - \tau_j^k)\big). \tag{C.8}$$

This term is shown to take values of up to ≈ 30 cm using experimental data taken from receivers unsynchronized by 5×10^{-4} s, which confirms that it is not neglegible, taking into account the very low noise present in the carrier phase observations. The correction term is then subtracted to the DD code and carrier phase measurements, in order to improve the accuracy of the attitude estimation solution.

Appendix D

Managing changes in the observed satellites and reference satellite

Both the number of satellites in view and the chosen reference satellite can vary freely during the processing of the GNSS data. Dealing with these variations entails changing both the estimated EKF state vector $\hat{\boldsymbol{x}}_{k|k}$ and its estimation error covariance matrix $\boldsymbol{P}_{k|k}$ entries relative to the DD ambiguities.

The methods used to achieve this rest on two different assumptions relating two consecutive epochs. If the reference satellite changes, the new reference satellite must have been in view during the previous epoch. Furthermore, the old reference satellite must still be in view during the new epoch. These assumptions are valid by definition – since the reference satellite is the one with highest elevation angle, it is very likely that is had been observed in the previous epoch, and will be observed in the next one.

The two types of changes are treated separately. When both happen in the transition between two consecutive epochs, the change in the observed satellites is addressed first, followed by the algorithm which deals with changes in the reference satellite.

Lastly, it can also happen that due to obstacles, loss of lock or other interference, the number of satellites observed at a certain epoch falls below the minimum allowed number. In this case, the GNSS observations can not be used to estimate the attitude. Furthermore, once the minimum number of satellites is recovered, some techniques can be used to improve the initial estimate of the new set of DD ambiguities.

Change in the number of satellites in view

For any satellite, r, that is no longer in view, the DD ambiguity N_{ij}^{pr} is simply eliminated. The same happens to the line and column vectors in $\boldsymbol{P}_{k|k}$ which are related to satellite g. If a satellite, s, is observed for the first time, the DD ambiguity N_{km}^{ps} and the related entries in $\boldsymbol{P}_{k|k}$ are initalized as 0 and 1×10^4, respectively..

At the end of this process, the DD ambiguities vector and the covariance matrix $\boldsymbol{P}_{k|k}$ have a structure that agrees with the new list of satellites in view, without loss of relevant information from the previous iterations.

Change in the reference satellite

Let p denote the reference satellite of the previous epoch, while g is the new reference satellite and q is any other satellite which is in view during both epochs. Then, N_{ij}^{pq} is the DD ambiguity in the previous epoch, and N_{ij}^{gq} the corresponding new ambiguity that one must compute to recover from the change. From the definition of the double differenced ambiguities, it is simple to realize that

$$N_{ij}^{gq} = N_{ij}^{pq} - N_{ij}^{pg}. \tag{D.1}$$

The DD ambiguity N_{ij}^{gp} is simply computed as $N_{ij}^{gp} = -N_{ij}^{pg}$. Applying these transformations to the DD ambiguity relative to each satellite, the $\boldsymbol{N}$ vector is reconstructed.

Several other expressions have been deduced in order to recover the covariance matrix $\boldsymbol{P}_{k|k}$. In the proposed solution, the state of the EKF includes the rotation quaternion, the angular velocity bias and the DD ambiguities. The new diagonal entries of $\boldsymbol{P}_{k|k}$, relative to the DD ambiguities, are now given by (D.2), where the notation $cov\{a, b\}$ represents the covariance between random variables a and b.

$$var\{N_{ij}^{gq}\} = var\{N_{ij}^{pq}\} + var\{N_{ij}^{qg}\} - 2cov\{N_{ij}^{pq}, N_{ij}^{pg}\} \tag{D.2}$$

As before, the new DD ambiguity N_{ij}^{gp} is a special case, such that $var\{N_{ij}^{gp}\} = var\{N_{ij}^{pg}\}$. Using these expressions, the diagonal of $\boldsymbol{P}_{k|k}$ is recovered.

The new entries of $\boldsymbol{P}_{k|k}$ which are covariances between two DD ambiguities are computed as (D.3), for q and r being any observed satellites other than the previous and new reference satellites.

$$cov\{N_{ij}^{gq}, N_{ij}^{gr}\} = \frac{1}{2}\left(var\{N_{ij}^{gq}\} + var\{N_{ij}^{gr}\} - var\{N_{ij}^{pq}\} - var\{N_{ij}^{pr}\} + 2cov\{N_{ij}^{pq}, N_{ij}^{pr}\}\right) \tag{D.3}$$

The special case of covariance between N_{ij}^{gp} and any other ambiguity relative to a satellite q is given by

$$cov\{N_{ij}^{gp}, N_{ij}^{gq}\} = \frac{1}{2}\left(var\{N_{ij}^{pg}\} + var\{N_{ij}^{gq}\} - var\{N_{ij}^{pq}\}\right). \tag{D.4}$$

Finally, the entries of $\boldsymbol{P}_{k|k}$ relative to covariances between any new DD ambiguity, other than N_{ij}^{gp}, and other variables of the state (denoted by x_i), such as the quaternions and angular velocity biases, is given in (D.5).

$$cov\{x_i, N_{ij}^{gq}\} = cov\{x_i, N_{ij}^{pq}\} - cov\{x_i, N_{ij}^{qg}\} \tag{D.5}$$

As before, the expression is different for N_{ij}^{gp}, where $cov\{x_i, N_{ij}^{gp}\} = -cov\{x_i, N_{ij}^{pg}\}$.

Recovery from epochs with insufficient GNSS data

During the execution of the filter, whenever there occurs an epoch where the collected data is not sufficient to update the attitude estimate, the correction step of the EKF is not executed. As such, the attitude output will rely only on the integration of the gyroscope measurements.

Once there is a recovery on the number of observed satellites, the estimated DD ambiguities relative to satellites that were momentarily lost must be reset, since loss of lock has occurred. However, by assuming that the present attitude estimate is valid, the new set of DD ambiguities can be estimated by resorting to the carrier phase observations. In fact, assuming that the attitude is known and using the nominal system observation model (3.12), the ambiguities can be estimated as

$$\boldsymbol{n}_k \approx \frac{1}{\lambda}\left[\boldsymbol{z}_{k,\Phi} - \begin{bmatrix} \boldsymbol{\Upsilon}\ \boldsymbol{q_k} \circ \boldsymbol{l}_{i1} \circ \boldsymbol{q_k}^* \\ \vdots \\ \boldsymbol{\Upsilon}\ \boldsymbol{q_k} \circ \boldsymbol{l}_{in} \circ \boldsymbol{q_k}^* \end{bmatrix}\right], \tag{D.6}$$

where $\boldsymbol{z}_{k,\Phi}$ are the DD carrier phase measurements. This expression gives the a vector of DD ambiguities relative to all the satellites in view. However, only the ones corresponding to new satellites are kept, while the remaining are discarded since their previous estimates are valid. Due to the low noise present in the carrier phase measurements, the accuracy of this initial estimate depends on the quality of the attitude estimate $\boldsymbol{q}_k$, and should improve the recovery performance of the proposed algorithm.