TÉCNICO LISBOA

Navigation Systems Based On Single Pseudo-Range Measurements: Design and Experimental Evaluation

João Manuel Lopes Agostinho Correia Franco

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisor: Prof. Pedro Tiago Martins Batista

Examination Committee

Chairperson: Prof. Paulo Jorge Coelho Ramalho Oliveira
Supervisor: Prof. Pedro Tiago Martins Batista
Member of the Committee: Prof. Bruno João Nogueira Guerreiro

January 2021

Manecas e Janica

Acknowledgments

First and foremost I would like to thank my parents who have supported me throughout all my academic journey. Without them none of this would be possible. I will keep on making sure your money was well spent.

To the rest of my family, my biggest thank you for making me the person I am today. Also, thank you, Leonor and Graciosa, for continuously asking me when was this work due; sometimes I did need the reminder. To my friends and girlfriend I thank for their support and for being so open to hear me ramble on and on about something they don't even care about.

I would also like to thank my supervisor Pedro Batista for helping me elaborate this Thesis. Also, for his role in changing many students' lives by teaching control theory with remarkable success.

Finally, I would like to thank my course fellow Tiago Oliveira and his master's thesis supervisors for their work on "Rapid Development and Prototyping Environment for Testing of Unmanned Aerial Vehicles". Without them, the completeness of this thesis would not be achievable.

This work was supported by the Fundação para a Ciência e a Tecnologia (FCT) through FCT project DECENTER [LISBOA-01-0145-FEDER- 029605], funded by the Lisboa 2020 and PIDDAC programs.

Resumo

O tema central desta tese é a navegação apoiada em sistemas de base acústica que pressupõem a aquisição de uma só medida de distância. A dificuldade de se obter, com a precisão adequada, a velocidade do som, problema especialmente relevante em aplicações marinhas mas também noutras, como aeroespaciais, é tomada em conta através da consideração de que as medidas de distância obtidas pelo sistema estão afetadas por um fator multiplicativo desconhecido. A não linearidade do problema de navegação é contornada através da aumentação do número de estados do sistema. É mostrado que o sistema linear obtido através desta abordagem é observável e, consequentemente, um filtro de Kalman é utilizado de forma a obterem-se estimativas dos estados; estas apresentam erros com decaimento exponencialmente estável para todas as condições iniciais. Posteriormente, são efetuadas simulações, nas quais se considera a existência de ruído, com a finalidade de comparar a solução proposta com duas técnicas comuns de estimação não linear, nomeadamente os filtros de Kalman estendido e *unscented*. Os dois últimos apresentam desempenhos semelhantes à nova solução, mas são incapazes de o fazer para todas as condições iniciais. São também efetuadas Simulações de Monte Carlo que fornecem uma perceção mais aprofundada do comportamento de cada um dos processos de estimação. Finalmente, são executadas experiências num ambiente controlado de modo a comprovar a aplicabilidade da solução proposta.

Palavras-chave: medida de distância única, navegação subaquática, velocidade do som desconhecida, filtro de Kalman, Bayesian Cramér-Rao bound, simulações de Monte Carlo

Abstract

This thesis refers to acoustic navigation systems based on single range measurements. The difficulty of accurately determining the speed of sound on the propagation medium, a matter of special relevance in sub aquatic and aeronautic applications, is addressed by considering that the range measurements acquired by the system are affected by an unknown multiplicative coefficient. The inherent nonlinear nature of the navigation problem is tackled by performing state augmentation. The discrete-time linear system obtained with this approach is shown to be observable and a Kalman filter is employed in order to obtain state estimates with globally exponentially stable error dynamics. Simulations, assuming noisy environments, are conducted to compare this solution with common nonlinear estimation techniques, namely the extended Kalman filter and the unscented Kalman filter. The latter are shown to obtain comparable results, but fail to provide global convergence guarantees. Monte Carlo simulations supply further insights on filtering performances. Finally, experiments in a controlled environment are carried out, attesting the applicability of the proposed solution.

Keywords: single range, underwater navigation, unknown speed of sound, Kalman filter, Bayesian Cramér-Rao bound, Monte Carlo simulations

Contents

List of Tables

List of Figures

Acronyms

GPS Global Positioning System

UV Underwater Vehicle

LBL Long Baseline

SBL Short Baseline

USBL Ultra-Short Baseline

EKF Extended Kalman Filter

GES Globally Exponentially Stable

UKF Unscented Kalman Filter

LKF Linear Kalman Filter

ESV Effective Sound Velocity

BO Bayesian Optimization

BCRB Bayesian Cramér-Rao Bound

DVL Doppler Velocity Log

AHRS Attitude Heading and Reference System

GRV Gaussian Randum Variable

UT Unscented Transform

MOCAP Motion Capture

Chapter 1

Introduction

1.1 Motivation

The topic of navigation has been around ever since the humankind started to explore new worlds. From celestial navigation to portable global positioning system (GPS) devices, from the first Polynesian navigators to nowadays work commuters, the field of navigation has traveled a long way. Nevertheless, no field of study is ever finished and, like so, neither is the world of navigation.

By the end of the 20^{th} century, great success had already been achieved in the subject of navigation above sea level. In fact, the development, in the '70s, of the GPS by the United State's government changed forever the paradigm of navigation. Its biggest flaw is, probably, the fact that it could not address sub aquatic applications. This flaw resulted in the development of new systems capable of operating under water, and, since the human mind is never satisfied, new flaws were discovered and newer solutions were invented.

The need for underwater vehicle (UV) positioning has increased in the recent years. Breakthroughs in the field of artificial intelligence have lead to a significant burst of autonomous UV applications. Other engineering quests, such as offshore energy production and seafloor mapping, often require the operation of UVs to achieve prosperous results. Also, the study of the population of marine species and their migrations can largely benefit from the advancements in underwater positioning.

Nowadays, acoustic systems are the standard for accurate underwater navigation. Three major classes of systems are available for position determination. These are the long baseline (LBL) systems, the short baseline (SBL) systems, and the ultra short baseline (USBL) systems. Each of these solutions comprises its own specific implementation and expected performance, but they all have in common the fact that, like in the GPS, multiple waves, from multiple transponders, are propagated between the navigation system and the user. From these multiple range measurement systems a series of problems arise. One of them, and, probably, the most important one is the fact that multiple range measurements, by requiring multiple transponders, amount to increasing system complexity and deployment costs. Another problem is the fact that a fail in one or more transponders might lead to the system being rendered useless, since common multilateration techniques might become compromised. A way to solve these

issues is to use just one transponder capable of performing the same task as all the systems mentioned before.

Having this in mind, acoustic navigation based in single range measurements has attracted the interest of the scientific community in the recent years. It is important to notice that only one range measurement is obviously not enough to uniquely determine the position of a vehicle, and thus, this type of approach usually requires the integration of other subsystems capable of acquiring information regarding vehicle trajectory. This, however, does not tend to pose a conceptual drawback, since advancements in micro and nanotechnologies have enabled even small vehicles to be equipped with these types of systems.

The possibility of extending single range measurement navigation to fields other than the sub aquatic world is also a fact of considerable matter. Progresses in this specific field are, in most cases, applicable to the aviation and aerospace segments. In fact, the vast majority of airplanes is already equipped with the systems required to achieve good navigation performance, should this be based on a single range measurement architecture. The same can be said about spaceships and rockets, which are widely considered the vehicles that will eventually carry the humankind towards a new era of space exploration.

Due to its clear advantages when compared to multi-measurement solutions, and due to its broad extendibility to different purposes, single-range systems designed for underwater navigation are a subject of the utmost importance.

1.2 State of the art

Studies in the field of single-transponder underwater acoustic navigation have diverged into two classes of solutions. As such, the available literature regarding the subject addresses both moving and stationary single-transponder solutions.

An early surface-moving transponder approach was pursued in [1], where the authors debate the utility of sonar readings to determine the motion of an underwater subject. More advanced work on the matter was carried out by various authors who considered well defined trajectories of the support surface vehicle in order to estimate target position. Instances of this approach are presented in [2] and [3], where nonlinear least-mean-square methods and centralized extended Kalman filters, respectively, are used to overcome the nonlinearities inherent to the localization problem. Still regarding surface moving beacons, extensive work has been done to determine support ship trajectories that minimize position estimation errors. To that end, [4] relies on the determination of Fisher information matrices, and [5] extends this method to noisy scenarios.

One of the first works addressing the concept of underwater navigation based on single range measurements to a stationary beacon dates back to the beginning of the century [6]. The author proposes a solution to overcome the known limitations of dead-reckoning navigation by combining a high performance dead-reckoning system with an innovative synthetic long baseline; the latter consisting on a set of consecutive discrete-time range measurements from the moving vehicle to a stationary beacon. Kalman filtering was applied to a linearized version of the system dynamics. In [7], the navigation prob-

lem is addressed with special concerns for known current velocities. System linearization is performed and an extended Kalman filter (EKF) is used to produce state estimates, but no guarantees of global asymptotically stable error dynamics are provided. The same strategy was then employed in [8] and [9], where unknown currents are considered. In [10], algebraic methods are used to study system observability and to perform state estimation. A state augmentation approach, which yields estimates with globally exponentially stable (GES) error dynamics, is brought forward in [11]. Also motivated by state augmentation processes, the authors of [12] propose a novel estimator based on the least entropy-like estimator [13] capable of dealing with unavoidable measurement outliers.

A very important factor when considering acoustic range measurements for navigation purposes is that the speed of sound needed to obtain those measurements tends to vary according to specific environment conditions. Temperature, salinity and pressure were considered in the pioneering article [14] to determine an empirical formula capable of defining the speed of sound underwater. Over the years, numerous other works have also addressed the problem of characterizing speed of sound profiles underwater, being the UNESCO algorithm [15] considered the international standard for this purpose. The domain of underwater navigation has extensively benefited from the determination of the speed of sound in water, since range measurement devices critically depend on this value. Nevertheless, speed of sound profiles are sometimes not possible to determine or are ill determined. Having this in mind, researchers have studied the underwater navigation problem while explicitly considering the speed of sound as an unknown variable available for estimation. Addressing this particular subject, [16] envisions a moving LBL positioning algorithm based in uncertain least-squares (ULS). The authors tackle system singularities by performing state estimation with an ULS-based unscented Kalman filter (UKF). In [17], system augmentation is used to obtain position, current speed, and speed of sound estimates in a LBL navigation configuration. A linear Kalman filter (LKF) provides these estimates with global convergence guaranties. The same strategy is employed in [18], where the author assumes a one-way-travel-time LBL configuration with unknown speed of sound and possible clock offset between system components.

Studies combining the concept of single range measurement navigation and the difficulties of proper speed of sound determination have recently emerged. In [19], speed of sound is explicitly estimated in a single pseudo-range measurement navigation/source localization problem, where state augmentation is the preferred route. The new system obtained with this technique is inspected for observability purposes and a LKF provides state estimates with GES error dynamics. The work [20] proposes a solution to overcome the errors associated with a misidentified effective sound velocity (ESV) in navigation models. The researchers treat ESV as a model parameter and use the expectation maximization method to obtain estimates of this unknown parameter. The overall single-transponder navigation method also incorporates a Kalman filter. To tackle noise sensitivity and initial vehicle position errors, which might lead to fails in the two previous solutions, [21] treats the ESV as a random variable with unknown properties. Both the ESV and its statistical properties are estimated resorting to a variational Bayesian approximation method.

An overview of the concepts presented throughout the previous paragraphs is shown in Fig. 1.1.

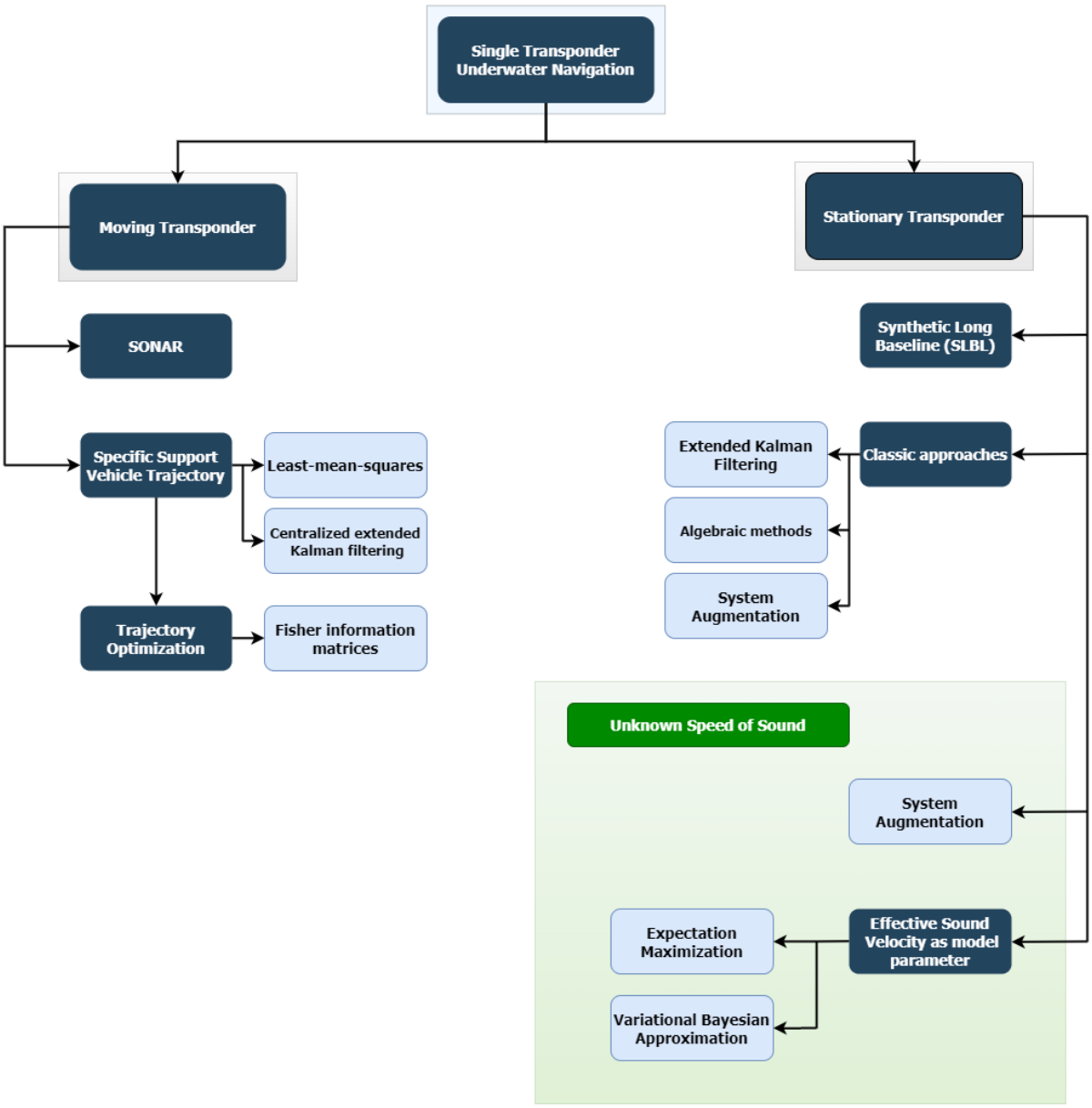

Figure 1.1: Literature overview

1.3 Contributions

In this thesis, the most simple form of underwater navigation in terms of setup and deployment, i.e., navigation with a single fixed transponder, is studied. The assumption of an unknown speed of sound in the medium is also a subject of this work. Along these lines, range measurements acquired from a transponder system are considered to be affected by an unknown multiplicative factor. Taking this into consideration, a nonlinear discrete-time system representative of the navigation problem is derived, and estimates of the position of the vehicle as well as the speed of sound unknown multiplicative coefficient are obtained with the solution proposed in [19].

In order to compare the novel solution with common nonlinear estimation techniques, an EKF and an UKF are also employed to provide vehicle and speed of sound estimates. Filtering parameters are subject to the Bayesian optimization (BO) process.

Extensive Monte Carlo simulations provide further comprehension of the performance of each filtering solution by allowing the determination of bias in the estimation processes, as well as estimation error covariance. The latter is then compared to the Bayesian Cramér-Rao bound (BCRB).

Finally, experiments in a controlled environment are performed attesting the applicability of the new solution.

Fig. 1.2 sums up the work done in this thesis.

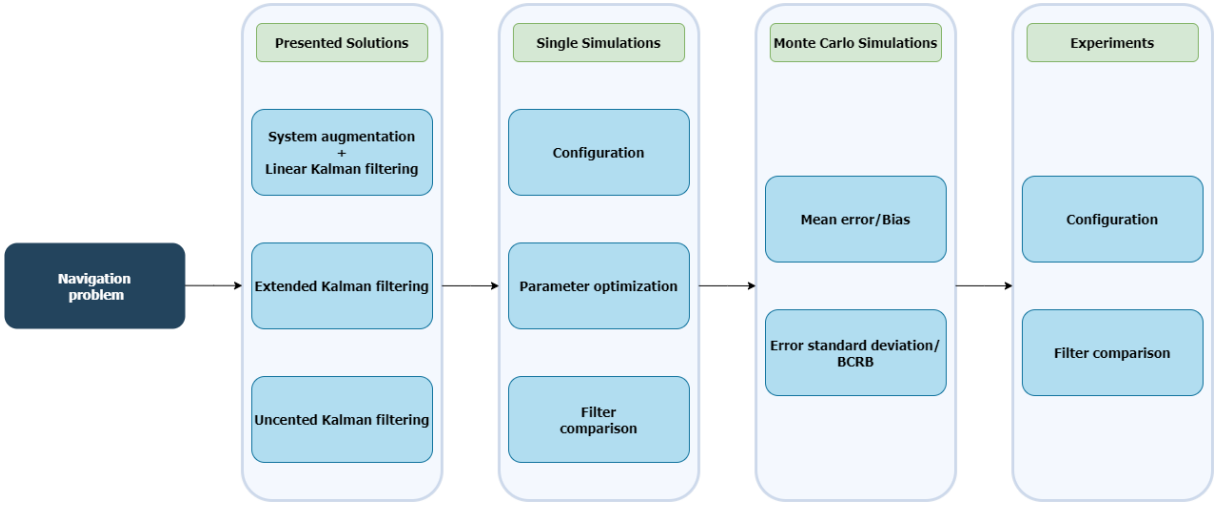

Figure 1.2: Thesis contributions

1.4 Thesis Outline

From this point onward, this thesis is organized as follows. Firstly, the problem statement and the notation used throughout this document are presented in Chapter 2. The novel solution proposed in [19] and its mathematical formulation are presented in Chapter 3. In order to compare this with extensively studied nonlinear estimation methods, Chapters 4 and 5 comprise the formulation and implementation of an extended Kalman filter and an unscented Kalman filter, respectively. An optimization algorithm called Bayesian optimization is introduced in Chapter 6, where its application to the problem at hand is explained. Chapter 7 addresses the mathematical formulation of the Bayesian Cramér-Rao bound. In Chapter 8, simulations regarding the three methods studied are performed and the results obtained are subject to comparison. Experimental results are found in Chapter 9. Lastly, conclusions related to this work are deduced in Chapter 10.

Chapter 2

Notation and problem statement

2.1 Notation

Throughout this thesis the following notation is employed. Vectors are represented in bold with each component appearing as an upper index. For example, $\mathbf{p}^x$, $\mathbf{p}^y$, and $\mathbf{p}^z$ represent the first, second, and third components of a vector $\mathbf{p} \in \mathbb{R}^3$. The bold upper case is reserved for matrices such as a $n \times n$ identity matrix, $\mathbf{I}_n$, or the zeros matrix, $\mathbf{0}_n$. A block diagonal matrix is represented by $diag(\mathbf{A}_1, ...\mathbf{A}_n)$. The inner product of $\mathbf{p} \in \mathbb{R}^3$ and $\mathbf{q} \in \mathbb{R}^3$ is denoted by $\mathbf{p} \cdot \mathbf{q}$. Finally, the special orthogonal group of dimension n is represented as $SO(n)$.

2.2 Problem statement

The navigation systems proposed in this thesis foresee the use of a single fixed acoustic transponder to acquire discrete-time range measurements to an operating vehicle. The aforementioned systems rely on a basic principle of wave propagation which states that, in the absence of a variable speed of sound, the distance traveled by the acoustic wave is equal to its speed of propagation in the medium multiplied by the time it took for the wave to travel said distance. The problem regarding this simple formulation is that the speed of sound considered to compute distances might not correspond to the true value, be it from inaccurate determination or complete lack of information. Having this in mind, one can assume that the range measurements provided by the system are affected by an unknown multiplicative factor. It is also assumed that these systems comprise the use of other navigation equipment such as a Doppler velocity log (DVL) and an attitude and heading reference system (AHRS) to determine the inertial speed of the vehicle and its orientation relative to a reference frame, respectively.

The following section takes an in-depth look at the mathematical formulation of the system under study.

2.2.1 System dynamics

As usual in navigation systems, consider the existence of $\{I\}$, a local coordinate inertial reference frame, and $\{B\}$, a coordinate frame that moves with the vehicle along its motion through the medium, the so-called body-fixed frame.

The variation of the inertial position of the vehicle, $\dot{\mathbf{p}}(\mathbf{t}) \in \mathbb{R}^3$, can be written as

$$\dot{\mathbf{p}}(t) = \mathbf{R}(t)\mathbf{v}(t), \tag{2.1}$$

where $\mathbf{R}(t) \in SO^3$ is the rotation matrix from $\{B\}$ to $\{I\}$ and $\mathbf{v}(t)$ is the velocity of the vehicle relative to $\{I\}$, expressed in body-coordinates. In this thesis, it is assumed that the vehicle is capable of acquiring both the rotation matrix $\mathbf{R}(t)$ and the velocity $\mathbf{v}(t)$. In practice, the first is usually obtained through the aforementioned AHRS and the second through the DVL, if the later has established bottom-lock.

Discrete-time range measurements to a stationary beacon, $r(k)$, can be obtained through

$$r(k) = v_s(t_k) \parallel \mathbf{s} - \mathbf{p}(t_k) \parallel \tag{2.2}$$

where $\mathbf{s} \in \mathbb{R}^3$ is the inertial position of the beacon, t_k is the sampling instant, defined with respect to the sampling period $T > 0$ and the initial time t_0, as $t_k := t_0 + kT$, $k \in \mathbb{N}$. Finally, $v_s(t_k) > 0$ denotes the strictly positive scaling factor that accounts for the uncertainties in the speed of propagation of the acoustic waves.

Considering, for the sake of simplicity, that the velocity of propagation in the medium does not change over time, the previous equations can be condensed to represent the system dynamics as

$$\begin{cases} \dot{\mathbf{p}}(t) = \mathbf{R}(t)\mathbf{v}(t) \\ \dot{v_s}(t) = 0 \\ r(k) = v_s(t_k) \parallel \mathbf{s} - \mathbf{p}(t_k) \parallel . \end{cases} \tag{2.3}$$

The above equations depict a continuous nonlinear system that can be discretized as

$$\begin{cases} \mathbf{p}(t_{k+1}) = \mathbf{p}(t_k) + \mathbf{u}(t_k) \\ v_s(t_{k+1}) = v_s(t_k) \\ r(k) = v_s(t_k) \parallel \mathbf{s} - \mathbf{p}(t_k) \parallel, \end{cases} \tag{2.4}$$

where the system input, $\mathbf{u}(t_k)$, accounts for the traveled distance, in inertial coordinates, between consecutive pseudo-range measurements. It can be computed through the integral

$$\mathbf{u}(t_k) = \int_{t_k}^{t_{k+1}} \mathbf{R}(\tau)\mathbf{v}(\tau)d\tau. \tag{2.5}$$

The problem addressed in this thesis is that of estimating $\mathbf{p}(t_k)$ and $v_s(t_k)$ assuming that both the pseudo-range measurements $r(k)$ and the system input $\mathbf{u}(t_k)$ are available at each time step k.

Chapter 3

Linear Kalman filter

To perform state estimation on the system under study, a novel solution proposed in [19] is first implemented. This consists primarily in performing state augmentation, thus obtaining a new augmented system. The new system is then analyzed for observability purposes and state estimates are obtained using a simple linear Kalman filter.

3.1 Theoretical foundation

Consider a generic system of the form

$$\begin{cases} \mathbf{x}_1(k+1) = \mathbf{x}_1(k) + \mathbf{u}(k) \\ x_2(k+1) = x_2(k) \\ r(k+1) = x_2(k+1) \parallel \mathbf{x}_1(k+1) \parallel . \end{cases} \tag{3.1}$$

This formulation is equivalent to (2.4) if $\mathbf{x}_1(k)$ and $x_2(k)$ are defined as

$$\begin{cases} \mathbf{x}_1(k) := \mathbf{p}(k) - \mathbf{s} \\ x_2(k) := v_s(k). \end{cases} \tag{3.2}$$

Assumption 1: All pseudo-ranges, $r(k)$, are greater than zero, i.e., $r(k) > 0$ for all k.

By applying the following state augmentation process, new system states are defined as

$$\begin{cases} \mathbf{z}_1(k) := x_2^2(k)\mathbf{x}_1(k) \\ z_2(k) := x_2^2(k) \\ z_3(k) := r(k). \end{cases} \tag{3.3}$$

Of these three states, only $z_3(k)$ requires some attention when deriving its evolution. Regarding $\mathbf{z}_1(k)$ and $z_2(k)$, these have a very easy-to-infer evolution which, when taking into account (3.1) and (3.3), can

be stated as

$$\begin{cases} \mathbf{z}_1(k+1) := \mathbf{z}_1(k) + z_2(k)\mathbf{u}(k) \\ z_2(k+1) := z_2(k). \end{cases} \tag{3.4}$$

Concerning the third state, notice that the squared range, $r^2(k+1)$, is

$$r^2(k+1) = x_2^2(k+1) \parallel \mathbf{x}_1(k+1) \parallel^2, \tag{3.5}$$

which, using (3.1) translates to

$$r^2(k+1) = x_2^2(k) \parallel \mathbf{x}_1(k) \parallel^2 + 2x_2^2(k)\mathbf{u}(k) \cdot \mathbf{x}_1(k) + x_2^2(k) \parallel \mathbf{u}(k) \parallel . \tag{3.6}$$

Taking a closer look at each term of (3.6), one can infer the following. From (3.1), the first term is equal to $r^2(k)$, and from (3.3), the second term is equal to $2\mathbf{u}(k) \cdot \mathbf{z}_1(k)$ and the third term is equal to $\parallel \mathbf{u}(k) \parallel^2 z_2(k)$. Substituting these equivalences in (3.6) gives

$$r^2(k+1) = 2\mathbf{u}(k) \cdot \mathbf{z}_1(k) + \parallel \mathbf{u}(k) \parallel^2 z_2(k) + r^2(k). \tag{3.7}$$

Resorting to Assumption 1, which is always true in practical terms, both sides of (3.7) can be divided by $r(k+1)$ yielding

$$r(k+1) = 2\frac{\mathbf{u}(k)}{r(k+1)} \cdot \mathbf{z}_1(k) + \frac{\parallel \mathbf{u}(k) \parallel^2}{r(k+1)} z_2(k) + \frac{r(k)}{r(k+1)} r(k). \tag{3.8}$$

Finally, judiciously replacing $r(k)$ by $z_3(k)$ in (3.8) results in

$$z_3(k+1) = 2\frac{\mathbf{u}(k)}{r(k+1)} \cdot \mathbf{z}_1(k) + \frac{\parallel \mathbf{u}(k) \parallel^2}{r(k+1)} z_2(k) + \frac{r(k)}{r(k+1)} z_3(k). \tag{3.9}$$

Now, notice that (3.4) and (3.9) linearly describe time-step transitions of $\mathbf{z}_1(k)$, $z_2(k)$, and $z_3(k)$ since both $\mathbf{u}(k)$ and $r(k)$ can be viewed as system inputs, which are available over time. Also, notice that $z_3(k) := r(k)$ can be viewed as a system measurement. Having this in mind and defining the state vector

$$\mathbf{z}(k) := \begin{bmatrix} \mathbf{z}_1(k) & z_2(k) & z_3(k) \end{bmatrix}^T, \tag{3.10}$$

the system dynamics can be written in the form

$$\begin{cases} \mathbf{z}(k+1) = \mathbf{A}(k)\mathbf{z}(k) \\ y(k+1) = \mathbf{C}\mathbf{z}(k), \end{cases} \tag{3.11}$$

with

$$\mathbf{A}(k) = \begin{bmatrix} \mathbf{I}_3 & \mathbf{u}(k) & \mathbf{0}_{3\times 1} \\ \mathbf{0}_{1\times 3} & 1 & 0 \\ 2\frac{\mathbf{u}^T(k)}{r(k+1)} & \frac{\|\mathbf{u}(k)\|^2}{r(k+1)} & \frac{r(k)}{r(k+1)} \end{bmatrix} \in \mathbb{R}^{5\times 5}$$

and

$$\mathbf{C} = \begin{bmatrix} \mathbf{0}_{1\times 3} & 0 & 1 \end{bmatrix} \in \mathbb{R}^{1\times 5}.$$

3.2 Observability analysis

The two measurements needed to determine the dynamics matrix $\mathbf{A}(k)$ can be viewed as discrete functions of time, since they are assumed to be available at every time instant k. For this reason, the system (3.11) can be seen as a discrete-time linear time-varying system from an observability perspective [17]. Theorem 1 refers to the observability of the system (3.11).

Theorem 1[19]: Assuming that the matrix $\mathbf{L}(k_a)$, defined for time instant $k_a \geq 0$ as

$$\mathbf{L}(k_a) := \begin{bmatrix} \mathbf{L}_0(k_a) \\ \vdots \\ \mathbf{L}_3(k_a) \end{bmatrix} \in \mathbb{R}^{4\times 4},$$

where

$$\mathbf{L}_i(k_a) := \begin{bmatrix} 2\sum_{j=0}^{i} \mathbf{u}(k_a + j) \\ \| \sum_{j=0}^{i} \mathbf{u}(k_a + j) \|^2 \end{bmatrix}^T \in \mathbb{R}^{1\times 4},$$

is a full rank one, i.e.,

$$rank(\mathbf{L}(k_a)) = 4, \tag{3.12}$$

then the initial state $\mathbf{z}(k_a)$ is uniquely determined by the input $\{\ \mathbf{u}(k) : k = k_a, ..., k_a + 4\}$ and the output $\{\ \mathbf{y}(k) : k = k_a, ..., k_a + 4\}$. This means that system (3.11) is observable in the time interval $[k_a, k_a + 5]$.

Nevertheless, showing that system (3.11) is observable is not enough to prove that state estimates of this relate to estimates of the original nonlinear system (3.1), since the state augmentation process required the output $r(k) = x_2(k)||x_1(k)||$ to be discarded and new artificial states to be created. The following theorem proves this relation.

Theorem 2[19]: If (3.12) holds, then :

1. The initial state of the original nonlinear system (3.1), $(\mathbf{x}_1(k_a), x_2(k_a))$, is uniquely determined by the input $\{\ \mathbf{u}(k) : k = k_a, ..., k_a + 4\}$ and the output $\{\ r(k) : k = k_a, ..., k_a + 4\}$. This is the same as saying that the nonlinear system (3.1) is observable in the time interval $[k_a, k_a + 5]$.

2. There is a match between the initial conditions of both the augmented (3.11) and the original (3.1) systems, i.e.,

$$\begin{cases} \mathbf{z}_1(k_a) = x_2^2(k_a)\mathbf{x}_1(k_a) \\ z_2(k_a) = x_2^2(k_a) \\ z_3(k_a) = x_2(k_a)||\mathbf{x}_1(k_a)|| \end{cases} \tag{3.13}$$

The proof of both Theorem 1 and Theorem 2 is given in [19].

3.3 Linear Kalman filter

Under the conclusions obtained in the previous sections, a Kalman filter was employed to perform state estimation on system (3.11). The estimates obtained with this solution will show GES error dynamics if the system (3.11) is shown to be uniformly completely observable [22]. In the previous section it was only shown that the system is observable, but an identical, yet more space consuming, process could be employed to prove uniform complete observability. A brief description of the Kalman filter, as first introduced by R. E. Kalman in [23], is now presented.

Consider a linear system in the form

$$\begin{cases} \mathbf{x}_k = \mathbf{A}_k \mathbf{x}_{k-1} + \mathbf{B}_k \mathbf{u}_k + \mathbf{w}_k \\ \mathbf{y}_{k-1} = \mathbf{C}_{k-1} \mathbf{x}_{k-1} + \mathbf{v}_{k-1}, \end{cases} \tag{3.14}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ represents the system state at time instant k (note the change in notation from parenthesis to subscript as it allows for more compact writing), $\mathbf{A}_k \in \mathbb{R}^{n \times n}$ the system dynamics matrix, and $\mathbf{B}_k \in \mathbb{R}^{n \times n}$ the control matrix for input $\mathbf{u}_k \in \mathbb{R}^n$. Furthermore, a system output, $\mathbf{y}_k \in \mathbb{R}^m$, is considered to be modeled by the observation matrix $\mathbf{C}_k \in \mathbb{R}^{m \times m}$. Additive process and measurement noise is depicted in vectors $\mathbf{w}_k \in \mathbb{R}^n$ and $\mathbf{v}_k \in \mathbb{R}^m$, respectively, which represent zero-mean Gaussian variables with covariance $\mathbf{Q}_k \in \mathbb{R}^{n \times n}$ and $\mathbf{R}_k \in \mathbb{R}^{m \times m}$.

The Kalman filter, unlike other techniques, is able to recursively perform state estimation based solely on the latest available filter information. This translates into computationally inexpensive operations. In each iteration, the filter outputs a state estimate and an error covariance matrix, $\hat{\mathbf{x}}_{r|s}$ and $\mathbf{P}_{r|s}$ respectively. These are represented with the subscript $r|s$, meaning they regard time instant r and are computed with all information up to time instant $s \leq r$. The LKF and, in general, all types of Kalman filters, operates in a two-step process.

The first step predicts the state and error covariance matrix on the next time instant, taking into account the system dynamics and input, as well as the process noise. The algorithm associated to this step is

$$\begin{cases} \hat{\mathbf{x}}_{k|k-1} = \mathbf{A}_k \hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}_k \mathbf{u}_k \\ \mathbf{P}_{k|k-1} = \mathbf{A}_k \mathbf{P}_{k-1|k-1} \mathbf{A}_k^T + \mathbf{Q}_k. \end{cases} \tag{3.15}$$

The correction step makes use of the observed system output to correct the predicted state and error covariance matrix. A measurement residual, $\tilde{\mathbf{y}}_k$, is computed by comparing the observed output value, $\mathbf{y}_k$, with the expected output value when considering the predicted state. Also, the measurement residual covariance, $\mathbf{S}_k$, is computed, allowing for the determination of the Kalman gain, $\mathbf{K}_k$. Having these three quantities, the corrected state estimate and error covariance matrix can be computed, thus ceasing a filter iteration. Mathematically, the correction step is stated as

$$
\begin{cases}
\tilde{\mathbf{y}}_k = \mathbf{y}_k - \mathbf{C}_k \hat{\mathbf{x}}_{k|k-1} \\
\mathbf{S}_k = \mathbf{C}_k \mathbf{P}_{k|k-1} \mathbf{C}_k^T + \mathbf{R}_k \\
\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{C}_k^T \mathbf{S}_k^{-1} \\
\\
\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \tilde{\mathbf{y}}_k \\
\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{C}_k) \mathbf{P}_{k|k-1}.
\end{cases}
\tag{3.16}
$$

It is important to notice that the performance of the LKF is controlled by the initial estimation parameters, $\hat{\mathbf{x}}_0$ and $\mathbf{P}_0$, as well as the assumed additive noise covariance matrices, $\mathbf{Q}_k$ and $\mathbf{R}_k$. The initial state estimate fed by the user ought to be tied to a decent error covariance matrix that demonstrates how confident one is with the initial estimate. The noise parameters are problem dependent and should, ideally, reproduce the true noise characteristics of the system. In the case at hand, the observation noise, $\mathbf{R}_k$, should be modeled the same as the noise of the pseudo-range measurement system. Inversely, the process noise, $\mathbf{Q}_k$, is not as simple to determine as it is related to the acquisition of the system input, which is based in multiple sensor indirect measures. Although mathematical derivations could have been employed to determine the covariance matrices that lead to the best filter performance, an optimization process known as Bayesian optimization was used instead. An overview of this algorithm is presented in Chapter 6.

Finally, recall the augmented system (3.11) for which the LKF provides state estimates. The estimated state $\mathbf{z}(k)$ does not correspond directly to the pursued estimates of the vehicle position, $\mathbf{p}(k)$, and the speed of sound multiplicative coefficient, $v_s(k)$. To obtain those, the following assumptions are needed.

Assumption 2: The speed of sound multiplicative factor, $v_s(k) = x_2(k)$, is comprehended between a maximum and a minimum value, $V_M > 0$ and $V_m > 0$, respectively, as given by

$$
V_m \leq x_2(k) \leq V_M. \tag{3.17}
$$

Assumption 3: The state tied to vehicle position, $\mathbf{x}_1(k)$, is norm-bounded.

Possessing estimates of $\mathbf{z}(k)$, the speed of sound multiplicative coefficient estimates, $\hat{v}_s(k)$, can be determined by

$$
\hat{v}_s(k) = \hat{x}_2(k) = \begin{cases}
V_m, & \hat{z}_2(k) < V_m^2 \\
\sqrt{\hat{z}_2(k)}, & V_m^2 \leq \hat{z}_2(k) \leq V_M^2 \\
V_M, & \hat{z}_2(k) > V_M^2,
\end{cases}
\tag{3.18}
$$

and the vehicle position estimates, $\hat{\mathbf{p}}(k)$, by

$$
\hat{\mathbf{p}}(k) = \hat{\mathbf{x}}_1(k) + \mathbf{s} = \frac{\hat{\mathbf{z}}_1(k)}{\hat{x}_2(k)} + \mathbf{s}. \tag{3.19}
$$

Furthermore, it can be shown that, as estimates of $\mathbf{z}(k)$ have GES error dynamics, then estimates of

speed of sound multiplicative coefficient and position also have GES error dynamics. The proof of this statement is shown in [17, Proposition 1].

Chapter 4

Extended Kalman filter

In order to compare the novel solution to common existing nonlinear estimation approaches, the following sections address the theoretical foundation and the implementation of the extended Kalman filter.

4.1 Theoretical foundation

The EKF adapts the formulation of the LKF to nonlinear systems, such as the one under study in this thesis.

To sum up the derivation present in [24], consider a system of the form

$$\begin{cases} \mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_{k-1} \\ \mathbf{y}_{k-1} = h(\mathbf{x}_{k-1}) + \mathbf{v}_{k-1}, \end{cases} \tag{4.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}^n$ and $h : \mathbb{R}^n \to \mathbb{R}^m$ represent the nonlinear state transition and observation functions, respectively. Also, $\mathbf{u}_k \in \mathbb{R}^n$ represents the system input, and $\mathbf{w}_k \in \mathbb{R}^n$ and $\mathbf{v}_k \in \mathbb{R}^m$ the process and observation noises; the latter two being modeled as zero-mean Gaussian random variables with covariance $\mathbf{Q}_k \in \mathbb{R}^{n\times n}$ and $\mathbf{R}_k \in \mathbb{R}^{m\times m}$, respectively.

As in the Kalman filter, the state estimation process is performed iteratively resorting to two essential steps: prediction and correction. The first corresponds to the simple propagation of the previous estimated state, $\hat{\mathbf{x}}_{k|k}$, and the system input, $\mathbf{u}_k$, through the nonlinear state transition function f. Also, the error covariance matrix, $\mathbf{P}_{k|k-1}$, is updated taking into account the assumed process noise covariance, $\mathbf{Q}_k$. The only difference to the Kalman filter here is that, as there is no dynamics matrix, one must linearize the system dynamics around the latest available state estimate. To achieve this, the Jacobian of the state transition function is computed and used to update the error covariance matrix. Mathematically, the prediction step can be formulated as

$$\begin{cases} \hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}) \\ \mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{Q}_k \end{cases} \tag{4.2}$$

where $\mathbf{F}_k$ is the Jacobian of the state transition function computed using the latest available state estimate and system input. Formally,

$$\mathbf{F}_k = \left.\frac{\partial f}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}}. \tag{4.3}$$

The correction step is also very similar to that found on the Kalman filter, but, again, using some linearization procedures. The algorithm for this step can be stated as

$$\begin{cases} \mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k \\ \mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T \mathbf{S}_k^{-1} \\ \mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_{k|k-1} \\ \hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{y}_k - h(\hat{\mathbf{x}}_{k|k-1})). \end{cases}. \tag{4.4}$$

Here, the $\mathbf{S}_k$ matrix, commonly known as residual covariance, is computed accounting for the observation noise covariance matrix, $\mathbf{R}_k$, and making use of the observation model Jacobian, $\mathbf{H}_k$, evaluated at the predicted state estimate, $\hat{\mathbf{x}}_{k|k-1}$. Formally,

$$\mathbf{H}_k = \left.\frac{\partial h}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_{k|k-1}}. \tag{4.5}$$

Using this linearization, the Kalman gain, $\mathbf{K}_k$, and the error covariance matrix, $\mathbf{P}_{k|k}$, can be determined in the same way as for the LKF. Finally, the new state estimate, $\hat{\mathbf{x}}_{k|k}$, assumes the quantification of the measurement residual, $\mathbf{y}_k - h(\hat{\mathbf{x}}_{k|k-1})$, where $\mathbf{y}_k$ is the observed output of the system at time instant k and $h(\hat{\mathbf{x}}_{k|k-1})$ is the propagation of the predicted state through the observation function.

4.2 Application

Having shown the basis of the EKF, its application to the system (2.4) is now described.

The state, $\mathbf{x}_k = \begin{bmatrix} \mathbf{p}(k)^T & v_s(k) \end{bmatrix}^T$, follows the linear transformation

$$\mathbf{x}_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{k-1} + \mathbf{u}_{k-1}. \tag{4.6}$$

Having this in mind, the determination of $\mathbf{F}_k$ becomes a trivial process as it is directly equal to the 4×4 identity matrix, i.e.,

$$\mathbf{F}_k = I_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \forall k. \tag{4.7}$$

Regarding the observation model, it is described by the nonlinear function

$$h(\mathbf{x}_k) = v_s(k) \parallel \mathbf{s} - \mathbf{p}(k) \parallel, \tag{4.8}$$

which, after applying the derivative operator, as stated in (4.5), leads to the matrix

$$H_k = \begin{bmatrix} -v_{s_{k|k-1}} \frac{2\mathbf{s}^x - 2\mathbf{p}^x_{k|k-1}}{2\sqrt{(\mathbf{s}^x - \mathbf{p}^x_{k|k-1})^2 + (\mathbf{s}^y - \mathbf{p}^y_{k|k-1})^2 + (\mathbf{s}^z - \mathbf{p}^z_{k|k-1})^2}} \\ -v_{s_{k|k-1}} \frac{2\mathbf{s}^y - 2\mathbf{p}^y_{k|k-1}}{2\sqrt{(\mathbf{s}^x - \mathbf{p}^x_{k|k-1})^2 + (\mathbf{s}^y - \mathbf{p}^y_{k|k-1})^2 + (\mathbf{s}^z - \mathbf{p}^z_{k|k-1})^2}} \\ -v_{s_{k|k-1}} \frac{2\mathbf{s}^z - 2\mathbf{p}^z_{k|k-1}}{2\sqrt{(\mathbf{s}^x - \mathbf{p}^x_{k|k-1})^2 + (\mathbf{s}^y - \mathbf{p}^y_{k|k-1})^2 + (\mathbf{s}^z - \mathbf{p}^z_{k|k-1})^2}} \\ \sqrt{(\mathbf{s}^x - \mathbf{p}^x_{k|k-1})^2 + (\mathbf{s}^y - \mathbf{p}^y_{k|k-1})^2 + (\mathbf{s}^z - \mathbf{p}^z_{k|k-1})^2} \end{bmatrix}^T . \tag{4.9}$$

Having identified both matrices $\mathbf{F}_k$ and $\mathbf{H}_k$, the EKF can be implemented. Notice that the initial conditions $\hat{\mathbf{x}}_0$ and $\mathbf{P}_0$, as well as the assumed covariance for both the process noise, $\mathbf{Q}_k$, and observation noise, $\mathbf{R}_k$, will determine the performance of this solution. In parallel to what was done in Chapter 3, both noise covariance matrices were determined through the optimization process described in Chapter 6.

As a final consideration about the EKF and taking into account its foundation shown in this section, one can easily infer that using a simple linearization to solve, in this case, a highly nonlinear problem might not lead to the best possible results. In fact, the EKF works by iteratively propagating state predictions, which correspond to the mean of a Gaussian random variable (GRV), through a nonlinear model and computing error covariances by linearizing the system dynamics. The problem with propagating only the mean of the GRV through the nonlinear function is that, unlike with linear systems, nonlinearities will most likely alter the distribution of the propagated variable to something different than a Gaussian distribution. The EKF does not take this into account.

Having this in mind, the following chapter addresses another solution, which provides a different view on nonlinear state estimation, the unscented Kalman filter.

Chapter 5

Unscented Kalman Filter

As stated at the end of the previous chapter, the unscented Kalman filter addresses system nonlinearities in a more thought-out way, as it makes use of the unscented transform (UT).

The UT relies on the determination of a finite set of points in a Gaussian distribution, sufficient to characterize it in terms of mean and covariance. Propagating these points through a nonlinear system yields a better comprehension of its dynamics. In fact, the UT is able to fully determine the mean and covariance of the posteriori random variable propagated through a nonlinear function up to the third order Taylor series expansion [25]. Oppositely, the EKF can only determine the true mean and covariance up to the first order.

5.1 Theoretical foundation

Regarding the UT, as first introduced in [26], consider $f : \mathbb{R}^n \to \mathbb{R}^m$, a nonlinear function. Its input is the random variable $\mathbf{x} \in \mathbb{R}^n$, with mean $\bar{x}$ and covariance $\mathbf{P}_x$, and its output another random variable, $\mathbf{y} \in \mathbb{R}^m$, such that $\mathbf{y} = f(\mathbf{x})$.

To compute the mean and covariance of $\mathbf{y}$, a set of $2n + 1$ points, commonly known as sigma points (χ_i), is chosen according to the following rules

$$\begin{cases} \chi_0 = \bar{\mathbf{x}} \\ \chi_i = \bar{\mathbf{x}} + (\sqrt{(n+\lambda)\mathbf{P}_x})_i \quad i = 1, ..., n \\ \chi_i = \bar{\mathbf{x}} - (\sqrt{(n+\lambda)\mathbf{P}_x})_i \quad i = n+1, ..., 2n+1. \end{cases} \tag{5.1}$$

The index i after the parenthesis represents the i-th row of the matrix square root, which is usually obtained via the Cholesky decomposition. The scaling parameter, $\lambda = \alpha^2(n + k) - n$, controls the filter performance. A good initial assumption is $\alpha = 10^{-3}$ and $k = 0$ [25].

To each of the sigma points correspond two weights, $W^{(m)}$ and $W^{(c)}$. These are later used to obtain the two pursued parameters, mean and covariance, and can be obtained through

$$\begin{cases} W_0^{(m)} = \frac{\lambda}{n+\lambda} \\ W_0^{(c)} = \frac{\lambda}{n+\lambda} + (1 - \alpha^2 + \beta) \\ W_i^{(m)} = W_i^{(c)} = \frac{1}{2(n+\lambda)} \qquad i = 1, ..., 2n. \end{cases} \tag{5.2}$$

The parameter β is directly related to the type of distribution of the random variable being propagated through f. Considering Gaussian random variables, $\beta = 2$ yields optimal results .

All sigma points are then propagated through the nonlinear function f so that a set of outputs,$\Upsilon_i = f(\chi_i)$, is obtained. Finally, the mean and covariance of $\mathbf{y}$ can be approximated by

$$\begin{cases} \bar{\mathbf{y}} \approx \sum_{i=0}^{2n} W_i^{(m)} \Upsilon_i \\ \mathbf{P_y} \approx \sum_{i=0}^{2n} W_i^{(c)} [\Upsilon_i - \bar{\mathbf{y}}][\Upsilon_i - \bar{\mathbf{y}}]^T. \end{cases} \tag{5.3}$$

Having described the UT, its application to state estimation purposes is now shown. In this work, an augmented state version of the UKF is employed as it is considered to yield better results than the non augmented version [27].

Consider the augmented state

$$\mathbf{x}_k^a = [\mathbf{x}_k^T \mathbf{w}_k^T \mathbf{v}_k^T]^T \in \mathbb{R}^{(n+n+m)}, \tag{5.4}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ represents the original non augmented state (as formulated in Section 4.2) and $\mathbf{w}_k \in \mathbb{R}^n$ together with $\mathbf{v}_k \in \mathbb{R}^m$ account for the process and observation noises, respectively. Also, consider the augmented error covariance matrix

$$\mathbf{P}_k^a = \begin{bmatrix} \mathbf{P}_k & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R}_k \end{bmatrix} \in \mathbb{R}^{(n+n+m)\times(n+n+m)}, \tag{5.5}$$

which includes the process noise covariance matrix, $\mathbf{Q}_k \in \mathbb{R}^{n\times n}$, and the observation noise covariance matrix, $\mathbf{R}_k \in \mathbb{R}^{m\times m}$. The algorithm starts with the computation of the sigma points using (5.1). Note that, with state augmentation, the number of sigma points increases from $2n + 1$ to $2(n + n + m) + 1$. Each sigma point is composed of 3 parts, the first being related to the original non augmented state and the other two to the noise variables. This is,

$$\chi_k^a = [(\chi_k^x)^T (\chi_k^w)^T (\chi_k^v)^T]^T. \tag{5.6}$$

Having the sigma points determined, the prediction step starts by propagating their process dynamics components, (χ^x) and (χ^w), together with the system input, $\mathbf{u}_k$, through the nonlinear state transition function f. Then, the mean and covariance of the resulting set of outputs is determined as in (5.3). Formally,

$$\begin{cases} \chi^x_{k|k-1} = f(\chi^x_{k-1|k-1}, \chi^w_{k-1|k-1}, u_{k-1|k-1}) \\ \hat{\mathbf{x}}_{k|k-1} = \sum_{i=0}^{2n} W_i^{(m)} \chi^x_{i,k|k-1} \\ \mathbf{P}_{k|k-1} = \sum_{i=0}^{2n} W_i^{(c)} [\chi^x_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}][\chi^x_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}]^T. \end{cases} \tag{5.7}$$

Then, the correction step begins with the propagation of χ^x_k and χ^v_k through the observation function. Subsequently, the statistical parameters on the output set are determined as in the prediction step. This time, however, a crossed covariance matrix, $\mathbf{P}_{\mathbf{x}_k\mathbf{y}_k}$, relating the predicted state and observation is also computed. Finally, an "unscented version" of the Kalman gain, $\mathbf{K}_k$, as well as the new state estimate, $\hat{\mathbf{x}}_{k|k}$, and its error covariance matrix, $\mathbf{P}_{k|k}$, are computed as in the EKF. Mathematically,

$$\begin{cases} \boldsymbol{\Upsilon}_{\mathbf{k|k-1}} = h(\chi^x_{k-1|k-1}, \chi^v_{k-1|k-1}) \\ \hat{\mathbf{y}}_{k|k-1} = \sum_{i=0}^{2n} W_i^{(m)} \Upsilon_{i,k|k-1} \\ \mathbf{P}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k} = \sum_{i=0}^{2n} W_i^{(c)} [\Upsilon^x_{i,k|k-1} - \hat{\mathbf{y}}_{k|k-1}][\Upsilon^y_{i,k|k-1} - \hat{\mathbf{y}}_{k|k-1}]^T \\ \mathbf{P}_{\mathbf{x}_k\mathbf{y}_k} = \sum_{i=0}^{2n} W_i^{(c)} [\chi^x_{i,k|k-1} - \hat{\mathbf{x}}_{k|k-1}][\Upsilon^y_{i,k|k-1} - \hat{\mathbf{y}}_{k|k-1}]^T \\ \mathbf{K} = \mathbf{P}_{\mathbf{x}_k\mathbf{y}_k} \mathbf{P}^{-1}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k} \\ \hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}(\mathbf{y}_{k|k} - \mathbf{y}_{k|k-1}) \\ \mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}\mathbf{P}_{\tilde{\mathbf{y}}_k\tilde{\mathbf{y}}_k}\mathbf{K}^T. \end{cases} \tag{5.8}$$

Note that the estimated state, $\hat{\mathbf{x}}_{k|k}$, and covariance matrix, $\mathbf{P}_{k|k}$, are non augmented variables, meaning that, when starting a new iteration of this filtering solution, they must be used to perform state and covariance matrix augmentation with (5.4) and (5.5).

Chapter 6

Bayesian optimization

The Bayesian optimization algorithm is commonly used in situations where a set of parameters controls the performance of a hard-to-evaluate system that can be viewed as a black box. In other words, a computationally expensive system on which there is close to no intuition on what a change on either parameter might lead to. This is not completely the case with the problem at hand, since Kalman filters are not computationally expensive algorithms and, as said in Section 3.3, there is some intuition on how noise covariance matrices should be modeled. Nevertheless, the greatest advantage of the BO algorithm and the reason why it was selected to perform parameter optimization is that, unlike "brute force" methods which unintelligently sweep lots of parameter combinations to output the best one, Bayesian optimization works according to reason. The BO algorithm is capable of iteratively harvest information from previous parameter combinations and their performance to choose a new set of parameters to test. This saves computational time by reducing the number of tests needed. Another remark on the Bayesian optimization algorithm is that it is subject to no human intuition in the sense that it cannot be associated to the belief of the person performing the optimization, unlike manual procedures normally are.

A more in-depth look into the BO algorithm is given below.

6.1 Algorithm

In short, the BO algorithm is a global minimization algorithm that works by creating a surrogate model [28], SM, of a scalar function, $f(\mathbf{x})$, being subject to minimization. To create the surrogate model, the algorithm uses all parameter combinations, $\mathbf{x}$, tested until each new iteration, and their outputs, $y = f(\mathbf{x})$. This surrogate model is then analyzed through the function, $a(\mathbf{x}|SM)$, commonly known as acquisition function, in order to determine the next point to be tested. To better describe the BO algorithm some notions on Gaussian processes and acquisition functions are given.

To update the surrogate model, a stochastic process (thus the Bayesian term) known as Gaussian process is often used. This is defined by two parameters: a mean, $m(\mathbf{x})$, and a kernel, $k(\mathbf{x}, \mathbf{x}')$. Regarding the kernel function, available literature considers various options, being the Matérn 5/2 kernel the one chosen for this purpose as it tends to be better suited for BO algorithms [29].

The maximization of the acquisition function yields the next set of parameters to test in each iteration. Various acquisition functions are also foreseen in the literature, being the expected-improvement (EI) acquisition function [30] the one chosen for its proven good performance when compared to other methods [31]. Another good advantage of the EI acquisition function is its explicit control of exploitation versus exploration of new solutions through a single input parameter. Algorithm 1 sums up the BO process.

Algorithm 1: Bayesian Optimization Algorithm

input : Number of iterations
output: Best parameter set

```
// Choose random parameter set
x_1 ← rand(χ);
// Compute system output and store parameter input and output observation to
   OBS
y_1 = f(x_1);
OBS = {x_1, y_1};
// Create surrogate model from observations
SM ← OBS;
for i ← 2 to Number of iterations do
    // Maximize acquisition function to choose new test set
    x_i = arg max a(x|SM);
    // Compute system ouput and append parameter input and output observation to
       OBS
    y_i = f(x_i);
    OBS = append(OBS, {x_i, y_i});
    // Update the surrogate model
    SM ← OBS;
end
return x_i corresponding to minimum y_i
```

Regarding the application to Kalman filters, parameter optimization was employed to obtain the noise covariance matrices $\mathbf{R}_k$ and $\mathbf{Q}_k$ which lead to the best filter convergence rate and steady-state behavior. Note that, in general, optimizing convergence rate and steady-state behavior at the same time is not possible. This lead to some compromises between the two performance components when applying this optimization process to the filter inputs.

The parameter optimization results are expressed in Chapters 8 and 9, where they were used to feed the three filtering solutions proposed in this thesis.

Chapter 7

Bayesian Cramér-Rao bound

The Bayesian Cramér-Rao bound is an useful tool employed to evaluate the performance of an estimator. It is used in this work to provide a lower bound for the error covariance matrix of an unbiased estimator designed for a system with linear process, but nonlinear output, and corrupted by additive white Gaussian noise. Since it also addresses a variety of other nonlinear systems besides the aforementioned ones, the BCRB becomes very important in nonlinear estimation, as optimal solutions are still a subject of study.

Consider the system

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{F}(k)\mathbf{x}(k) + \mathbf{B}(k)\mathbf{u}(k) + \mathbf{n}_{\mathbf{x}}(k) \\ \mathbf{y}(k) = \mathbf{h}(\mathbf{x}(k)) + \mathbf{n}_{\mathbf{y}}(k), \end{cases} \tag{7.1}$$

where $\mathbf{F}(k)$ and $\mathbf{B}(k)$ represent the typical process matrices, $\mathbf{x}(k)$ is the state vector, $\mathbf{u}(k)$ the system input, and $\mathbf{h}(\mathbf{x}(k))$ the nonlinear function that modulates the output $\mathbf{y}(k)$. The system is also considered to be corrupted by additive noise, here represented by $\mathbf{n}_{\mathbf{x}}(k)$ and $\mathbf{n}_{\mathbf{y}}(k)$, which both follow a zero-mean Gaussian distribution with covariance $\mathbf{Q}_{\mathbf{x}}(k)$ and $\mathbf{Q}_{\mathbf{y}}(k)$, respectively.

The BCRB, $\mathbf{P}_L(k)$, is obtained through a recursion similar to that of the EKF, that also resorts to Jacobian matrices [32]. The major difference to the EKF is that Jacobian matrices are evaluated at the true state, rather than at the estimates of the state. Formally, the BCRB is

$$\mathbf{P}_L(k) = \mathbf{J}^{-1}(k), \tag{7.2}$$

where $\mathbf{J}(k)$ follows

$$\mathbf{J}(k+1) = \left[\mathbf{Q}_{\mathbf{x}}(k) + \mathbf{F}(k)\mathbf{J}^{-1}(k)\mathbf{F}^T(k)\right]^{-1} + E_{\mathbf{x}(k+1)}\left\{\tilde{\mathbf{H}}^T(\mathbf{x}(k+1))\mathbf{Q}_y^{-1}(k+1)\tilde{\mathbf{H}}(\mathbf{x}(k+1))\right\}. \tag{7.3}$$

The second part of this sum accounts for the observations and its impact on the error covariance. It implies the computation of $\tilde{\mathbf{H}}(\mathbf{x}(k+1))$, the Jacobian of the nonlinear observation function, h, evaluated at the true state $\mathbf{x}(k+1)$, and the expected value of the product $\tilde{\mathbf{H}}^T(\mathbf{x}(k+1))\mathbf{Q}_y^{-1}(k+1)\tilde{\mathbf{H}}(\mathbf{x}(k+1))$, typically computed by means of Monte Carlo simulations. Since this thesis studies the performance of the filtering solutions along a specific nominal trajectory, $\bar{\mathbf{x}}(k)$, Monte Carlo simulations are not needed

to determine the BCRB and (7.3) can be reduced to

$$\mathbf{J}(k+1) = \left[\mathbf{Q}_{\mathbf{x}}(k) + \mathbf{F}(k)\mathbf{J}^{-1}(k)\mathbf{F}^{T}(k)\right]^{-1} + \tilde{\mathbf{H}}^{T}(\bar{\mathbf{x}}(k+1))\mathbf{Q}_{y}^{-1}(k+1)\tilde{\mathbf{H}}(\bar{\mathbf{x}}(k+1)). \tag{7.4}$$

In order to apply this method to the problem under study, the matrix $\mathbf{P}_L$ should be initialized with the initial error covariance matrix used to feed the filter, $\mathbf{P}_0$. Also, notice that the system (2.4) can be written in the form expressed in (7.1), which implies that the matrices $\mathbf{F}(k)$ and $\mathbf{H}(\bar{\mathbf{x}}(k+1))$ are those computed in (4.3) and (4.5), respectively. The matrices $\mathbf{Q}_x(k)$ and $\mathbf{Q}_y(k)$ should be set with the covariance of the noise that is assumed to model state and observation transitions. Monte Carlo simulations provide error standard deviations of each filtering solution, which, in turn, can be compared to the square root of the diagonal of the BCRB, yielding further insights on the filtering performance.

Chapter 8

Simulations

Having provided, in the previous chapters, a basic description of the principles of each adopted solution, Chapter 8 pretends to demonstrate and compare the results obtained by applying said solutions. Results of each filtering solution were obtained through simulations run in the Matlab™ environment.

First, the simulation configuration is presented. Then, the results obtained for each proposed solution are shown and discussed. Finally, further comprehension of the filtering performance is achieved by performing Monte Carlo simulations.

8.1 Configuration

The setup of the simulations consisted in the determination of a trajectory to be followed by the vehicle as well as the creation of the measurements needed to implement either of the three filters.

8.1.1 Trajectory

The trajectory described by the vehicle was created by determining a set of waypoints in the three-dimensional space. The aforementioned set was obtained by consecutively adding inertial position variations to the trajectory already described. In order to avoid valueless results of the filtering solutions, a rich trajectory, i.e., one without straight lines and lack of changes in direction, was preferred. The variation of each position component $(\Delta\mathbf{p}^x, \Delta\mathbf{p}^y, \Delta\mathbf{p}^z)$, in meters, was modeled as

$$\begin{cases} \Delta\mathbf{p}^x(k) = cos\left(\frac{2\pi}{30}kT\right) \\ \Delta\mathbf{p}^y(k) = cos\left(\frac{\pi}{10}kT + \frac{\pi}{6}\right) \\ \Delta\mathbf{p}^z(k) = cos\left(\frac{2\pi}{45}kT + \frac{\pi}{9}\right), \end{cases} \tag{8.1}$$

where $k \in \mathbb{N}$ represents the time step and T the update rate used to compute the position variations. Assuming new positions are determined at a frequency of 1 Hz, it follows that $T = 1$s. Each trajectory was limited to 4000 seconds (approximately one hour). The initial inertial position of the vehicle was set to the origin of the inertial reference frame, i.e., $\bar{\mathbf{p}}_0 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$. A tree-dimensional view of the

trajectory is presented in Fig. 8.1. Note that the method used to create the trajectory resembles the

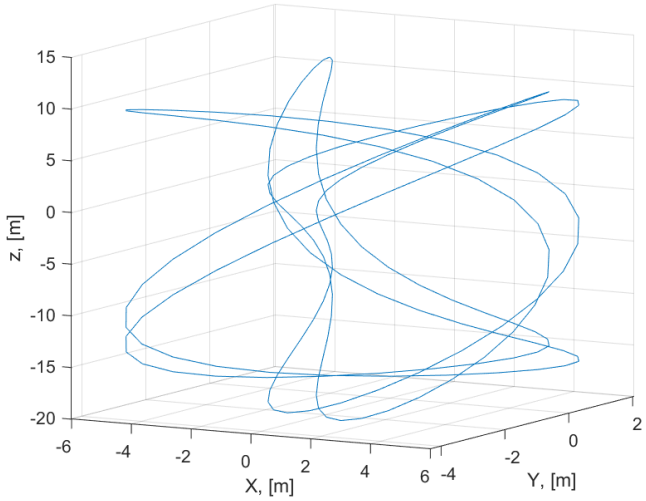

Figure 8.1: Vehicle trajectory

first equation of the system dynamics presented in (2.4), where a new position, $\mathbf{p}(k+1)$, is assumed to be equal to the last one, $\mathbf{p}(k)$, plus a variation, $\mathbf{u}(k)$. This will become helpful when creating the measurements and comparing the estimated values with the real trajectory.

8.1.2 Measurements

Recall the discrete-time system that mathematically formulates the navigation problem

$$\begin{cases} \mathbf{p}(t_{k+1}) = \mathbf{p}(t_k) + \mathbf{u}(t_k) \\ v_s(t_{k+1}) = v_s(t_k) \\ r(k) = v_s(t_k) \parallel \mathbf{s} - \mathbf{p}(t_k) \parallel, \end{cases} \tag{8.2}$$

where

$$\mathbf{u}(t_k) = \int_{t_k}^{t_{k+1}} \mathbf{R}(\tau)\mathbf{v}(\tau)d\tau. \tag{8.3}$$

Regarding this formulation, two quantities are needed to implement the estimation processes, the pseudo-ranges to the transponder, $r(k)$, and the position variation integral, $\mathbf{u}(t_k)$.

The first quantity was obtained by computing the euclidean distance between the true vehicle position, $\bar{\mathbf{p}}(t_k)$, and the position of the fixed beacon, $\mathbf{s}$, and then multiplying it by the coefficient $v_s(t_k)$. The latter was initialized with a value of 1.1, and, although it is not expressed in (8.2), its evolution was assumed to be corrupted by additive zero-mean white Gaussian noise with a standard deviation of 0.01 for BCRB determination purposes. This difference concerns the fact that (8.2) is a deterministic model, whereas in practice one aims to be able to estimate slowly time-varying parameters. Formally,

$$r(t_k) = v_s(t_k)\sqrt{(s^x - \bar{\mathbf{p}}^x(t_k))^2 + (s^y - \bar{\mathbf{p}}^y(t_k))^2 + (s^z - \bar{\mathbf{p}}^z(t_k))^2}. \tag{8.4}$$

This quantity is assumed to be available at a frequency of 1 Hz, the same frequency used to create the trajectory waypoints.

The position variation integral, $\mathbf{u}(k)$, was assumed as equal to the position variations used to determine the trajectory of the vehicle, $\Delta_{\mathbf{p}}(k)$. This implies that, with the exception of measurement noise, the integral would be perfectly determined by the sensors on board the vehicle. Estimation results assuming the computation of the position variation integral through measured rotation matrices, $\mathbf{R}(t)$, and vehicle velocities, $\mathbf{v}(t)$, are later addressed in Chapter 9.

Finally, both measurements were assumed to be corrupted with additive zero-mean white Gaussian noise with the standard deviations present in Table 8.1. Having generated both the position variations, $\mathbf{u}(k)$, and the pseudo-range measurements, $r(k)$, any of the three filtering solutions can be materialized.

Measure	Standard deviation [m]
$r(k)$	10^{-2}
$\mathbf{u}(k)$	5×10^{-2}

Table 8.1: Standard deviation of the noise

8.2 Linear Kalman filter

Concerning the LKF, the initial estimation parameters $\hat{\mathbf{x}}_0$ and $\mathbf{P}_0$ were arbitrarily chosen to exemplify the goodness of the solution. Furthermore, noise covariance matrices, $\mathbf{Q}_k$ and $\mathbf{R}_k$, were obtained via the BO algorithm presented in Chapter 6. All the parameters, except for the initial state estimate, are portrayed in Table 8.2.

Parameter	Value
$\mathbf{P}_0$	I_5
$\mathbf{Q}_k$	$diag(0.20655\mathbf{I}_3, 6.4659 \times 10^{-5}, 0.87563)$
$\mathbf{R}_k$	0.5332

Table 8.2: LKF parameters

Two initial state estimate vectors, $\hat{\mathbf{x}}_0 = \begin{bmatrix} 0 & 6 & 8 & 1.0 & 5 \end{bmatrix}^T$ and $\hat{\mathbf{x}}_0 = \begin{bmatrix} 100 & 141.42 & 141.42 & 1.0 & 100 \end{bmatrix}^T$, were used to feed the filter. The first corresponds to an initial estimate where the norm of the position of the vehicle is incorrect by 10 meters, the sound speed multiplicative factor by 0.1, and the range by 5 meters. The second vector ”pushes” the initial guess even farther away from the true state by having the position of the vehicle misjudged by approximately 200 meters and range measurement by 100 meters, while keeping the same speed of sound multiplicative coefficient. The initial convergence of the position and speed of sound coefficient errors, as well as steady-state performance of the LKF are shown in Figures 8.2 and 8.3. As can be seen, the novel solution is able to lead, in both cases, the estimation errors to zero quite rapidly, while showing a good steady-state performance, with the norm of position error being consistently below 0.5 meters. Due to space limitations, no more results regarding even

higher initial condition errors are shown. Nevertheless, it was observed that, for the noise properties assumed and within the considered simulation length, error convergence was guaranteed up to initial absolute position errors of more than 1000Km.

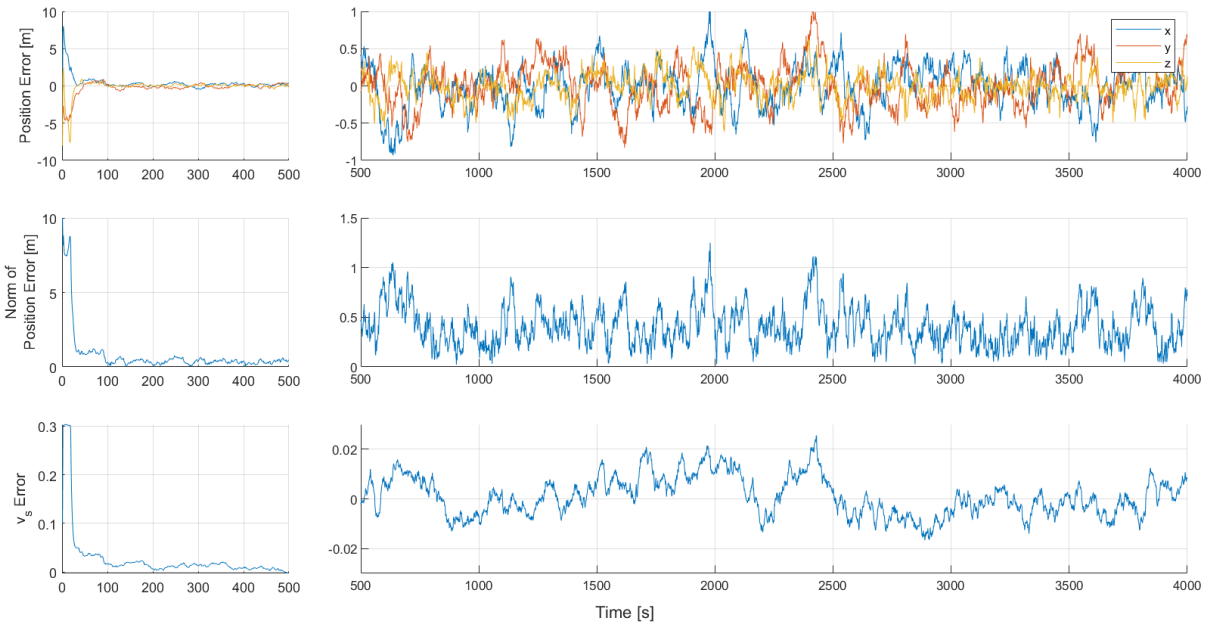

Figure 8.2: LKF state estimation error with low initial error

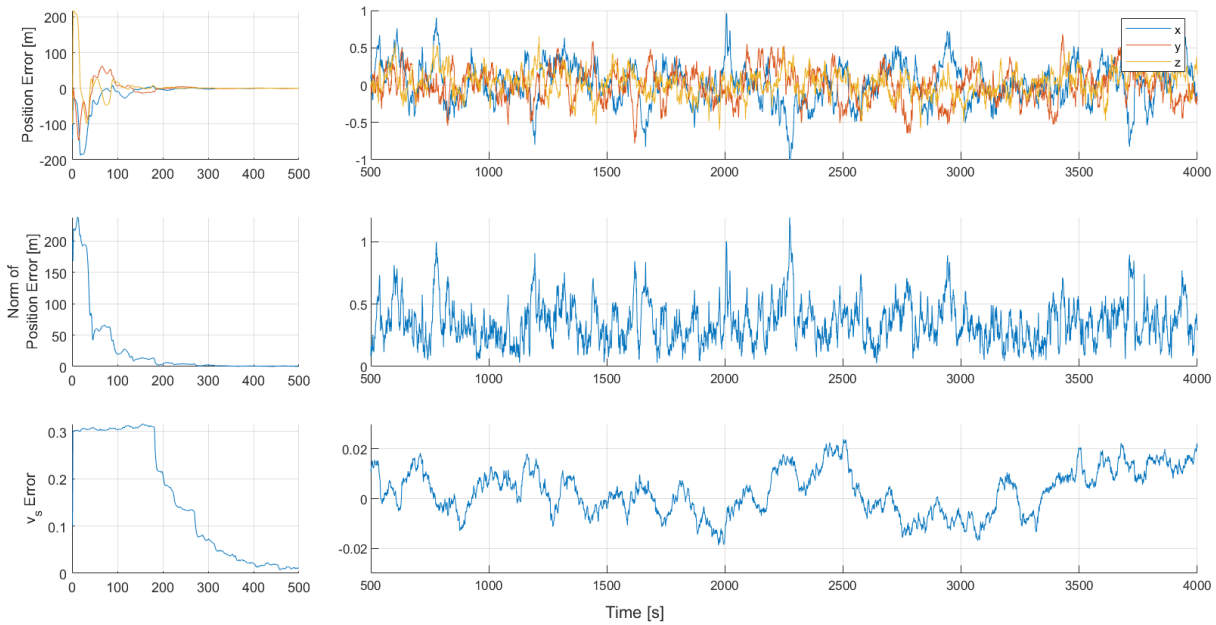

Figure 8.3: LKF state estimation error with high initial error

8.3 Extended Kalman filter

Similarly to what was done in the previous section, two initial conditions were used to feed the EKF. The tuning of the parameters was also performed resorting to the BO algorithm, being the results of this process shown in Table 8.3, along with the initial error covariance matrix.

Parameter	Value
$\mathbf{P}_0$	$\mathbf{I}_4$
$\mathbf{Q}_k$	$diag(0.98339\mathbf{I}_3, 0.00015631)$
$\mathbf{R}_k$	0.99992

Table 8.3: EKF parameters

Figures 8.4 and 8.5 show two distinct results of the EKF. In the first set of results, the initial state estimate was set to $\hat{\mathbf{x}}_0 = \begin{bmatrix} 0 & 6 & 8 & 1.0 \end{bmatrix}^T$. This represents an initial error of 10 meters in terms of position estimate and 0.1 in terms of the unknown coefficient estimate. As can be inferred, these initial conditions lead to a result fairly similar to that of the LKF, since both initial convergence and steady-state performance are comparable. Oppositely, a divergence in the state estimation process can be found in the second set of figures. This was a result of changing the norm of the initial position error to 30 meters, while maintaining the speed of sound multiplicative coefficient error at 0.1. The initial assumption that lead to the diverging result was $\hat{\mathbf{x}}_0 = \begin{bmatrix} 20 & 20 & 10 & 1.0 \end{bmatrix}^T$.

As demonstrated in this section, the EKF is a valid solution to solve the state estimation required for the navigation problem, since it yields converging results, although these can only be obtained when setting the initial state estimate close to the true initial state. Therefore, the EKF fails to provide global convergence guarantees.

8.4 Unscented Kalman filter

The results of state estimation performed by the UKF are shown in this section. Similarly to what was done for the LKF and the EKF, the process and measurement noise covariance matrices were determined resorting to the Bayesian optimization process. The parameters used to feed the UKF are presented in Table 8.4.

Parameter	Value
$\mathbf{P}_0$	$\mathbf{I}_4$
$\mathbf{Q}_k$	$diag(0.032846\mathbf{I}_3, 0.0052031)$
$\mathbf{R}_k$	0.98731
α	0.001
k	0
β	2

Table 8.4: UKF parameters

As in the previous section, two sets of figures represent two distinct results obtained when using

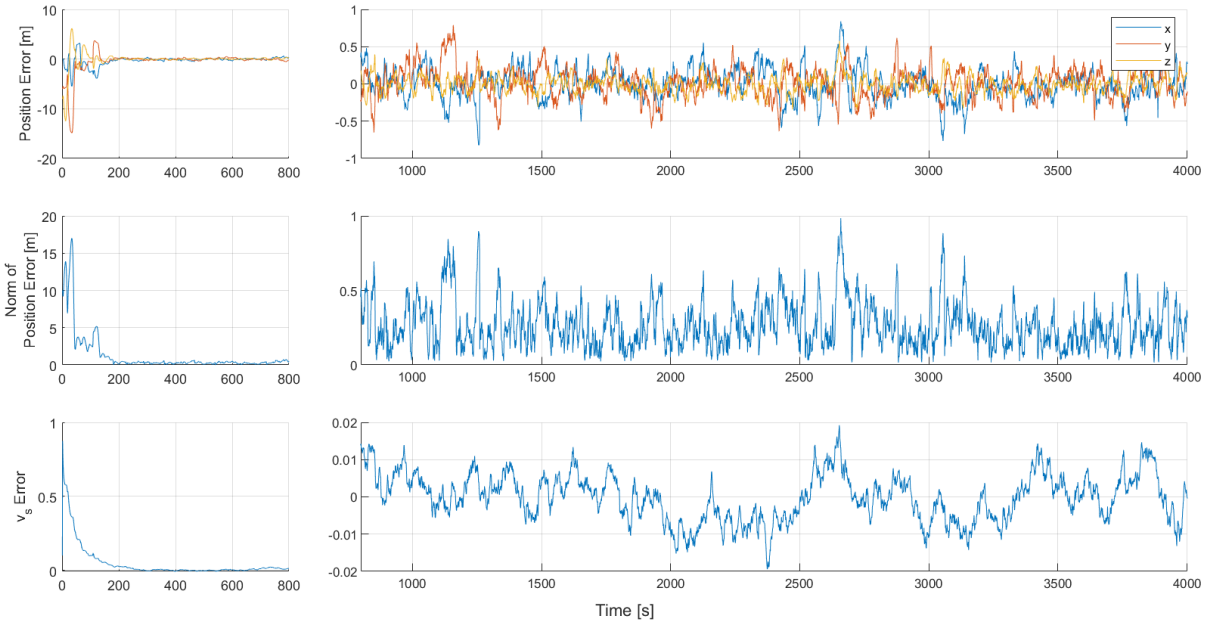

Figure 8.4: EKF state estimation error convergence

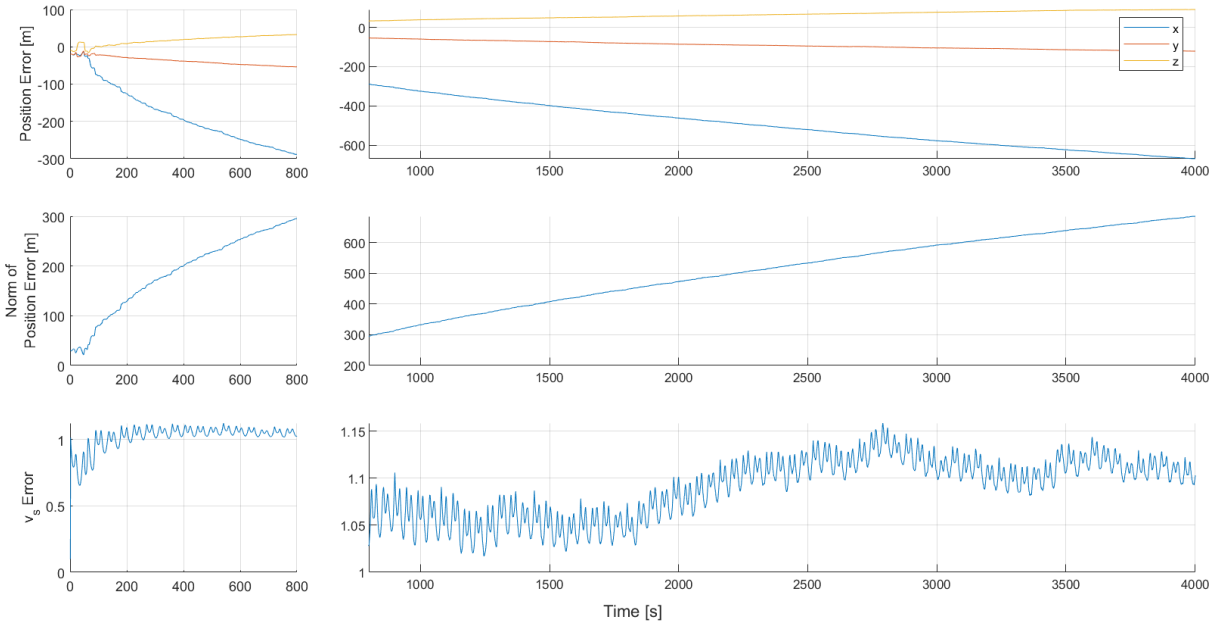

Figure 8.5: EKF state estimation error divergence

the UKF. Firstly, the initial condition which ultimately lead to diverging state estimate errors when using the EKF, $\hat{\mathbf{x}}_0 = \begin{bmatrix} 20 & 20 & 10 & 1.0 \end{bmatrix}^T$, now results in a valid output as depicted in Fig. 8.6. This confirms that the UKF exhibits better performance than the EKF. Notice, however, that the transients take longer, increasing by a factor of ten when compared to the transients of the EKF. Secondly, Fig. 8.7 shows the results obtained after simulating the filter with the initial condition $\hat{\mathbf{x}}_0 = \begin{bmatrix} 100 & 141.42 & 141.42 & 1.0 \end{bmatrix}^T$. This time, although the UKF manages to attain error convergence, this happens at a very slow rate. In fact,

convergence was only achieved after extending the simulation length to 100000 seconds (approximately 27 hours). Although this result does not depict a diverging estimation process, it shows that the UKF is rendered useless for some initial filter conditions.

To sum up, the UKF, like the EKF, is able to obtain valid estimation results, but lacks the capability to offer, in an acceptable time span, convergence guarantees for all initial conditions.

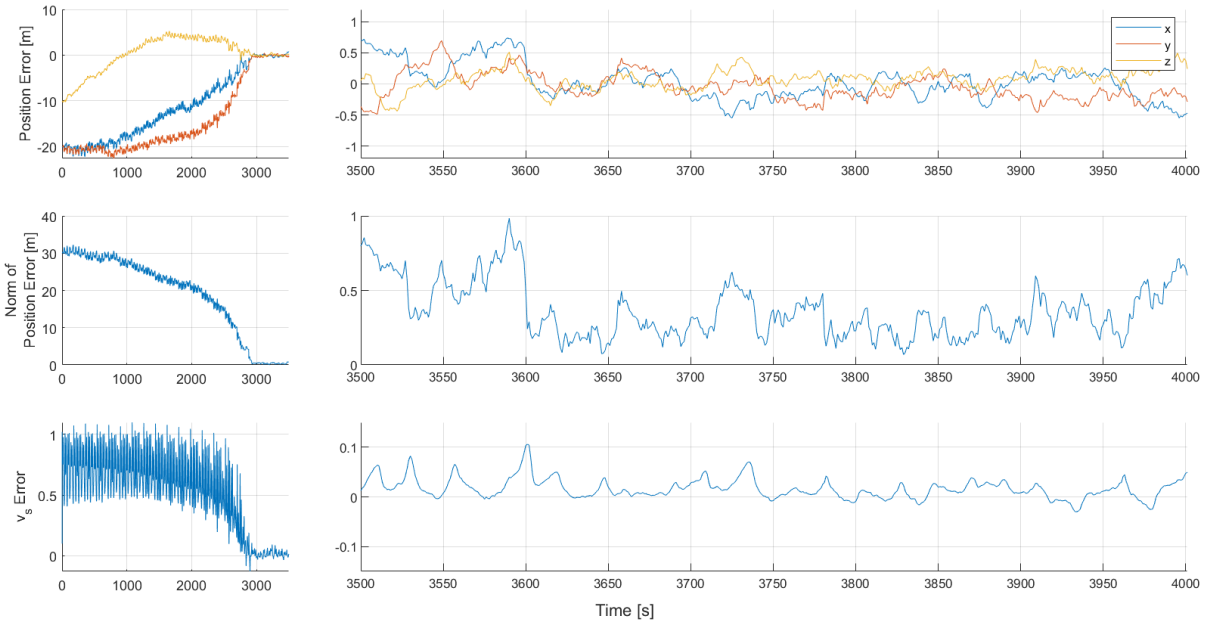

Figure 8.6: UKF state estimation error convergence

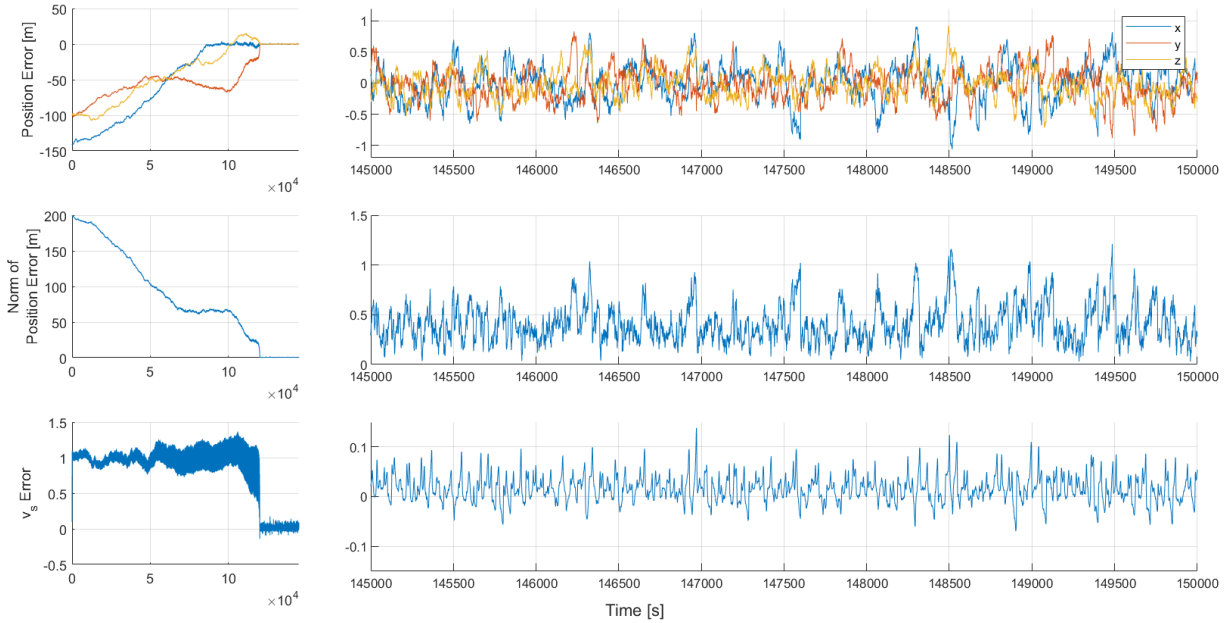

Figure 8.7: UKF state estimation error slow convergence

8.5 Monte Carlo Simulations

To better determine the performance of the filtering solutions, Monte Carlo simulations were performed. This method is useful to detect possible filter bias. Also, Monte Carlo simulations allow for the determination of error standard deviations which, when compared to the BCRB, grant further insights of filter performance.

Each simulation comprised its own randomly generated noise signal and initial state, the later being portrayed by a Gaussian distribution centered about the true initial state and with covariance equal to the initial error covariance of the filter, $\mathbf{P}_0$. Covariance matrices used for each Kalman filter are expressed in Table 8.5. Note, in particular, that the variation allowed for the initial unknown coefficient, $\sigma^2_{v_S} = 1$, is

	LKF	EKF	UKF
Error Covariance	$\mathbf{I}_5$	$\mathbf{I}_4$	$\mathbf{I}_4$

Table 8.5: Initial state error covariance in the Monte Carlo simulations

a very liberal one. This makes for a very broad-scenario of initial v_s conditions when performing Monte Carlo simulations. A total of 1000 runs of each filtering process were performed.

The evolution of the mean estimation error of each state when using all three solutions is shown in Figures 8.8, 8.9, and 8.10. Further analysis of the mean error was performed by computing its steady-

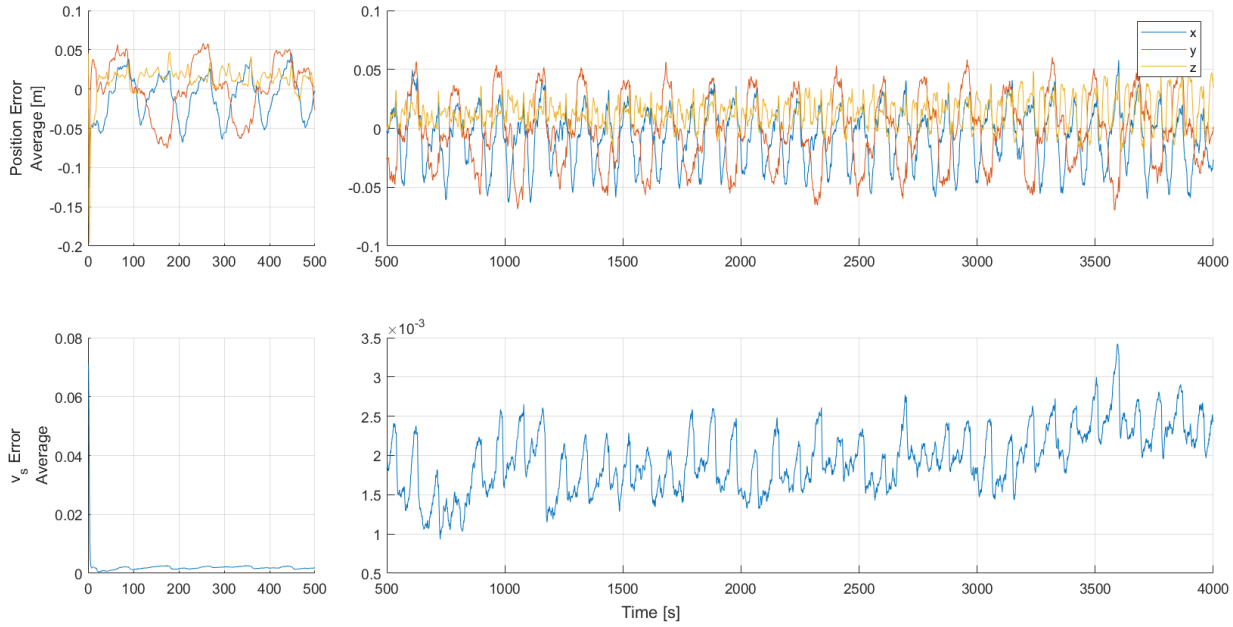

Figure 8.8: LKF Monte Carlo estimation average error

state ($t > 500s$) average. Results are shown in Table 8.6. All three filters show some bias, especially in the third position component, $\mathbf{p}^z$, and in the speed of sound multiplicative coefficient, v_s.

Finally, the standard deviations of the estimation errors were determined and compared to the BCRB. Fig. 8.11 shows the evolution of the standard deviations over time, focusing both on transient and steady-

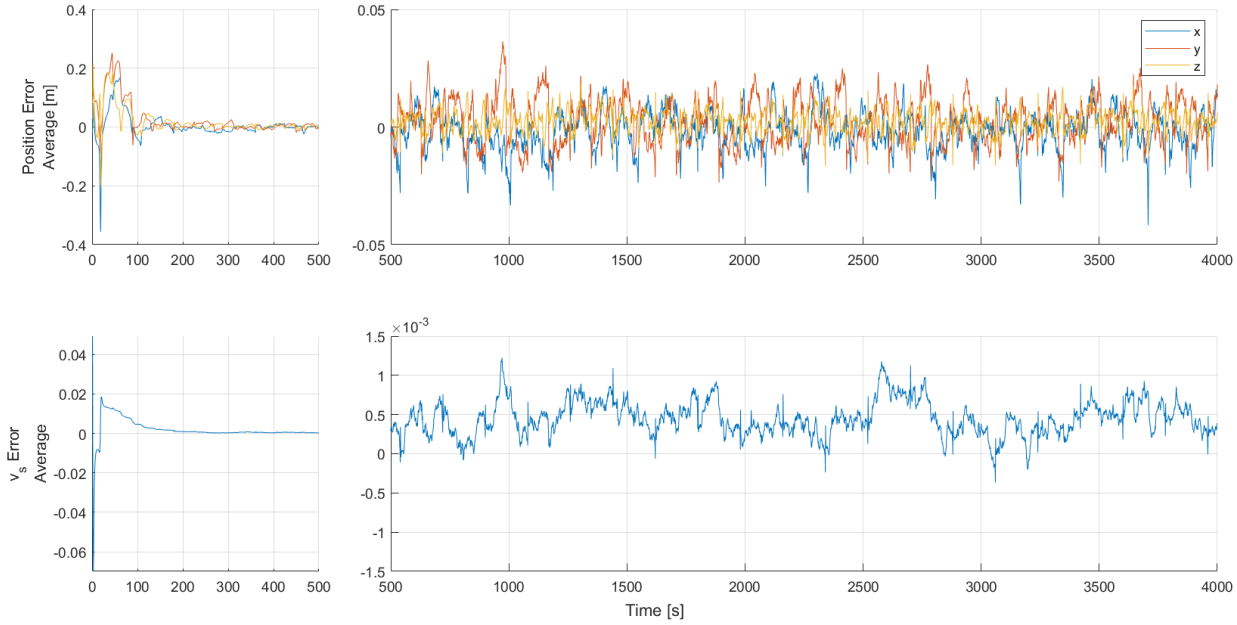

Figure 8.9: EKF Monte Carlo estimation average error

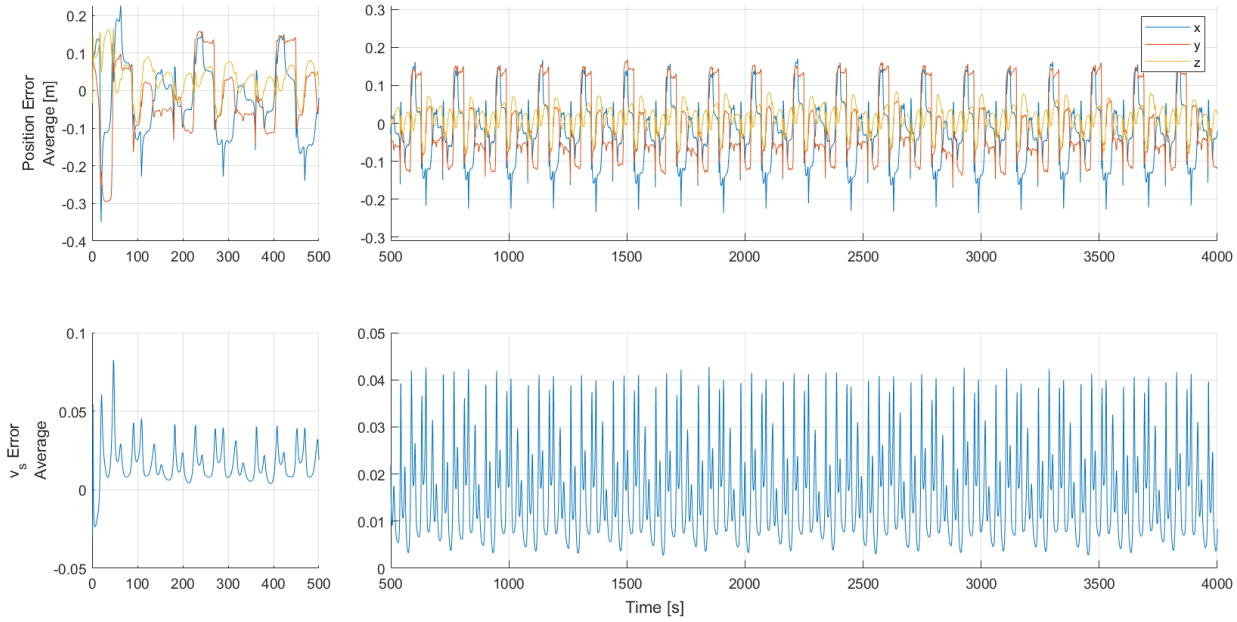

Figure 8.10: UKF Monte Carlo estimation average error

Filter	$\mathbf{p}^x[m]$	$\mathbf{p}^y[m]$	$\mathbf{p}^z[m]$	v_s
LKF	-0.0072	-0.0035	0.0133	1.9×10^{-3}
EKF	-0.0020	0.0029	0.0024	4.5×10^{-4}
UKF	-0.0266	-0.0072	0.0131	1.51×10^{-2}

Table 8.6: Monte Carlo steady-state estimation average error

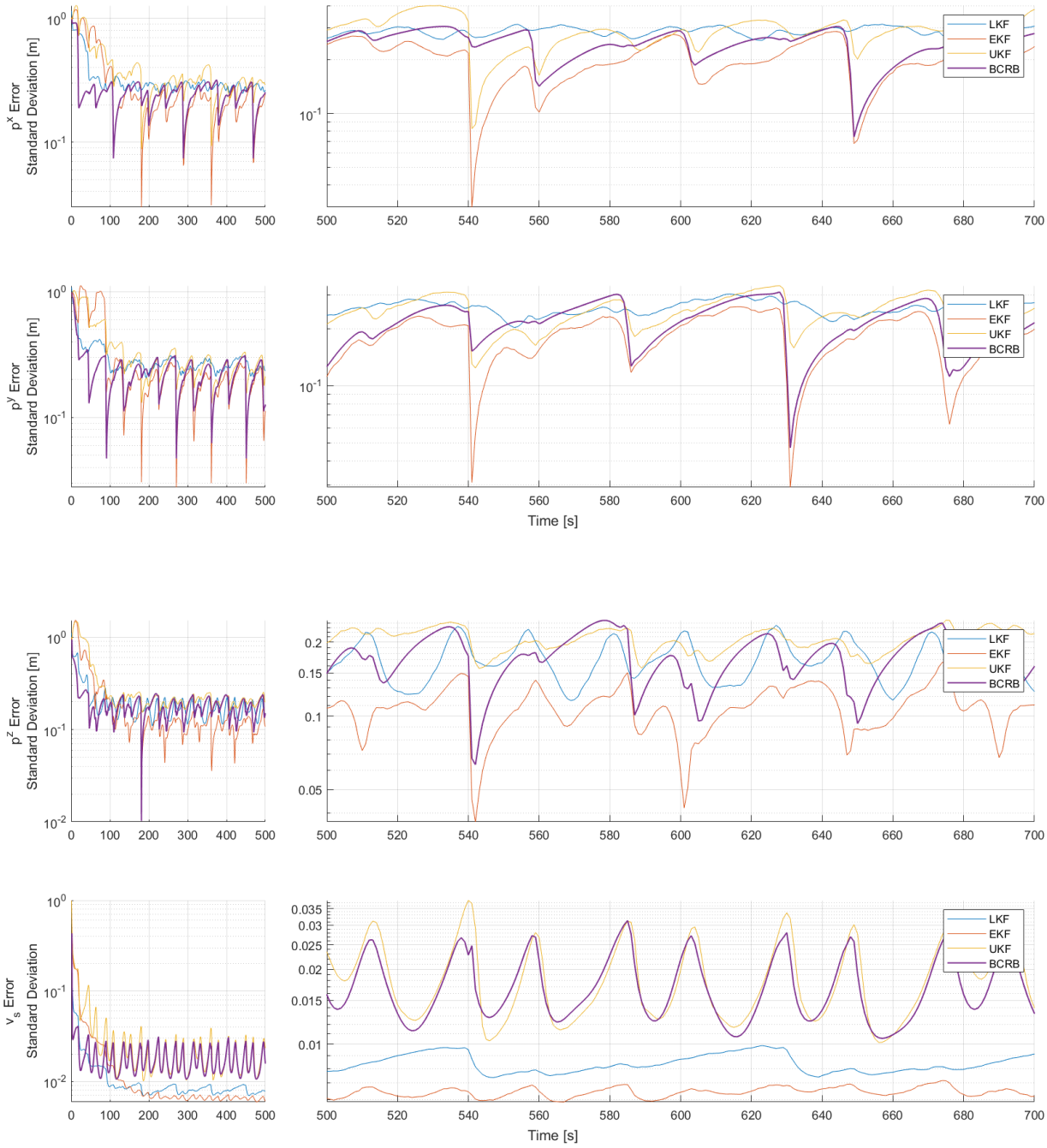

Figure 8.11: Monte Carlo standard deviation of the estimation error

state performance. Notice that the evolution, over time, of each curve demonstrates a certain periodicity. This is due to the fact that the trajectory used to compute these simulations also has a periodical nature, as stated before. The ”valleys” of the curves, where the standard deviations decrease considerably, are also deeply related to shape of the trajectory in the sense that, in those time intervals, the inertial variations of the position of the vehicle are more pronounced affecting the observability of the system. It is clear that all filters perform fairly close to the lower bound provided by the BCRB. Going into further detail, it can be seen that the EKF tends to consistently over-perform the BCRB. Also, for the third

position component, $\mathbf{p}^z$, and for v_s, the standard deviation of the LKF is often below the BCRB. The same conclusions can be drawn from Table 8.7, where steady-state error standard deviations appear averaged. These odd observations are explained by the biased nature of the filters. Since the BCRB only sets a lower limit on the error standard deviations for unbiased estimators, a biased estimator is allowed to achieve such results.

Filter	$\mathbf{p}^x[m]$	$\mathbf{p}^y[m]$	$\mathbf{p}^z[m]$	v_s
LKF	0.2864	0.2595	0.1796	0.0088
EKF	0.1992	0.1842	0.1091	0.0065
UKF	0.2908	0.2503	0.2083	0.0189
BCRB	0.2392	0.2165	0.1757	0.0173

Table 8.7: Monte Carlo steady-state averaged standard deviation of the estimation error

Chapter 9

Experiments

Experiments were performed to further test the filtering solutions presented throughout this work. These were carried out at the Institute for Systems and Robotics' installations, and consisted of a drone autonomously describing a trajectory inside the flight arena depicted, with the chosen inertial frame of reference, in Fig. 9.1. Throughout the length of the experiment, flight measurements were acquired for *a posteriori* analysis. The configuration of the experiment, measurement acquisition, and results are presented in the following sections.

Figure 9.1: Test arena

9.1 Configuration

The test arena is equipped with a motion capture system (MOCAP) and off-board computers capable of performing drone position control [33]. The specifics of the aforementioned system are out of the scope of this work, although it is known that it can establish a link with the drone in order to control its position to an inputted continuous-time trajectory. As the trajectory presented in Chapter 8 exceeded arena bounds, it was modified to

$$\mathbf{p}'(t) = \begin{bmatrix} 1.6cos\left(\frac{2\pi}{30}t\right) \\ 2.6cos\left(\frac{\pi}{10}t + \frac{\pi}{6}\right) \\ 1.2 + 0.6cos\left(\frac{2\pi}{45}t + \frac{\pi}{9}\right), \end{bmatrix}, \tag{9.1}$$

where $t > 0 \in \mathbb{R}$ is the continuous variable representative of the time elapsed since the start of the trajectory. The full experiment lasted for approximately five minutes. This included the movement between the initial position, set by hand close to the center of the reference frame, and $\mathbf{p}'(0)$, and that between the final point of $\mathbf{p}'(t)$ and the stopping point on the ground. A plot of the trajectory is shown in Fig. 9.2.

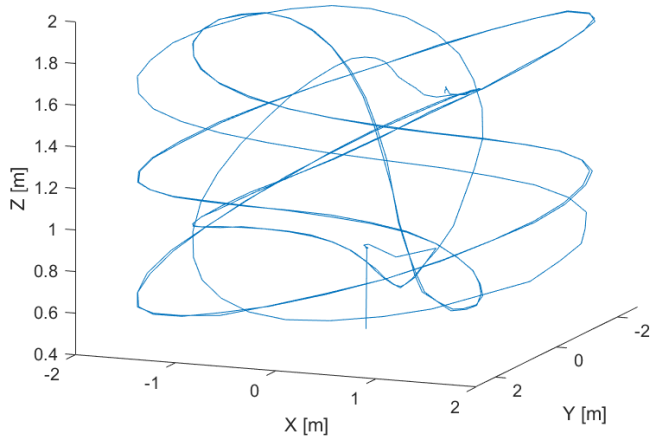

Figure 9.2: Experimental trajectory

Two completely independent systems were used to acquire both measurements needed to implement either of the three filters discussed in the previous chapters. The pseudo-ranges, $r(k)$, were acquired by an off-the-shelf acoustic "Indoor Positioning and Navigation System" offered by Marvelmind Robotics™ [34]. The basic version of this system consists of four stationary beacons, a moving transponder (physically equal to the stationary beacon), and a modem to connect all the modules of the system to a computer. All of these components are shown in Fig. 9.3. Although the aforementioned system was not originally designed to provide the distance between one stationary beacon and a moving transponder, it can be programmed to that end. By making use of the C programming language code distributed by Marvelmind Robotics™ [35], one can set the moving transponder to output the distance between each of the stationary beacons and itself. Using only one stationary beacon, the moving

Figure 9.3: Marvelmind™ ”Indoor Positioning and Navigation System” set

transponder outputs the essential range measurements needed to implement the filters. Notice that, as these measurements can only be obtained via the moving transponder, a connection between it and a data processor is needed. In order to reduce the complexity of the implemented solution and not have the transponder connected to the drone, the stationary beacon was embarked on the drone and the moving transponder was fixed to the arena floor and connected to a computer. Although counter-intuitive, this solution does not affect the obtained results as only range measurements are being collected from this system.

Regarding the position variations, $\mathbf{u}(k)$, these were obtained through the MOCAP system present in the arena. Although discrete-time inertial position variations could have been obtained directly from the MOCAP system, a more realistic approach was implemented. This consisted on both Euler angles and vehicle inertial velocity, expressed in body coordinates, being acquired in order to compute said position variations. Notice that, in realistic cases, such as underwater navigation, these quantities are generally obtained by systems like DVLs and AHRSs. Also, for airborne vehicles, orientation and velocity can be determined by inertial measurement units and pitot tubes, respectively. The MOCAP system is therefore mimicking the existence of similar devices embarked on the drone.

A more in-depth look into measurement acquisition and computation is provided in the following section.

9.2 Measurements

As said before, pseudo-range measurements were acquired by implementing a simple solution based on an off-the-shelf product. The non embarked beacon was placed on the floor of the arena in the inertial coordinates $\mathbf{s} \approx \begin{bmatrix} -2.31 & -0.10 & 0.01 \end{bmatrix}^T [m]$. Regarding the update rate of these measurements, this was set to 1 Hz. Finally, since all data analysis was done after the physical experiments took place, the system output comprised a .csv file where each row contained the pseudo-range, in meters, followed

by a timestamp, in milliseconds. An example of the .csv file is shown in Table 9.1. Notice that the

Range [m]	Timestamp [ms]
4.847	1892191
4.722	1893189
4.134	1894188
3.245	1896187
1.497	1897185
2.587	1898184
4.349	1899183
3.847	1900181
...	...

Table 9.1: Pseudo-range measurements acquisition system output example.

variation between timestamps is never exactly 1000 milliseconds as it should be. This corresponds to an unavoidable system limitation, which must be taken into account when analyzing the results. Also, some range measurements are missing from the output file, as can be seen between the third and fourth measurements. To tackle this randomly occurring problem, one can take into consideration that Kalman filters are based on a prediction-correction algorithm. This means that, if there are no range measurements available in one or more time intervals, then state estimates corresponding to these time intervals can be computed with the predictive step only, which, in turn, can be done using the inertial position variation measurements only.

To determine the position variations, $\mathbf{u}(k)$, the MOCAP system was used. Having finalized the autonomous trajectory, this system outputs another .csv file, which contains various flight parameters sampled at a frequency of 30 Hz. An example of the output .csv file is presented in Appendix A. Although more trajectory parameters are measured, only the following were used:

- Velocity (x, y, z), $[ms^{-1}]$
- Euler angles (ψ, ϕ, θ), $[rad]$
- Position (x, y, z), $[m]$.

The set of measurements comprises vehicle inertial velocity, expressed in body-coordinates, and orientation, given in terms of roll (ϕ), pitch (ψ), and yaw (θ), more commonly known as Euler angles. Furthermore, the system also outputs the actual position of the drone along the trajectory, which can be considered as the ground truth. This latter information is of great importance, since it allows to analyze the filter performance in what concerns the determination of position estimation errors. The position variations, sampled at a frequency of 1 Hz (to match the pseudo-range measurements frequency), were computed as given by (2.5). The rotation matrix $\mathbf{R}(t_k)$ is a function of the Euler angles (ψ,ϕ,θ), which can be computed using

$$\mathbf{R}(t_k) = \mathbf{R}(\psi(t_k))\mathbf{R}(\phi(t_k))\mathbf{R}(\theta(t_k)) =$$

$$\mathbf{R}(t_k) = \begin{bmatrix} cos(\psi(t_k)) & -sin(\psi(t_k)) & 0 \\ sin(\psi(t_k)) & cos(\psi(t_k)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos(\phi(t_k)) & 0 & sin(\phi(t_k)) \\ 0 & 1 & 0 \\ -sin(\phi(t_k)) & 0 & cos(\phi(t_k)) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\theta(t_k)) & -sin(\theta(t_k)) \\ 0 & sin(\theta(t_k)) & cos(\theta(t_k)) \end{bmatrix}. \tag{9.2}$$

As both rotation matrices and velocities needed to determine the integral are available in discrete-time instants, the latter can only be approximated through a sum. To that end, the trapezoidal rule was employed as described by

$$u(k) \approx T^+ \sum_{n=0}^{\frac{T^-}{T^+}-1} \frac{\mathbf{R}(t_k + nT^+)\mathbf{v}(t_k + nT^+) + \mathbf{R}(t_k + (n+1)T^+)\mathbf{v}(t_k + (n+1)T^+)}{2}, \tag{9.3}$$

where the period T^+ corresponds to the frequency at which rotation matrices and vehicle velocities were obtained, and T^- to the pseudo-range measurements update rate. In this case, we have $T^+ = \frac{1}{30Hz}$ and $T^- = \frac{1}{1Hz}$.

A very important aspect regarding these experiments has to do with the fact that the MOCAP system tends to provide flight information very precisely. Accounting for this, and to achieve more realistic results, zero mean white Gaussian noise with the standard deviations present in Table 9.2 was added to the Euler angles and velocities acquired by the MOCAP system.

Measure	Standard deviation
Roll (ϕ)	$0.03°$
Pitch (ψ)	$0.03°$
Yaw (θ)	$0.3°$
Velocity	0.01m/s

Table 9.2: Standard deviation of the noise

Another point worth mentioning is the synchronization between the two measurement acquisition systems. As there was no relation between system time-stamps, synchronization was done manually with the help of the ground truth values provided by the MOCAP system. Having the true positions of the drone, the expected distance between it and the stationary beacon fixed in the ground can be determined. Then, by simultaneously plotting both experimental and expected values, a synchronization process consisting in the attempt of overlapping both lines, as shown in Fig.9.4, provides system synchronization. Although this process might introduce some errors in the final results, it was selected as it is a simple and expedite way of synchronizing the two independent systems.

9.3 Results

Having acquired the measurements described in the previous section, filtering solutions can be employed to perform state estimation.

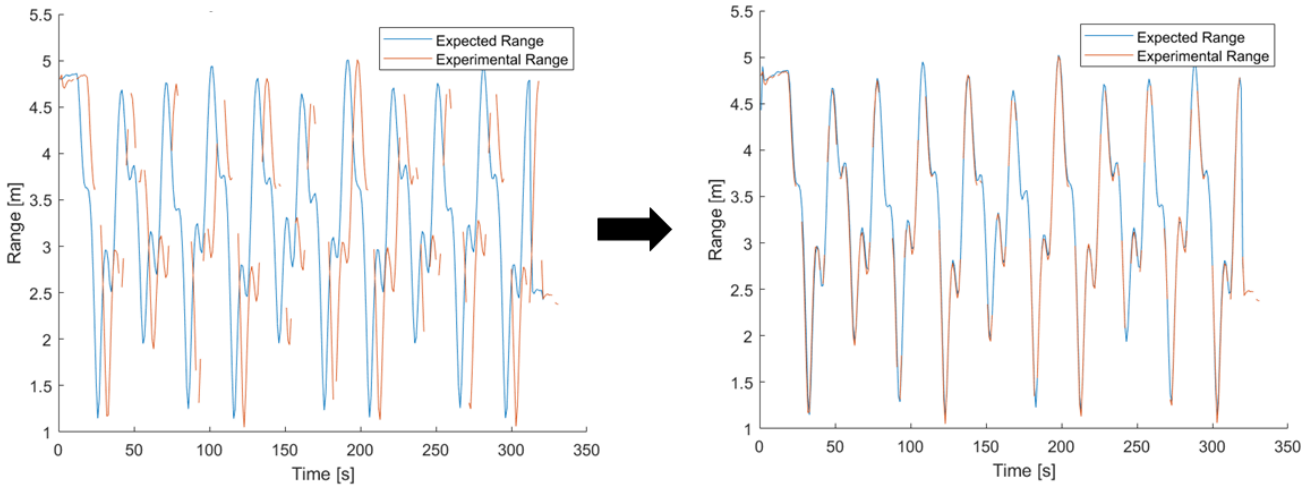

Figure 9.4: Synchronization process visualization

All three filters were tested using different settings. Each filter was initialized with a position error on coordinate x. Input parameters of each filter were subject to the optimization process described in Chapter 6. Table 9.3 summarizes all input parameters. Notice that, besides covariance matrix inputs

	LKF	EKF	UKF
Q_k	$diag(0.2405\mathbf{I}_3, 0.0021, 0.0125)$	$diag(0.7264\mathbf{I}_3, 00120)$	$diag(0.8172\mathbf{I}_3, 0.0020)$
R_k	0.6922	0.9982	0.2535
$\mathbf{P}_0$	$\mathbf{I}_5$	$\mathbf{I}_4$	$\mathbf{I}_4$
$\hat{\mathbf{p}}_0 - \mathbf{p}_0$	$\begin{bmatrix}100 & 0 & 0\end{bmatrix}^T [m]$	$\begin{bmatrix}10 & 0 & 0\end{bmatrix}^T [m]$	$\begin{bmatrix}10 & 0 & 0\end{bmatrix}^T [m]$

Table 9.3: Experimental filter input parameters

being different for each filter, also the initial state estimate differs from the LKF to both the EKF and UKF, since too large initial state errors lead to non-converging results in the latter two, as seen in Sections 8.3 and 8.4.

The results of position estimation error are shown in Figures 9.5, 9.6 and 9.7. The better performing solution is clearly the LKF since, even having started with an initial error an order of magnitude above that of the competing solutions, it leads to fast estimation convergence while attaining remarkable steady-state performance, with errors contained within a margin of 50 centimeters. It can also be seen that the EKF performs worse than the LKF. However, it could be used as a viable way to perform estimation in situations where there is a good knowledge of the true initial position. Finally, the UKF represents the worst performing solution since, within the time span of the experience (approx. 5 min), it could not manage to lead state estimates to their true values. Notice, however, this result is in accordance with what was concluded on Section 8.4 about the convergence rates of the UKF.

Aiming at the completeness of this experiment, estimates of the speed of sound unknown multiplicative coefficient using the LKF are shown in Fig. 9.8. No error can be computed to evaluate this particular

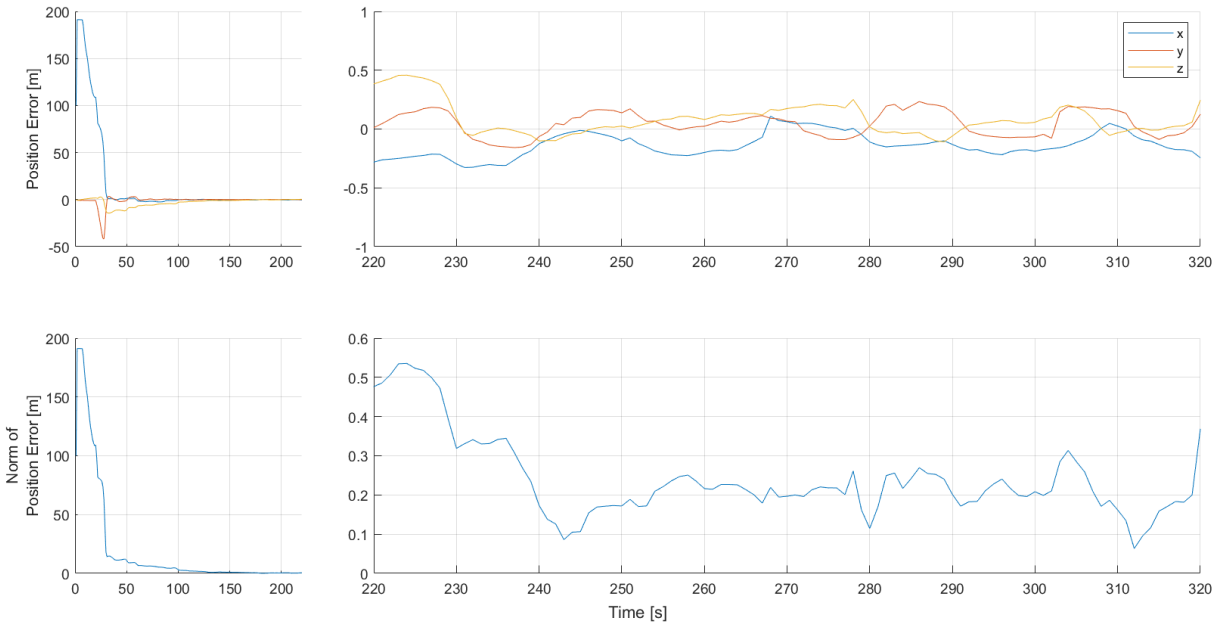

Figure 9.5: LKF position estimation error

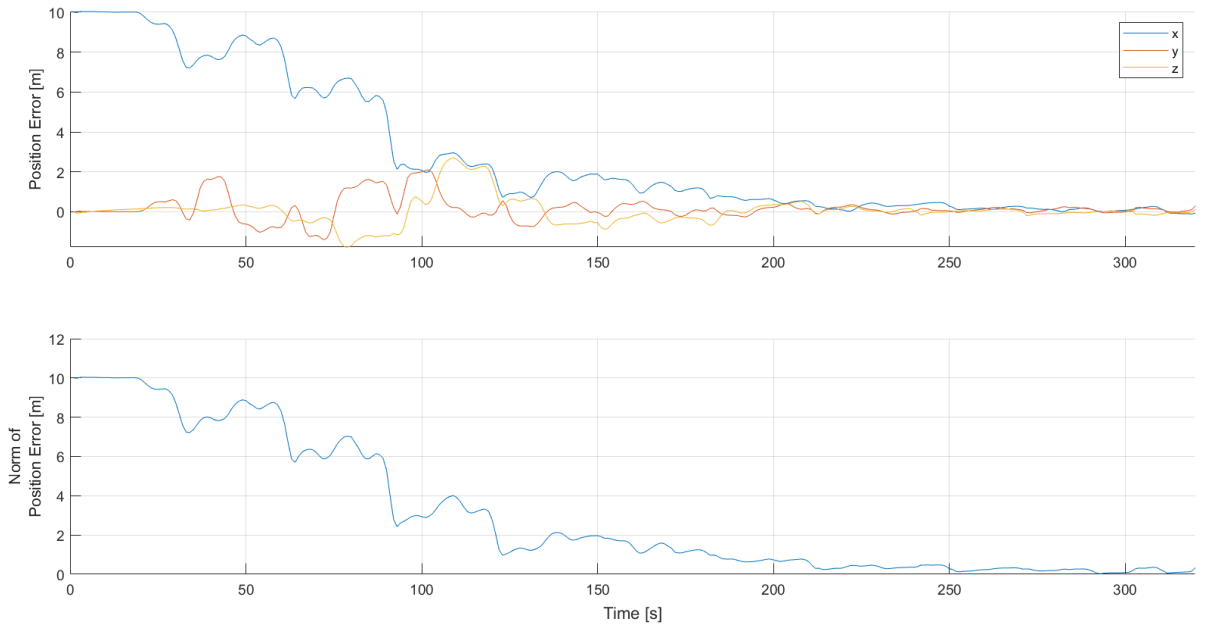

Figure 9.6: EKF position estimation error

estimate as true speed of sound in the arena is not available; only rough estimates based on temperature could have been performed. Nevertheless, it is clear that the coefficient estimates tend to a value of approximately 1.05, thus confirming the expected operation of the filter. To further confirm the proper functioning of the speed of sound multiplicative coefficient estimation, an artificial multiplicative error of 1.1 was applied to the pseudo-range measurements acquired by the Marvelmind™system. The results presented in Fig. 9.9 show an approximate 10% increase in the steady-state speed of sound multiplica-

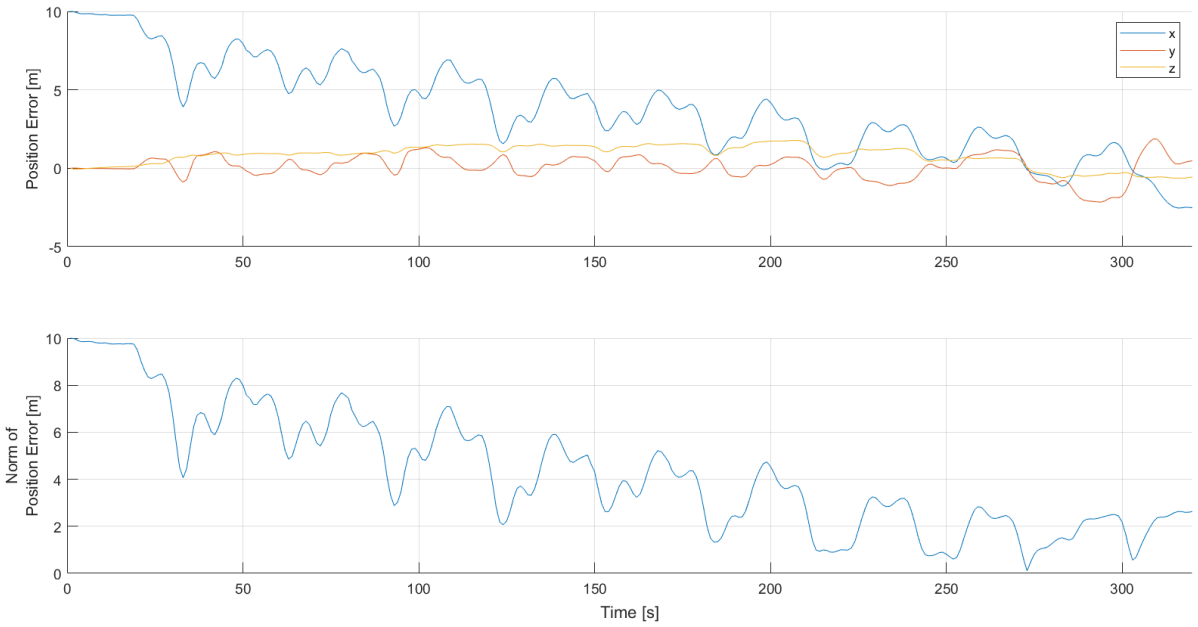

Figure 9.7: UKF position estimation error

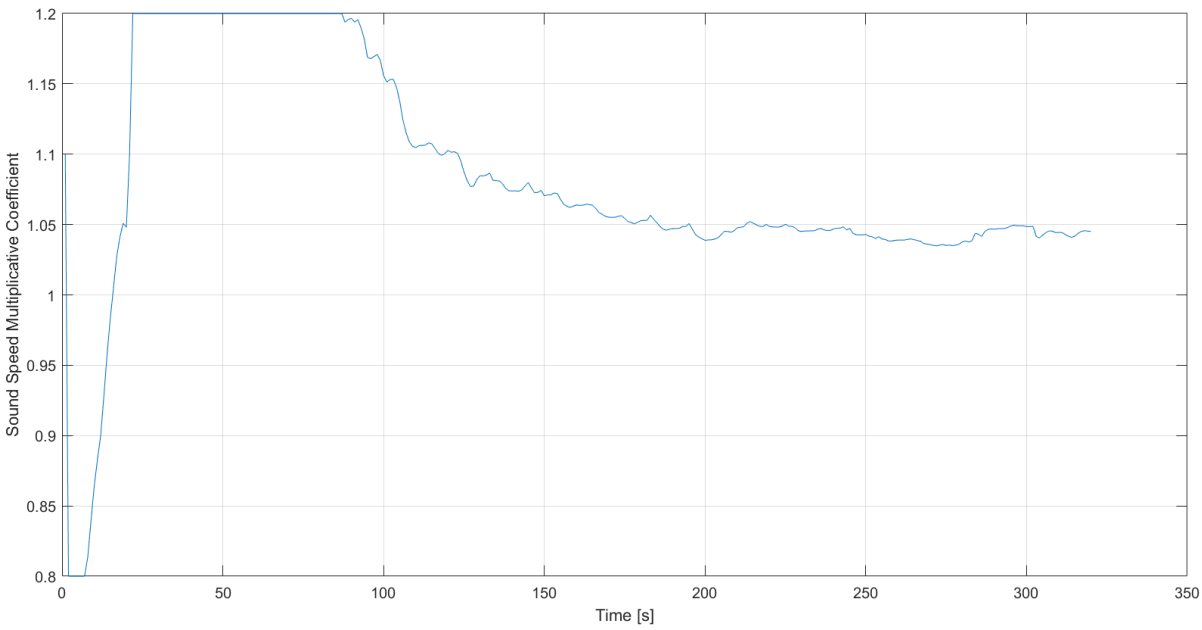

Figure 9.8: LKF sound speed multiplicative coefficient estimation results

tive coefficient estimates in comparison to those obtained without artificial pseudo-range inputs. Hence, the proper functioning of the LKF is confirmed.

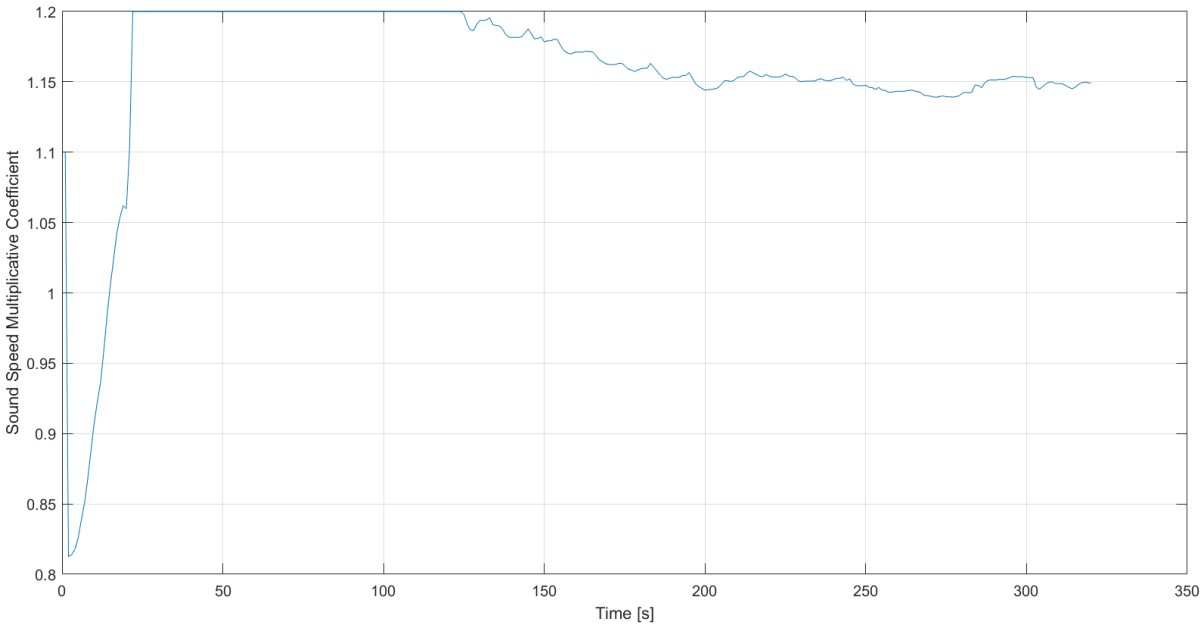

Figure 9.9: LKF sound speed multiplicative coefficient estimation results with artificial pseudo-range input

Chapter 10

Conclusions

In this thesis a new solution is employed to tackle the nonlinear estimation problem of underwater acoustic navigation with an unknown speed of sound in the medium. This novel solution provides position and unknown speed of sound multiplicative coefficient estimates with errors that tend rapidly to zero, while achieving respectable steady-state performance. Its major advantage is the fact that it has mathematically proven globally exponentially stable error dynamics. This is something that neither one of the common nonlinear estimation techniques used to compare this approach to, namely the EKF and the UKF, is able to provide.

A very interesting result is that, even though the system derived to model the navigation problem did not assume a variable unknown speed of sound multiplicative coefficient, all the filters were able to accurately estimate it when a slow variation was introduced.

It was also seen, through Monte Carlo simulations, that the average steady-state position and speed of sound coefficient estimation errors are well condensed within a thin margin. Nevertheless, a small bias in all three filters was found. Regarding optimal performance, all the solutions showed error standard deviations close to the lower bound given by the Baeysian Cramér-Rao bound. In fact, due to the biased nature of the filters, error standard deviations were sometimes bellow the optimal bound provided by the BCRB.

The experiments carried out in the test arena showed that the new solution can be seen as a viable method to perform vehicle navigation, even when the initial position of the vehicle is poorly determined. Also, the experimental results proved that an unknown speed of sound in the propagation medium does not pose an impediment to the correct work of the proposed solution.

Due to the installation and equipment limitations found when preparing the experimental procedure of this thesis, the inertial measurements needed to implement the estimation processes could not be obtained from sensors specifically designed for that purpose, instead, a motion capture system was used to mimic the existence of these sensors. Also an underwater vehicle had to be substituted by a drone flying in a test arena, which showed the broad applicability of the solution. With this in mind, further work on this subject could be done by projecting and executing a new experiment with real underwater vehicles equipped with inertial measurement units. This configuration would unequivocally demonstrate

the true potential of the new solution when applied to real-life situations. Although it might be difficult to achieve, an experimental procedure could also be devised in order to accurately determine the errors associated with the estimation of the speed of sound multiplicative coefficient.

Bibliography

[1] J. H. Been, R. de Koning, W. L. de Vlieger. On improving target motion analysis trough own ship maneuvering. *Digital Signal Processing*, 1991.

[2] S. D. McPhail and M. Pebody. Range-Only Positioning of a Deep-Diving Autonomous Underwater Vehicle From a Surface Ship. *IEEE Journal of Oceanic Engineering*, 34(4):669–677, oct 2009. ISSN 1558-1691. doi: 10.1109/JOE.2009.2030223.

[3] S. Webster, R. Eustice, H. Singh, and L. Whitcomb. Advances in single-beacon one-way-travel-time acoustic navigation for underwater vehicles. *International Journal of Robotic Research - IJRR*, 31: 935–950, 2012. doi: 10.1177/0278364912446166.

[4] D. Moreno-Salinas, A. M. Pascoal, and J. Aranda. Underwater Target Positioning with a Single Acoustic Sensor. *IFAC Proceedings Volumes*, 46(33):233–238, 2013. ISSN 1474-6670. doi: https://doi.org/10.3182/20130918-4-JP-3022.00067. URL `http://www.sciencedirect.com/science/article/pii/S147466701646163X`.

[5] D. Moreno-Salinas, A. M. Pascoal, and J. Aranda. *Optimal sensor trajectories for mobile underwater target positioning with noisy range measurements*, volume 19. IFAC, 2014. ISBN 9783902823625. doi: 10.3182/20140824-6-za-1003.02407. URL `http://dx.doi.org/10.3182/20140824-6-ZA-1003.02407`.

[6] M. B. Larsen. Synthetic long baseline navigation of underwater vehicles. *Oceans Conference Record (IEEE)*, 3:2043–2050, 2000. ISSN 01977385. doi: 10.1109/oceans.2000.882240.

[7] A. S. Gadre and D. Stilwell. *Toward underwater navigation based on range measurements from a single location*, volume 5. jan 2004. ISBN 0-7803-8232-3. doi: 10.1109/ROBOT.2004.1302422.

[8] A. S. Gadre and D. J. Stilwell. Underwater Navigation In the Presence of Unknown Currents Based On Range Measurements From a Single Location. In *American Control Conference*, volume 2, pages L.02–L02, 2005. ISBN 0-7803-9098-9.

[9] A. S. Gadre and D. Stilwell. *A complete solution to underwater navigation in the presence of unknown currents based on range measurements from a single location*. sep 2005. doi: 10.1109/IROS.2005.1545230.

[10] J. Jouffroy and J. Reger. An algebraic perspective to single-transponder underwater navigation. *Proceedings of the IEEE International Conference on Control Applications*, pages 1789–1794, 2006. doi: 10.1109/CACSD-CCA-ISIC.2006.4776912.

[11] P. Batista, C. Silvestre, and P. Oliveira. Single range aided navigation and source localization: Observability and filter design. *Systems and Control Letters*, 60(8):665–673, 2011. ISSN 01676911. doi: 10.1016/j.sysconle.2011.05.004. URL http://dx.doi.org/10.1016/j.sysconle.2011.05.004.

[12] G. Indiveri, D. De Palma, and G. Parlangeli. Single Range Localization in 3-D: Observability and Robustness Issues. *IEEE Transactions on Control Systems Technology*, 24(5):1853–1860, 2016. ISSN 10636536. doi: 10.1109/TCST.2015.2512879.

[13] G. Indiveri. An Entropy-Like Estimator for Robust Parameter Identification. *Entropy*, 11, 2009. doi: 10.3390/e11040560.

[14] W. D. Wilson. Speed of Sound in Sea Water as a Function of Temperature, Pressure, and Salinity. *The journal of the acoustical society of america*, 1960.

[15] C.-T. A. Chen and F. Millero. Speed of sound in seawater at high pressures. *Journal of The Acoustical Society of America - J ACOUST SOC AMER*, 62:1129–1135, 1977. doi: 10.1121/1.381646.

[16] W. Yan, W. Chen, and R. Cui. Moving long baseline positioning algorithm with uncertain sound speed. *Journal of Mechanical Science and Technology*, 29:3995–4002, 2015. doi: 10.1007/s12206-015-0845-z.

[17] P. Batista. GES long baseline navigation with unknown sound velocity and discrete-time range measurements. *IEEE Transactions on Control Systems Technology*, 23(1):219–230, 2015. ISSN 10636536. doi: 10.1109/TCST.2014.2321973.

[18] P. Batista. Long baseline navigation with explicit pseudo-range clock offset and propagation speed estimation. *European Journal of Control*, 49:116–130, 2019. ISSN 0947-3580. doi: https://doi.org/10.1016/j.ejcon.2018.12.010. URL http://www.sciencedirect.com/science/article/pii/S0947358017302790.

[19] P. Batista. Navigation and source localization based on single pseudo-ranges. *Proceedings of the American Control Conference*, 2020-July:5237–5242, 2020. ISSN 07431619. doi: 10.23919/ACC45564.2020.9148010.

[20] H.-D. Qin, X. Yu, Z.-B. Zhu, and Z.-C. Deng. An expectation-maximization based single-beacon underwater navigation method with unknown ESV. *Neurocomputing*, 378:295–303, 2020. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2019.10.066. URL http://www.sciencedirect.com/science/article/pii/S0925231219315048.

[21] H.-D. Qin, X. Yu, Z.-B. Zhu, and Z.-C. Deng. A variational Bayesian approximation based adaptive single beacon navigation method with unknown ESV. *Ocean Engineering*, 209:107484, 2020. ISSN 0029-8018. doi: https://doi.org/10.1016/j.oceaneng.2020.107484. URL http://www.sciencedirect.com/science/article/pii/S0029801820304996.

[22] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*. 1970.

[23] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(1):35–45, 1960. URL http://dx.doi.org/10.1115/1.3662552.

[24] A. Gelb. *Optimal linear filtering. In Applied Optimal Estimation*, volume 8. 1974. ISBN 0818683864.

[25] E. A. Wan and R. Van Der Merew. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pages 153–158, Lake Louise, Alberta, Canada, 2000. doi: 10.1109/ASSPCC.2000.882463.

[26] S. J. Julier and J. K. Uhlmann. New extension of the Kalman filter to nonlinear systems. *Signal Processing, Sensor Fusion, and Target Recognition VI*, 3068:182, 1997. ISSN 0277786X. doi: 10.1117/12.280797.

[27] Y. Wu, D. Hu, M. Wu, and X. Hu. Unscented Kalman filtering for additive noise case: Augmented vs. non-augmented. *Proceedings of the American Control Conference*, 6:4051–4055, 2005. ISSN 07431619. doi: 10.1109/acc.2005.1470611.

[28] I. Roman, J. Ceberio, A. Mendiburu, and J. A. Lozano. Bayesian optimization for parameter tuning in evolutionary algorithms. *2016 IEEE Congress on Evolutionary Computation, CEC 2016*, pages 4839–4845, 2016. doi: 10.1109/CEC.2016.7744410.

[29] J. Snoek, H. Larochelle, and R. P. Adams. Practical Bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 4:2951–2959, 2012. ISSN 10495258.

[30] A. Zilinkas, V. Tiesis, and J. Mockus. Toward Global Optimization. pages 117–128. 1978.

[31] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger. Information-theoretic regret bounds for Gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58 (5):3250–3265, 2012. ISSN 00189448. doi: 10.1109/TIT.2011.2182033.

[32] P. Batista, C. Silvestre, and P. Oliveira. Navigation systems based on multiple bearing measurements. *IEEE Transactions on Aerospace and Electronic Systems*, 51(4):2887–2899, 2015. ISSN 00189251. doi: 10.1109/TAES.2015.140515.

[33] T. Oliveira. *Rapid Development and Prototyping Environment for Testing of Unmanned Aerial Vehicles*. Msc thesis, Instituto Superior Técnico, 2020.

[34] M. Robotics. Precise (±2cm) Indoor Positioning and Navigation for autonomous robots, drones, vehicles and humans, 2020/07, . URL https://marvelmind.com/.

[35] M. . Robotics. Downloads, 2020/07, . URL https://marvelminds.com/download/.

Appendix A

MOCAP .csv file example

Timestamp[ms]	$\mathbf{v}^x[ms^{-1}]$	$\mathbf{v}^y[ms^{-1}]$	$\mathbf{v}^z[ms^{-1}]$	$\hat{\mathbf{p}}^x[m]$	$\hat{\mathbf{p}}^y[m]$	$\hat{\mathbf{p}}^z[m]$	$\phi[rad]$	$\psi[rad]$	$\theta[rad]$
1603131593	-0.6111	-0.2553	-0.2863	2.2503	1.6566	1.4416	0.0984	-0.1644	-3.1240
1603131593	-0.5418	-0.2126	-0.2808	2.2706	1.6635	1.4558	0.0972	-0.1649	-3.1252
1603131593	-0.4906	-0.1751	-0.2747	2.2829	1.6677	1.4653	0.0963	-0.1644	-3.1260
1603131593	-0.4347	-0.1452	-0.2690	2.2966	1.6713	1.4768	0.0953	-0.1653	-3.1271
1603131593	-0.3764	-0.1147	-0.2728	2.3105	1.6747	1.4897	0.0938	-0.1643	-3.1278
...	...	...	...	...	...	...	...	...	..
1603131594	0.2216	0.1404	-0.1095	2.2356	1.5766	1.6386	-0.0106	0.0059	3.1203
1603131594	0.2208	0.1348	-0.1147	2.2278	1.5717	1.6429	-0.0119	0.0072	3.1187
1603131594	0.2114	0.1335	-0.1062	2.2219	1.5682	1.6455	-0.0125	0.0086	3.1171
1603131594	0.2027	0.1220	-0.1052	2.2147	1.5645	1.6476	-0.0126	0.0102	3.1154
1603131594	0.1987	0.1176	-0.0993	2.2074	1.5608	1.6515	-0.0127	0.0114	3.1147
...	...	...	...	...	...	...	...	...	..
1603131595	0.0397	0.0047	-0.0536	2.1098	1.5337	1.6938	0.0038	0.0139	3.1314
1603131595	0.0329	0.0038	-0.0545	2.1088	1.5340	1.6948	0.0040	0.0135	3.1320
1603131595	0.0260	0.0071	-0.0470	2.1077	1.5336	1.6954	0.0049	0.0136	3.1327
1603131595	0.0225	0.0037	-0.0505	2.1072	1.5341	1.6960	0.0050	0.0135	3.1333
1603131595	0.0134	0.0066	-0.0414	2.1064	1.5340	1.6970	0.0052	0.0129	3.1335

Table A.1: MOCAP system output example