Formation Control of Quadrotor Vehicles

João Pedro Lopes Rocha

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Supervisors: Dr. Rita Maria Mendes de Almeida Correia da Cunha
Dr. Carlos Jorge Ferreira Silvestre

Examination Committee

Chairperson:	Dr. João Manuel Lage de Miranda Lemos
Supervisor:	Dr. Rita Maria Mendes de Almeida Correia da Cunha
Member of the Committee:	Dr. João Pedro Castilho Pereira Santos Gomes

December 2015

À minha avó, Maria

Acknowledgments

First of all, I would like to start by thanking my advisors Prof. Rita Cunha and Prof. Carlos Silvestre for the guidance and support throughout this thesis and for being always available to help me and give me their precious feedback about this work. It was a pleasant experience working with them.

To all the members of the DSOR Lab, I would like to express my gratitude for contributing to a good work environment during these last months.

To all the friends I made during the last five years in Lisbon and in Delft, I would also like to thank for making university such a fantastic experience that I will always remember.

To all my childhood friends from Leiria, I am very grateful that we kept in touch and proved that distance does not ruin relationships.

To my girlfriend, who was always there for me, I am highly indebted for all the encouragement and endless support during these years.

Last but not least, my final words of gratitude are addressed to my parents and my family, who always supported me and are my source of inspiration, giving me everything I needed throughout my whole life.

Resumo

O uso de múltiplos quadcópteros para executar tarefas cooperativas é um campo de investigação ativo e de extrema importância. Uma missão específica de interesse envolve uma formação de quadcópteros que realiza inspeções a infraestruturas de forma autónoma. Esta tese descreve uma situação onde um quadcóptero-alvo está a executar uma tarefa num ambiente em que não tem sinal GPS e dois quadcópteros-sensores com sinal de GPS estimam a posição do alvo, combinando as suas posições com medições de direção. A escolha das posições dos sensores pode afetar significativamente o desempenho do processo de estimação, motivando a otimização das várias funções-objetivo com base na Matriz de Informação de Fisher. Dois novos algoritmos de estimação são criados e testados e os resultados sugerem configurações geométricas que aumentam a precisão da estimativa. Assumindo um modelo de duplo-integrador para o quadcóptero-alvo, um controlador é associado ao estimador não-linear para controlar a posição do alvo. Os resultados mostram a influência do posicionamento dos sensores não apenas no erro de estimação, mas também na diferença entre a posição real e a pretendida do alvo. Por último, é considerado um problema de localização numa formação constituída por múltiplos quadcópteros-alvos e dois sensores com pleno conhecimento da sua posição. Os quadcópteros-alvos estimam a sua posição usando não só medições de direção provenientes de um dos sensores, mas também provenientes de um outro alvo. É descrito e demonstrado um método de construção de formações que garante sempre a convergência das estimativas dos alvos para as suas posições reais.

Palavras-chave: Sistemas de múltiplos quadcópteros, Geometrias de localização ótimas, Medições de direção, Localização de formações.

Abstract

The use of multiple quadrotors to perform cooperative tasks is an active research field of crucial importance. One particular mission of interest involves a formation of quadrotors performing infrastructure inspection autonomously. This thesis describes a scenario where a Target quadrotor is performing a task in a GPS-denied environment, and two Sensor quadrotors with GPS capabilities estimate the Target's position combining their positions with bearings-only measurements. The selection of the Sensors' positions can greatly affect the estimation performance, motivating the optimization of several objective functions based on the Fisher Information Matrix (FIM). Two novel estimation algorithms are created and tested for both cases where the Target is stationary and moving. The results show geometric configurations that increase the estimation accuracy, illustrating the importance of maximizing the FIM for Target localization using quadrotors. Assuming a double integrator model for the target vehicle, an LQR controller is combined with the nonlinear estimator to achieve tracking of a desired trajectory. The results show the influence of the Sensors' positioning not only on the estimation error, but also on the difference between the true and desired position of the Target. Finally, a network localization problem with multiple Targets and two Sensors with full knowledge of their position is considered. The Targets estimate their position using not only bearings measurements coming from one of the Sensors, but also coming from another Target. A construction method of network formations that guarantees the convergence of the Targets' estimates to their actual positions is described and demonstrated.

Keywords: Multi-quadrotor systems, Optimal localization geometries, Bearings-only measurements, Network localization.

Contents

List of Tables

List of Figures

List of Symbols

$\cdot \times \cdot$	Cross product operator
$\lambda_{max}(\cdot)$	Maximum eigenvalue of the argument matrix
$\lambda_{min}(\cdot)$	Minimum eigenvalue of the argument matrix
$\|\|.\|\|$	Absolute value operator
$\|\mathbf{A}\|$	Matrix determinant operator
$\mathbb{E}\{\cdot\}$	Expected value operator
$\mathbb{R}^n$	Set of n-dimensional vectors with real numbers
$\mathbf{0}$	Matrix of zeros, with dimensions obvious from context
$\mathbf{\Sigma}$	Noise covariance matrix
$\mathbf{A}$	Matrix variable
$\mathbf{a}$	Vector variable
$\mathbf{A} \prec \mathbf{0}$	Negative definite matrix, in the sense that $\mathbf{x}^T\mathbf{A}\mathbf{x} < 0$ for all $\mathbf{x} \neq 0$; a negative semidefinite matrix is denoted as $\mathbf{A} \preceq 0$
$\mathbf{A} \succ \mathbf{0}$	Positive definite matrix, in the sense that $\mathbf{x}^T\mathbf{A}\mathbf{x} > 0$ for all $\mathbf{x} \neq 0$; a positive semidefinite matrix is denoted as $\mathbf{A} \succeq 0$
$\mathbf{I}$	Identity matrix, with dimensions obvious from context
$\mathbf{I}_n$	Identity matrix of dimensions $n \times n$
$\mathbf{J}(\mathbf{x})$	Fisher Information Matrix
$\mathbf{P}$	Error covariance matrix
$\mathbf{r}$	Relative vector between target and sensor positions
$\mathbf{s}$	Sensor position vector
$\mathbf{t}$	Target position vector

$\mathbf{z}$ Measurement vector

$\text{cov}(\cdot)$ Covariance operator

$\ln(\cdot)$ Natural logarithm operator

$\text{tr}(\cdot)$ Trace of a matrix operator

$\nabla_{\mathbf{x}}(\cdot)$ Gradient operator

ϕ Azimuth angle

ψ Elevation angle

$\rho(\cdot)$ Spectral radius operator

φ LQR cost functional

a, α Scalar variable or constant

d Bridge's width

$S(\cdot)$ Skew-symmetric matrix operator, such that $S(\mathbf{a})\mathbf{b} = \mathbf{a} \times \mathbf{b}$

t Time variable

V Lyapunov function

Subscripts

0 Initial condition

i, j, k Computational indexes

LQR Linear Quadratic Regulator

x, y, z Cartesian components.

Superscripts

* Desired

$^+$ In the next instant of time

$\dot{}$ First derivative

^{T} Transpose

$^{-1}$ Inverse

$\ddot{}$ Second derivative

$\hat{}$ Estimated

List of Acronyms

2D	Two-Dimensional Space
3D	Three-Dimensional Space
CRLB	Cramer Rao Lower Bound
DES	Directed Edge Splitting
DILOC	Distributed Iterative Localization Algorithm
DSOR	Dynamical Systems and Ocean Robotics
DVA	Directed Vertex Addition
EKF	Extended Kalman Filter
FIM	Fisher Information Matrix
GDOP	Geometric Dilution Of Precision
GPS	Global Positioning System
IMM	Interactive Multiple Models
IMU	Inertial Measurement Unit
KF	Kalman Filter
LQR	Linear Quadratic Regulator
LTI	Linear Time Invariant
MAP-PF	Maximum A Posteriori Penalty Function
MC	Monte Carlo
MMPF	Multi-Mode Particle Filter
PCRLB	Posterior Cramer Rao Lower Bound
PDF	Probability Density Function
RMSE	Root Mean Squared Error
UAV	Unmanned Aerial Vehicles
UKF	Unscented Kalman Filter

Chapter 1

Introduction

1.1 Motivation

The problem of controlling multiple vehicles to perform cooperative tasks poses important challenges to automatic control. It has been the scope of a number of publications and experimental results are beginning to appear, attesting the recent interest in the topic. Aerial robotic vehicles can play an important role in a wide range of applications, such as critical infrastructure inspection, given their ability to reach and fly in a close proximity to points of interest, and acquire valuable high-quality data for damage diagnosis, thus providing an attractive alternative to standard methods, which involve heavy and expensive logistics and require direct human intervention. Vertical take-off and landing rotorcraft, with hover flight capabilities, form a large and important class of UAVs (Unmanned Aerial Vehicles) ideally suited for this type of missions. In particular, the quadrotor stands out as an ideal platform for the implementation of robotic systems and for the development and test of new control strategies, due to its simplicity, high maneuverability, and ability to hover.

1.2 Problem Description

This section describes a multi-vehicle mission scenario, where a leader quadrotor vehicle is performing a task in a GPS-denied environment, like, for example, structural health monitoring under a bridge deck, and two follower quadrotor vehicles are flying in formation with the leader, but far enough from the structure to obtain GPS position information and localize themselves. Local sensor data, such as bearings-only measurements, are used to complement the GPS data and estimate the position of the leader quadrotor. Throughout this work, the leader quadrotor is also called the Target and the followers are called Sensors. By combining the bearings obtained by at least two Sensors, an estimate of the Target localization can be computed through triangulation as it can be seen in Figure 1.1. Each Sensor is equipped with a camera which recognizes the Target and is able to find the direction of the line which intersects the Sensor and the Target, expressed in the camera's reference frame. Then, knowing the camera's location in the Sensor and its attitude (which can be easily obtained with an IMU installed

on-board), the direction of this line can be expressed in an inertial reference frame. For the purpose of this thesis, the measurements obtained between the Sensors and the Target are, in the 2D case, the azimuth bearings measured positive clockwise from north (y-axis), as Bishop et al. [4] use. In the 3D case, in addition to the azimuth bearing angle, an elevation bearing measurement needs to be considered. Given two Sensor measurements, the true Target position can be correctly estimated.

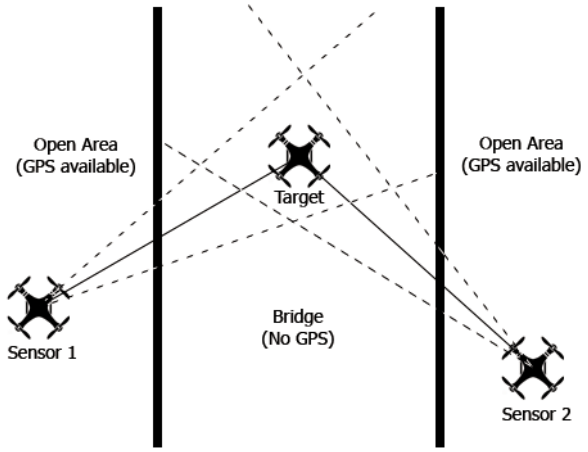
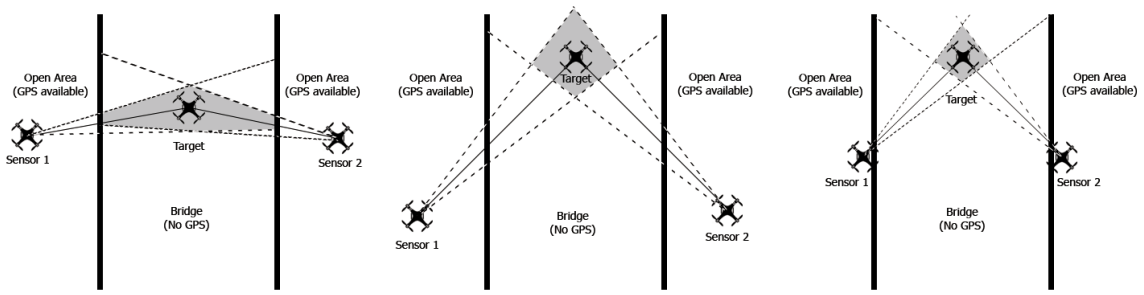


Figure 1.1: Target Localization under a Bridge using Quadrotors with Bearings-only Sensors

The task of Target estimation presents several challenges. Most modern small quadrotors are equipped with GPS capabilities but are designed to be inexpensive and therefore include low quality sensors and poor navigation filtering algorithms, leading to inaccurate vehicle state estimation. The total Target estimation error is an accumulation of several errors, namely the errors in the sensors' position, the sensors' noise and the errors associated with the Target trajectory. For this work, some assumptions have to be made in order to simplify the problem and for this reason the Sensors' localization and Target trajectory errors are considered non-existent. On the other hand, the sensors' noise is included and increases the uncertainty about the Target estimation. Typical sensor errors involve standard deviations for heading using a magnetometer or electronic compass of about 5 degrees. Moreover, the magnetometer readings can be critically affected by surrounding magnetic fields and thus the resulting heading errors are much larger in this cases. One of the objectives of this thesis is to find the Sensors-Target configuration that best mitigates this effect.

The uncertainty of an estimation is directly affected by the Sensors-Target geometric configuration as represented in Figure 1.2. The amount of information a set of measurements produces is highly dependent not only on the angle between the measurements but also on the distance between the Sensors and the Target. The grey area shown in Figure 1.2 represents the uncertainty on Target estimation for a non-orthogonal Sensors-Target geometry close to 180^{o} (Figure 1.2(a)), an orthogonal geometry (Figure 1.2(b)) and also a reduced distance configuration (Figure 1.2(c)). Orthogonal measurements produce a lower area of uncertainty than non-orthogonal measurements. However, the distance between the Sensors and the Target must also be taken into account since reducing it greatly improves the Target localization.

For the purpose of this thesis, the Target quadrotor can be stationary or move with a predefined trajectory so that it can, for example, investigate the existence of faults along the bridge. The stationary



(a) Non-orthogonal measurements (b) Orthogonal measurements (c) Reduced distance measurements

Figure 1.2: Uncertainty on Target Localization Estimation due to Sensors' Noise for different Configurations

case is the most simple one which will be used to compare the results obtained for different configurations and the performance of the estimation algorithms. As for the other cases, a sinusoidal trajectory is chosen for the Target to go through the bridge since it is simple to compute and can cover a great area of the bridge's deck.

The geometric configurations need to be quantified in order to determine the amount of information each set of measurements produces. The Fisher Information Matrix is used to provide that information.

Another problem addressed in this thesis is the network localization in a scenario where two Sensor quadrotors have available GPS data and several Target quadrotors are under the bridge with no knowledge of their exact locations. These Targets estimate their position not only with the information coming from one of the Sensor quadrotors, but also with the measurements coming from another Target, which is as well estimating its position simultaneously. The aim is to design and prove a method to construct a network formation which guarantees that the estimations of all Targets in the network converge to their actual positions.

1.3 State-of-the-art

1.3.1 Bearings-only Target Estimation and Tracking

There are several estimator techniques which can be used for the bearings-only Target estimation. Bar-Shalom et al. [2] explain how to use linear estimators such as the least squares approach for a nonlinear problem like a bearings-only Target estimation with only one Sensor. Bar-Shalom et al. [2] and Kay [10] also present and derive state estimation techniques for nonlinear dynamic systems such as the Extended Kalman Filter (EKF). Since the bearings-only estimation problem involves nonlinear measurements, an EKF needs to be used instead of the traditional Kalman Filter. Nevertheless, the EKF has some drawbacks since the approximations done due to the linearization process in the EKF algorithm can lead to the divergence of the filter and consequently to unbounded estimation errors [2]. Therefore, the EKF is sensitive to the initial conditions of the problem and also to measurement errors, which will lead to instability.

Tremois and Le Cadre [23] explore Target estimation and tracking from a multi array of Sensors. Bell [3] also uses a network of Sensor arrays to Target position estimation, introducing a maximum a posteriori penalty function (MAP-PF) approach based on the EKF. Ristic and Arulampalam [19] propose two different algorithms for Target tracking using interactive multiple models (IMM), namely the EKF and the Unscented Kalman Filter (UKF). A multi-mode particle filter (MMPF) which uses a sequential Monte Carlo estimation method is also presented and in the end the three algorithms are compared with the Cramer Rao lower bound (CRLB) in order to determine their optimality. All three filters are suboptimal but their performance shows remarkable agreement with the theoretical bound. The MMPF is the most accurate of the three, practically attaining the CRLB in the first . However, the particle filter requires more computational resources and it has the same problem as the EKF's since it can diverge, depending on the initial conditions and on the accuracy of the measurement likelihood function.

Quintero et al. [18] focus on Target tracking using dynamic programming and deduce an expression for the fused geolocation covariance matrix using two Sensors collecting independent measurements of the Target. Quintero et al. [17] use this deduction to compute the fused measurement using the best linear unbiased estimate, which combines the two measurements from the Sensors. The measurements weight is inversely proportional to its error covariance matrix, making sense that a Sensor which is located near the Target has a lower error covariance matrix and therefore has a greater influence on the Target localization estimation.

1.3.2 Target Localization Optimization

An optimization of the Target localization can be obtained by maximizing or minimizing a performance criterion or objective function that contains the geometry of the localization problem. One method widely used in literature implies using the Fisher Information Matrix (FIM). The FIM can be interpreted as a quantification of the amount of information that a certain set of measurements has about the unobservable Target states to be estimated. Its inverse, the Cramér-Rao Lower Bound (CRLB), corresponds to the lower bound on the covariance of an unbiased estimator. This property is expressed in the Crámer-Rao inequality as shown in [4] and [24]:

$$\mathrm{cov}(\hat{\mathbf{x}}) \geq \mathbf{J}(\mathbf{x})^{-1} \tag{1.1}$$

where $\hat{\mathbf{x}}$ is an unbiased estimator and $\mathbf{J}$ is the FIM. It can be very useful to compute the CRLB since it is independent of the estimator used, so it provides a benchmark to compare the performance of any unbiased estimator. Also, if an unbiased estimator attains the CRLB, Kay [10] names it an *efficient* (minimum variance) estimator, in the sense that it efficiently uses the data. Lowering the CRLB will naturally result in a lower covariance if the estimator is efficient, still it is important to bear in mind that a non-efficient estimator is not guaranteed to have a lower covariance than another estimator even if its CRLB is lower. However, in practice, lowering the CRLB results in lower covariance for most well-designed estimators, as mentioned by Ponda [16].

Since the CRLB depends on the Sensors-Target geometry, it can be used as an objective function to be minimized in order to identify the best geometric configuration which corresponds to the estimation with

the lowest possible covariance. Another possibility would be to maximize its inverse, the FIM, which is a symmetric and positive definite matrix. Nevertheless, maximizing the Fisher Information Matrix is not an easy task and for that reason a scalar performance measure needs to be derived from the FIM. Ucinski [24] presents some scalar functions such as the determinant or the trace of the FIM, and while some of these functions, like the trace, are less complex to compute, others such as the determinant provide more geometric information, so it is necessary to take this two factors into account when choosing the objective function.

Bishop et al. [4] solve the bearings-only localization 2D geometry problem with two Sensors and one Target by using the determinant of the FIM as the cost function. Maximizing the determinant of the FIM is equivalent to minimizing the area of the uncertainty ellipse around the estimate of the Target. Bishop et al. [4] deduce a corollary which states that, for an unconstrained 2D problem, the optimal Sensors-Target geometry is unique and occurs when the angle subtended at the Target by the two Sensors is 90^{o}. Hammel et al. [7] derive numerically the determinant of the FIM to generate the optimal Sensor placement with bearing measures. Martínez and Bullo [15] investigate the optimal Sensor placement in 2D and 3D using range Sensors, deduce the FIM for both cases and compute the determinant of the FIM to use it as the cost function. Hernandez [9] introduces a new performance criteria which involves calculating the largest Target location root mean squared error (RMSE) bound, i.e., the maximum root value of the CRLB diagonal terms. Hernandez [9] also uses optimization approaches such as single-step and two-step planning to determine iteratively the best manoeuvre of each Sensor. Tichavsky et al. [21] provide an elegant Riccati like recursive method to compute the FIM for the general non-linear estimation problem. Taylor [20] shows that the inverse Fisher Information matrix propagates according to the same equations as the Kalman Filter covariance matrix. Tremois and Le Cadre [23] show that the FIM is additive for multiple independent Sensors.

1.3.3 Localization in Sensor Networks

Another area of research that is presented in this thesis is the localization using Sensor networks. It is fundamental that all the Sensors' positions are accurately known to guarantee that the information collected from their measurements is consistent. In many applications, equipping each Sensor with a GPS receiver can become very expensive both in monetary and power terms. Also, since GPS requires line-of-sight between the receiver and satellites, it does not work well indoors or when it is obstructed by buildings, mountains or some other kind of obstacle that blocks the view to the GPS satellites. In military applications, as Khan [12] mentions, it can be a drawback using GPS equipment since it is not robust to jamming. The localization algorithms presented in literature can be segmented into centralized and distributed algorithms. While in a centralized algorithm the computations are done only in one Sensor, in a distributed algorithm each Sensor estimates its position. Even though a centralized algorithm is adequate in small networks, in a large Sensor network it becomes highly costly in terms of the communication resources needed and therefore, in that case, a distributed algorithm may be more appropriate. Khan [12] develops a distributed iterative localization algorithm called DILOC where the Sensors find their locations exchanging distance measurements only with a set of their neighbours.

The DILOC method exploits the structure of the system matrix which results from the topology of the communication graph of the network. Khan [12] also demonstrates the convergence of DILOC to the exact locations of the Sensors by partitioning and analysing the iteration matrix.

The uniqueness of network localization solutions is also addressed in this thesis. One approach widely explored in literature is the theory of rigid graphs. Cauchy (1813) was one of the first to develop theorems in the rigidity theory. Laman [13] , who is responsible for the Laman's Theorem which serves as a basis for many posterior works, investigates about the combinatorial properties of a rigid skeletal structure and finds that those properties can be adequately described by a class of graphs. Whiteley [25] uses the Laman's Theorem and shows how the number of the edges and vertices of a graph are related with its rigidity. A definition of global rigidity for networks and the conditions for unique network localizability with distance measurements was given by Aspnes et al. [1]. Eren et al. [6] explore strategies for maintaining formations with limited communication/sensing requirements and present some construction methods for generic rigid point formations, such as the Henneberg Construction and the Delaunay Triangulation methods. Later, Eren [5] studies the singularity of network localization solutions both with bearings and distance information by using the theory of rigid formations. Eren [5] reaches the conclusion that a Sensor network using bearings is localizable under some conditions such as its underlying graph being globally rigid and at least two of the Sensors in the network being anchors, i.e., having absolute knowledge of their position. Hendrickx et al. [8] introduce the concept of persistence which is closely related to the rigidity and derive various properties of persistent directed graphs. Zhang et al. [26] study robot network localization with directed graphs using bearings and range measurements and compare the impact of different directed graphs in the quality of the network localization using experimental data.

1.4 Thesis Objectives and Contributions

This thesis focuses on Target localization estimation under a bridge using bearings-only measurements. The main goal of this thesis is to search for the best geometric configuration that increase the information provided to the estimation, taking into account the limitations imposed by the physical barriers of the bridge, and to show that by using these optimal configurations the estimation performance can be greatly enhanced.

Another objective is to develop and prove a method that guarantees the construction of rigid network formations involving several quadrotors which use bearings-only measurements to localize themselves.

A contribution of this work involves testing several different objective functions and studying the consequent geometric configurations. Then, the estimation results obtained with these objective functions based on the Fisher Information Matrix are compared with a non-optimal criterion. In the end, the results show that optimal criteria such as the E-optimality and the A-optimality are the best suited objective functions for this 3D bearings-only Target localization problem, leading to better results than the non-optimal "Constant Offset" criterion.

A further contribution involves developing estimation algorithms for this specific 3D bearings-only Target localization problem. Since this is a nonlinear estimation problem, traditional filtering methods

such as the Kalman Filter cannot be used and for that reason a "Stationary Target" estimator is designed for a scenario where the Target is immobile and a "Moving Target" estimator is created to track moving Targets.

One more contribution involves analysing the influence of the several geometric configurations on the error between the desired and the true Target's position, when a controller is added to the system. In general, from the results obtained, it can be concluded that a criterion which leads to better estimation results is also responsible for better controlling results. Thus, the optimality-criteria are considered best suited than for example the non-optimal "Constant Offset" criterion.

Finally, another contribution of this thesis is the implementation of a construction method for directed rigid formations and the demonstration of the convergence and uniqueness of the stationary solution obtained with that method.

1.5 Thesis Layout

This section describes the layout of the thesis. Chapter 1 provides an introduction and motivates the problem of controlling a formation of multiple quadrotor vehicles in a GPS-denied environment. After describing the main challenges of Target localization with different geometric configurations in such a scenario, this chapter includes a review of the state-of-the-art in the fields relevant for this work. It ends with a description of the thesis objectives and contributions.

Chapter 2 initially explains the concepts of Cramer-Rao Lower Bound and Fisher Information Matrix and their influence on the Target localization estimation process using bearings-only measurements. It also presents the criteria used to quantify the performance of an estimator and some popular and also some new estimation algorithms. In the end, the distinct geometries obtained after optimizing the Fisher Information Matrix with different criteria are compared in terms of the uncertainty ellipse area. Then, the results obtained for the different criteria using an estimator to predict the Target location are analysed.

Chapter 3 presents the problem of controlling the trajectory of the Target in this specific scenario. It ends with the analysis of the Simulink model including the system dynamics, the estimator dynamics and the controller. The same comparisons made between the different criteria in Chapter 2 are now performed when a controller is added to command the Target's trajectory.

Chapter 4 introduces the definition of network and point formations and the concept of rigidity in directed graphs. It then describes a possible construction method for directed rigid formations and demonstrates the convergence and uniqueness of the stationary solution constructed with that method. It ends with the implementation of the theory previously developed in order to construct a solid network formation.

Chapter 5 concludes with a summary and discussion of the results and a description of future work.

Chapter 2

Target Position Estimation

In this Chapter, the concepts of Cramer-Rao Lower Bound and Fisher Information Matrix are explained and their influence on the Target localization estimation process using bearings-only measurements is studied. Then, the criteria used to quantify the performance of an estimator are introduced. The next Subsection describes in detail the already known Kalman Filter and Extended Kalman Filter and the newly designed estimators, the "Stationary Target" Estimator and the "Moving Target" Estimator. In the end, the distinct geometries obtained after optimizing the Fisher Information Matrix with different criteria are compared in terms of the uncertainty ellipse area. Then, the results obtained for the different criteria using the "Moving Target" estimator to predict the Target location are analysed.

2.1 Cramer-Rao Lower Bound and Fisher Information Matrix

The Cramer-Rao Lower Bound (CRLB) corresponds to the lower bound on the covariance of an unbiased estimator. It can be very useful to compute the CRLB since it provides a benchmark against which it is possible to compare the performance of any unbiased estimator. Also, if an unbiased estimator achieves the CRLB it can be concluded that it is a minimum-variance unbiased estimator. The CRLB inverse is also referred as the Fisher Information Matrix (FIM), which can be interpreted as a quantification of the amount of information that a certain set of measurement has about the states to be estimated.

The derivation of the Fisher Information Matrix starts with the information inequality, as described by Lehmann and Casella [14] and by Bishop et al. [4], which states that the error covariance matrix $\mathbf{P}$ of an unbiased estimator $\hat{\mathbf{x}}$ of the state vector $\mathbf{x}$ with n components is bounded from below as follows

$$\mathbf{P} = \mathbb{E}\big\{(\hat{\mathbf{x}} - \mathbf{x})(\hat{\mathbf{x}} - \mathbf{x})^T\big\} \geq \mathbf{J}(\mathbf{x})^{-1} \tag{2.1}$$

where $\mathbf{J}(\mathbf{x})$ is the Fisher Information Matrix (FIM) defined as in [22]

$$\begin{aligned} \mathbf{J}(\mathbf{x}) &= \mathbb{E}\big\{[\nabla_{\mathbf{x}} \ln f(\mathbf{x}, \mathbf{z})][\nabla_{\mathbf{x}} \ln f(\mathbf{x}, \mathbf{z})]^T\big\} \\ &= \mathbb{E}\big\{ -\nabla_{\mathbf{x}}[\nabla_{\mathbf{x}} \ln f(\mathbf{x}, \mathbf{z})]^T\big\} \end{aligned} \tag{2.2}$$

being $\mathbb{E}\{\cdot\}$ the expected value operator, $\nabla_{\mathbf{x}}$ the gradient operator with respect to $\mathbf{x}$ and $f(\mathbf{x}, \mathbf{z})$ the joint probability density function of $\mathbf{x}$ and the measurement vector $\mathbf{z} = \mathbf{h}(\mathbf{x}) + \mathbf{n}$, being $\mathbf{n}$ a vector of Gaussian random variables with zero mean and a constant covariance matrix $\mathbf{\Sigma}$. As Bishop et al. [4] state, under the assumption of Gaussian measurement errors and deterministic $\mathbf{x}$, the joint probability function is given by

$$f(\mathbf{x}, \mathbf{z}) = \frac{1}{(2\pi)^{N/2}|\mathbf{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{z} - \mathbf{h}(\mathbf{x}))^T \mathbf{\Sigma}^{-1}(\mathbf{z} - \mathbf{h}(\mathbf{x}))\right) \tag{2.3}$$

where $|\mathbf{\Sigma}|$ is the determinant of the covariance matrix $\mathbf{\Sigma}$ and N is the number of elements of $\mathbf{z}$.

When the error covariance is independent of $\mathbf{x}$, the entire Fisher Information Matrix is simply given by

$$\mathbf{J}(\mathbf{x}) = \nabla_{\mathbf{x}}\mathbf{h}(\mathbf{x})^T \mathbf{\Sigma}^{-1} \nabla_{\mathbf{x}}\mathbf{h}(\mathbf{x}) \tag{2.4}$$

The Cramer–Rao Lower Bounds (CRLB) of the state vector components are calculated as the diagonal elements of the FIM inverse [22]

$$\text{CRLB}\{[\hat{\mathbf{x}}]_i\} = [\mathbf{J}(\mathbf{x})^{-1}]_{ii} \qquad \text{for } i = 1, 2..., n \tag{2.5}$$

2.2 Optimal Sensor-Target Geometry

2.2.1 Bearings-only Localization in 2D and 3D

In the two dimensional space, the measured azimuth bearing ϕ_i from Sensor i to the Target can be expressed by

$$\phi_i = \text{atan2}\left(\frac{t_x - s_{i_x}}{t_y - s_{i_y}}\right) = \text{atan2}\left(\frac{r_{i_x}}{r_{i_y}}\right) \tag{2.6}$$

where $\mathbf{t} = \begin{bmatrix} t_x & t_y \end{bmatrix}^T$ is the Target position, $\mathbf{s}_i = \begin{bmatrix} s_{i_x} & s_{i_y} \end{bmatrix}^T$ are the Sensor locations $\forall i \in \{1, \ldots, N \geq 2\}$, $\mathbf{r}_i = \begin{bmatrix} r_{i_x} & r_{i_y} \end{bmatrix}^T$ is the relative vector between the Sensor i and the Target and atan2 is the standard four quadrant tangent inverse. A scenario involving one Target and two bearing measurement angles ϕ_1 and ϕ_2 from Sensors 1 and 2 is illustrated in Figure 2.1.

In the 3D case, depicted in Figure 2.2, the Target position is given by $\mathbf{t} = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^T$, the Sensor i position by $\mathbf{s}_i = \begin{bmatrix} s_{i_x} & s_{i_y} & s_{i_z} \end{bmatrix}^T$, the relative vector between the Sensor i and the Target by $\mathbf{r}_i = \begin{bmatrix} r_{i_x} & r_{i_y} & r_{i_z} \end{bmatrix}^T$ and the measurement model by the azimuth angle ϕ_i and also the elevation angle ψ_i between Sensor i and the Target

$$\phi_i = \text{atan2}\left(\frac{t_x - s_{i_x}}{t_y - s_{i_y}}\right) = \text{atan2}\left(\frac{r_{i_x}}{r_{i_y}}\right) \tag{2.7}$$

$$\psi_i = \text{atan2}\left(\frac{t_z - s_{i_z}}{\sqrt{(t_x - s_{i_x})^2 + (t_y - s_{i_y})^2}}\right) = \text{atan2}\left(\frac{r_{i_z}}{\sqrt{r_{i_x}^2 + r_{i_y}^2}}\right) \tag{2.8}$$

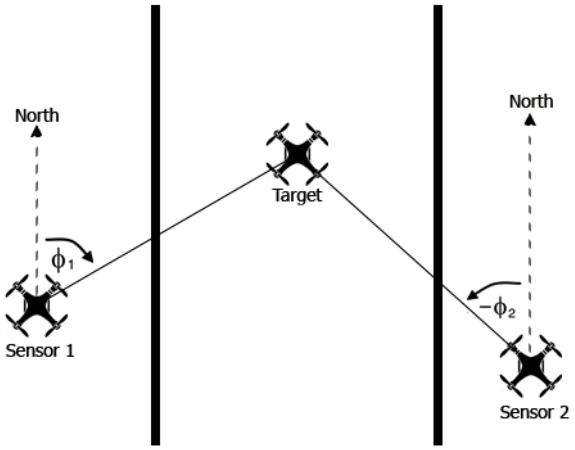


Figure 2.1: Azimuth Bearing Angles ϕ_1 and ϕ_2 between the Target and two Sensors

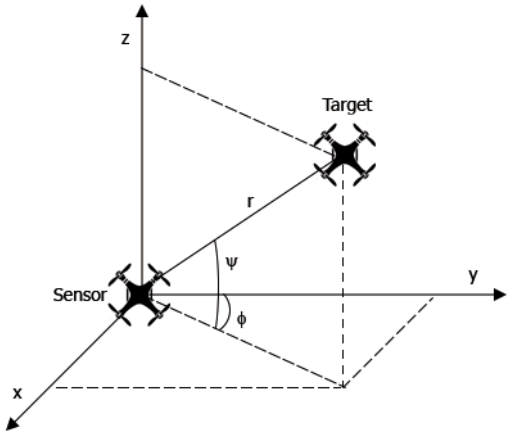


Figure 2.2: Azimuth (ϕ) and Elevation (ψ) Angles between Target and Sensor

2.2.2 Geometry of Bearings-only Localization

As computed by Bishop et al. [4], the entire Fisher Information Matrix for bearings-only localization problem in 2D is given by

$$\mathbf{J} = \frac{1}{\sigma^2} \sum_{i=1}^{N} \frac{1}{|\mathbf{r}_i|^2} \begin{bmatrix} \cos^2 \phi_i & -\frac{\sin(2\phi_i)}{2} \\ -\frac{\sin(2\phi_i)}{2} & \sin^2 \phi_i \end{bmatrix} \tag{2.9}$$

where σ is the standard deviation of the bearing measurements. To calculate the Fisher Information Matrix for the 3D case, the measurement model consisting of the two bearing angles (azimuth and elevation) is used

$$\begin{aligned}\mathbf{J} =&\frac{1}{\sigma^2}\sum_{i=1}^{N}\left(\frac{1}{|\mathbf{r}_{i_2}|^2}\begin{bmatrix}\cos^2\phi_i & -\sin\phi_i\cos\phi_i & 0\\ -\sin\phi_i\cos\phi_i & \sin^2\phi_i & 0\\ 0 & 0 & 0\end{bmatrix}+\dots\right.\\ &\left.\frac{1}{|\mathbf{r}_i|^2}\begin{bmatrix}\sin^2\phi_i\sin^2\psi_i & \sin\phi_i\cos\phi_i\sin^2\psi_i & -\sin\phi_i\sin\psi_i\cos\psi_i\\ \sin\phi_i\cos\phi_i\sin^2\psi_i & \cos^2\phi_i\sin^2\psi_i & -\cos\phi_i\sin\psi_i\cos\psi_i\\ -\sin\phi_i\sin\psi_i\cos\psi_i & -\cos\phi_i\sin\psi_i\cos\psi_i & \cos^2\psi_i\end{bmatrix}\right)\end{aligned} \tag{2.10}$$

where $|\mathbf{r}_{i_2}| = \sqrt{r_{i_x}^2 + r_{i_y}^2}$ and $|\mathbf{r}_i| = \sqrt{r_{i_x}^2 + r_{i_y}^2 + r_{i_z}^2}$.

The expressions in (2.9) and (2.10) clearly show the separation between the range and bearing information in the computation of the FIM. With the FIM in this form, it becomes explicit that the maximization of the FIM involves, amongst other, lowering the range. Also, it shows some properties such as the fact that the FIM is additive and that the FIM contributions of each Sensor can be computed separately (assuming that the Sensors are independent and uncorrelated).

2.3 Selection of an Objective Function

It is necessary to choose a measure that is related to the expected accuracy of the parameter estimates to be able to obtain the optimal placement of the Sensors and compare it with other placements. Usually, this measure is based on the Fisher Information Matrix (FIM) and the optimal placement is obtained by maximizing it. However, maximizing the Fisher Information Matrix is not an easy task and therefore a scalar function representing the FIM must be found and selected as the objective function in the optimization problem. It is inevitable that a weaker metric is obtained by reducing the FIM to a scalar function, since there is loss of information. Nevertheless, there are various choices for such a function, including the following:

- D-optimality (determinant) criterion:

$$f(\mathbf{J}) = -\ln|\mathbf{J}| \tag{2.11}$$

- E-optimality (smallest eigenvalue) criterion:

$$f(\mathbf{J}) = \lambda_{max}(\mathbf{J}^{-1}) \tag{2.12}$$

- A-optimality (trace) criterion:

$$f(\mathbf{J}) = \mathrm{tr}(\mathbf{J}^{-1}) \tag{2.13}$$

- Sensitivity criterion:

$$f(\mathbf{J}) = -\text{tr}(\mathbf{J}) \tag{2.14}$$

In order to choose a criterion, it is important to take into account not only the quantity of information each criterion provides but also the computational complexity involved when performing the optimization. For instance, the D-optimality criterion provides more geometric information for the bearings-only problem than the Sensitivity criterion, which only takes into account the trace of the FIM and loses most of its angular information. However, the trace of a matrix is simpler to compute and can be updated recursively whereas the determinant is more complex and must be recomputed every time.

The eigenvalues of the FIM are related to the uncertainty ellipsoid of the Target estimation, since the FIM is related to the estimation error covariance. More specifically, the length of each axis of the uncertainty ellipsoid is given by one over an eigenvalue squared for each eigenvalue of the FIM. Therefore, maximizing the eigenvalues of the FIM leads to a smaller uncertainty ellipsoid and a more accurate estimation.

A D-optimum design minimizes the volume of the uncertainty ellipsoid for the estimates. An E-optimum design minimizes the length of the largest axis of the same ellipsoid. An A-optimum design suppresses the average variance of the estimates. The sensitivity criterion is often used due to its simplicity. As Ponda [16] points out, the A-optimality criterion closely resembles the GDOP criteria used in GPS navigation.

2.4 Estimation Algorithms and the Posterior CRLB

The choice of an estimator depends upon many considerations. If an initial estimate of the state is available, a Bayesian approach such as the Kalman Filter can be adopted. The following Subsections outline the classical Kalman Filter and Extended Kalman Filter and describe in detail two newly designed estimators, the "Stationary Target" Estimator and the "Moving Target" Estimator.

2.4.1 Kalman Filter

The Kalman Filter (KF) is considered optimal when dealing with a linear time-invariant (LTI) system since it minimises the mean square error of the estimates based on all available measurements and considering all noise is Gaussian. A LTI system is given by the following equations

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k \tag{2.15}$$

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \tag{2.16}$$

The expression in (2.15) represents the state transition dynamics with state $\mathbf{x}_k$ and input $\mathbf{u}_k$ and (2.16) represents the measurement model with output $\mathbf{z}_k$. The process and measurement noises $\mathbf{w}_k$ and $\mathbf{v}_k$, respectively, for the sake of simplicity are assumed to be uncorrelated, Gaussian, white with zero mean and covariance $\mathbf{Q}_k$ and $\mathbf{R}_k$ (i.e. $\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k)$ and $\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k)$).

The Kalman Filter algorithm is composed of a prediction step and an update step. The prediction step involves developing a state and covariance estimate of the next time step based on the current estimates and the system dynamic model. The update step involves processing the new measurement and updating the prediction made using the new information. The equations for the KF are shown below, where $\mathbf{P}$ is the estimate covariance matrix, $\mathbf{K}$ is the optimal Kalman matrix and the notation $\hat{\mathbf{x}}_{k|k-1}$ represents the estimate of $\mathbf{x}$ at time k given the observations/measurements up to time $k-1$:

Prediction Phase:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}\mathbf{u}_k \tag{2.17}$$

$$\hat{\mathbf{z}}_{k|k-1} = \mathbf{H}\hat{\mathbf{x}}_{k|k-1} \tag{2.18}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}\mathbf{P}_{k-1|k-1}\mathbf{F}^T + \mathbf{Q}_k \tag{2.19}$$

Kalman Gain:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}^T\left[\mathbf{H}\mathbf{P}_{k|k-1}\mathbf{H}^T + \mathbf{R}_k\right]^{-1} \tag{2.20}$$

Update Phase:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1}) \tag{2.21}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k\mathbf{H}\mathbf{P}_{k|k-1} \tag{2.22}$$

2.4.2 Extended Kalman Filter

When the system is nonlinear, the Extended Kalman Filter (EKF) can be used instead of the KF. Considering a nonlinear system with dynamics

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{w}_k \tag{2.23}$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \tag{2.24}$$

where $\mathbf{f}(\mathbf{x}_k)$ and $\mathbf{h}(\mathbf{x}_k)$ are the nonlinear dynamic and measurement models of the system, respectively, and $\mathbf{w}_k$ and $\mathbf{v}_k$ are the process and measurement noises, which for the sake of simplicity are assumed to be uncorrelated, Gaussian, white with zero mean and covariance $\mathbf{Q}_k$ and $\mathbf{R}_k$ (i.e. $\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k)$ and $\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R}_k)$).

Similarly to the KF, the Extended Kalman Filter algorithm is composed of a prediction step and an update step. The prediction step involves developing a state and covariance estimate of the next time step based on the current estimates and the system dynamic model. The update step involves processing the new measurement and updating the prediction made using the new information. The equations for the EKF are shown below, where $\mathbf{P}$ is the estimate covariance matrix, $\mathbf{K}$ is the optimal Kalman matrix and the notation $\hat{\mathbf{x}}_{k|k-1}$ represents the estimate of $\mathbf{x}$ at time k given the observations/measurements up

to, and including at time $k-1$:

Prediction Phase:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{f}(\hat{\mathbf{x}}_{k-1|k-1}) \tag{2.25}$$

$$\hat{\mathbf{z}}_{k|k-1} = \mathbf{h}(\hat{\mathbf{x}}_{k|k-1}) \tag{2.26}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}_k \mathbf{P}_{k-1|k-1} \mathbf{F}_k^T + \mathbf{Q}_k \tag{2.27}$$

Kalman Gain:

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T \left[\mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R}_k \right]^{-1} \tag{2.28}$$

where $\mathbf{F}_k$ and $\mathbf{H}_k$ are the Jacobians of the system dynamic and measurement models, respectively, with respect to the state, evaluated at the predicted state $\hat{\mathbf{x}}_{k|k-1}$.

The equations for $\mathbf{F}_k$ and $\mathbf{H}_k$ are given by

$$\mathbf{F}_k = \nabla_{\mathbf{x}_k} \mathbf{f}(\mathbf{x}_k) \tag{2.29}$$

$$\mathbf{H}_k = \nabla_{\mathbf{x}_k} \mathbf{h}(\mathbf{x}_k) \tag{2.30}$$

Update Phase:

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k (\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1}) \tag{2.31}$$

$$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1} - \mathbf{K}_k \mathbf{H}_k \mathbf{P}_{k|k-1} \tag{2.32}$$

It is interesting to see that the Extended Kalman Filter covariance matrix equation has some resemblances with the inverse of the FIM, as stated by Taylor [20].

As described in [21], since $\mathbf{x}$ is now a random vector, the joint PDF becomes $f(\mathbf{x}, \mathbf{z}) = f(\mathbf{z}|\mathbf{x}) \cdot f(\mathbf{x})$ and an analogous to the CRLB, the so-called Posterior Cramer-Rao Lower Bound (PCRLB), can be defined. This new bound can also be described by (2.1) but involves a new definition of the Fisher Information Matrix $\mathbf{J}$. Considering a nonlinear model of the form given in (2.23) and (2.24), the Fisher Information Matrix can be described by

$$\mathbf{J}(\mathbf{X}_k) = \mathbf{J}_D(\mathbf{X}_k) + \mathbf{J}_P(\mathbf{X}_k) \tag{2.33}$$

where $\mathbf{X}_k = (\mathbf{x}_0, \ldots, \mathbf{x}_k)$, $\mathbf{J}_D(\mathbf{X}_k)$ represents the information obtained from the measurements, which is similar to the standard FIM, and $\mathbf{J}_P(\mathbf{X}_k)$ the *a priori* information

$$\mathbf{J}_D(\mathbf{X}_k) = \mathbb{E}\big\{ -\nabla_{\mathbf{X}_k} [\nabla_{\mathbf{X}_k} \ln f(\mathbf{z}_k|\mathbf{X}_k)]^T \big\} \tag{2.34}$$

$$\mathbf{J}_P(\mathbf{X}_k) = \mathbb{E}\big\{ -\nabla_{\mathbf{X}_k} [\nabla_{\mathbf{X}_k} \ln f(\mathbf{X}_k)]^T \big\} \tag{2.35}$$

Let $\mathbf{J}_k$ be the inverse of the right-lower block of $\mathbf{J}(\mathbf{X}_k)$, such that $\mathbf{J}_k^{-1}$ provides a lower bound on the mean square error of estimating $\mathbf{x}_k$. Then, $\mathbf{J}_k$ can be calculated using a recursive method as defined in [21]. In the end, the recursive form of the FIM becomes

$$\mathbf{J}_{k+1} = [\mathbf{Q}_k + \mathbf{F}_k \mathbf{J}_k^{-1} \mathbf{F}_k^T]^{-1} + \mathbb{E}\{\mathbf{H}_{k+1}^T \mathbf{R}_{k+1}^{-1} \mathbf{H}_{k+1}\} \tag{2.36}$$

Assuming the absence of process noise and that the Jacobian $\mathbf{H}_k$ is deterministic, the equation reduces to

$$\mathbf{J}_{k+1} = [\mathbf{F}_k^T]^{-1} \mathbf{J}_k \mathbf{F}_k^{-1} + \mathbf{H}_{k+1}^T \mathbf{R}_{k+1}^{-1} \mathbf{H}_{k+1} \tag{2.37}$$

which, as Taylor [20] claims, shows that the inverse Fisher Information matrix propagates according to the same equations as the Kalman Filter covariance matrix.

2.4.3 Nonlinear Position Estimation for Stationary Targets

In what follows, a nonlinear observer that estimates the position of a stationary target using bearings measurements is presented. The estimator, called from now on "Stationary Target" estimator, makes use of the normalized relative vector $\mathbf{u}_i = \begin{bmatrix} u_{i_x} & u_{i_y} & u_{i_z} \end{bmatrix}^T$ between the Sensor i $\mathbf{s}_i = \begin{bmatrix} s_{i_x} & s_{i_y} & s_{i_z} \end{bmatrix}^T$ and the Target $\mathbf{t} = \begin{bmatrix} t_x & t_y & t_z \end{bmatrix}^T$, described by the following

$$\mathbf{u}_i = \frac{\mathbf{t} - \mathbf{s}_i}{\|\mathbf{t} - \mathbf{s}_i\|} \tag{2.38}$$

The normalized components of the relative vector $\mathbf{u}_i$ can be written as functions of the azimuth angle ϕ_i and also the elevation angle ψ_i between Sensor i and the Target as

$$\begin{aligned} u_{i_x} &= \sin(\phi_i)\cos(\psi_i) \\ u_{i_y} &= \cos(\phi_i)\sin(\psi_i) \\ u_{i_z} &= \sin(\psi_i). \end{aligned} \tag{2.39}$$

The "Stationary Target" estimator takes advantage of some properties of the cross product between the position vectors. Since $\mathbf{u}_i$ and $(\mathbf{t} - \mathbf{s}_i)$ are collinear vectors, it follows that

$$\mathbf{u}_i \times (\mathbf{t} - \mathbf{s}_i) = 0 \tag{2.40}$$

where $\mathbf{u}_i \times (\mathbf{t} - \mathbf{s}_i)$ represents the cross product between the vectors $\mathbf{u}_i$ and $(\mathbf{t} - \mathbf{s}_i)$.

From now on, for the sake of simplicity, the Skew operator $S(\cdot)$ will be used and it is defined as

$$S(\mathbf{a}) = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{2.41}$$

where $\mathbf{a} = \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}^T$. From the definition of $S(\cdot)$, $S(\mathbf{a})\mathbf{b} = \mathbf{a} \times \mathbf{b}$.

After some algebraic manipulation, an equivalent expression involving the estimation error $\tilde{\mathbf{t}} = \hat{\mathbf{t}} - \mathbf{t}$ can be obtained as

$$S(\mathbf{u}_i)(\hat{\mathbf{t}} - \mathbf{s}_i) = S(\mathbf{u}_i)(\hat{\mathbf{t}} - \mathbf{s}_i - (\mathbf{t} - \mathbf{s}_i)) = S(\mathbf{u}_i)\tilde{\mathbf{t}} \tag{2.42}$$

This equality will be used afterwards while designing the "Stationary Target" estimator. In order to ensure the stability of the "Stationary Target" estimator, the Lyapunov stability theory will be adopted. A candidate Lyapunov function that can be used to prove the stability of the estimator is the following

$$V = \frac{1}{2}\tilde{\mathbf{t}}^T\tilde{\mathbf{t}} \tag{2.43}$$

According to Khalil [11], V is a candidate Lyapunov function if it is a locally positive-definite function, i.e.

$$V(0) = 0 \qquad \text{and } V(\tilde{\mathbf{t}}) > 0 \text{ in } D \setminus \{0\} \tag{2.44}$$

being D a domain containing $\tilde{\mathbf{t}} = 0$.

To prove that $\tilde{\mathbf{t}} = 0$ is asymptotically stable

$$\dot{V}(\tilde{\mathbf{t}}) < 0 \qquad \text{in } D \setminus \{0\} \tag{2.45}$$

The derivative of the Lyapunov function can be easily obtained

$$\dot{V} = \tilde{\mathbf{t}}^T(\dot{\hat{\mathbf{t}}} - \dot{\mathbf{t}}) \tag{2.46}$$

and if the Target is stationary at some position, $\dot{\mathbf{t}} = 0$ and the estimator takes the form $\dot{\hat{\mathbf{t}}} = -\mathbf{M}(t)\tilde{\mathbf{t}}$, then

$$\dot{V} = \tilde{\mathbf{t}}^T\dot{\hat{\mathbf{t}}} = -\tilde{\mathbf{t}}^T\mathbf{M}(t)\tilde{\mathbf{t}} \tag{2.47}$$

if $\mathbf{M}(t) \succ 0$.

In order to design an estimator with a positive definite matrix $\mathbf{M} \succ 0$, the Skew properties will be explored.

$$\sum_{i=1}^{N} S(\mathbf{u}_i)^T S(\mathbf{u}_i) = -\sum_{i=1}^{N} S(\mathbf{u}_i)^2 \succeq \mathbf{0} \tag{2.48}$$

The expression in (2.48) shows that $-\sum_{i=1}^{N} S(\mathbf{u}_i)^2$ is at least positive semi-definite matrix for all $\mathbf{u}_i$. Nevertheless, it is possible to prove that this matrix is in fact positive definite under some conditions. It is then necessary to find the non-null vectors $\mathbf{x}$ that solve the following

$$\mathbf{x}^T \sum_{i=1}^{N} S(\mathbf{u}_i)^2 \mathbf{x} = 0 \tag{2.49}$$

For example, for $N = 2$:

$$\mathbf{x}^T \left(S(\mathbf{u}_1)^2 + S(\mathbf{u}_2)^2 \right) \mathbf{x} = 0 \tag{2.50}$$

which leads to:

$$\mathbf{x}^T S(\mathbf{u}_1)^2 \mathbf{x} = -\mathbf{x}^T S(\mathbf{u}_2)^2 \mathbf{x} \tag{2.51}$$

Since $\mathbf{x}^T S(\mathbf{u}_1)^2 \mathbf{x} \leq 0$ and $-\mathbf{x}^T S(\mathbf{u}_2)^2 \mathbf{x} \geq 0$, the only solution of (2.51) occurs when vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ are collinear, meaning that $-(S(\mathbf{u}_1)^2 + S(\mathbf{u}_2)^2)$ is a positive definite matrix unless the Target and Sensors positions are all collinear. For that reason, a possible expression for the dynamics of the estimator is

$$\dot{\hat{\mathbf{t}}} = (S(\mathbf{u}_1)^2 + S(\mathbf{u}_2)^2)\tilde{\mathbf{t}} \tag{2.52}$$

This can be extended to N Sensor measurements and in the end, using the deduction previously made in (2.42), an expression for a convergent estimator is

$$\dot{\hat{\mathbf{t}}} = \sum_{i=1}^{N} S(\mathbf{u}_i)^2 (\hat{\mathbf{t}} - \mathbf{s}_i) \tag{2.53}$$

When working on discrete time, consider the following estimator

$$\hat{\mathbf{t}}_{k+1} = \hat{\mathbf{t}}_k + \sum_{i=1}^{N} \alpha_i S(\mathbf{u}_{i_k})^2 \tilde{\mathbf{t}}_k \tag{2.54}$$

where $\alpha_i > 0$ is a constant value limited by some restrictions which will be described later in this section. Assuming the same scenario analysed before where the Target is stationary and therefore $\mathbf{t}_{k+1} = \mathbf{t}_k = \mathbf{t}$, the following equation is obtained for the estimation error

$$\tilde{\mathbf{t}}_{k+1} = \hat{\mathbf{t}}_{k+1} - \mathbf{t} \tag{2.55}$$

Using (2.54), this equation can be rewritten as

$$\tilde{\mathbf{t}}_{k+1} = \left(\mathbf{I} + \sum_{i=1}^{N} \alpha_i S(\mathbf{u}_{i_k})^2 \right) \tilde{\mathbf{t}}_k = \mathbf{A}_k \tilde{\mathbf{t}}_k \tag{2.56}$$

Considering that the Lyapunov function for this discrete case is the same as the one used in (2.43)

$$V_{k+1} - V_k = -\tilde{\mathbf{t}}_k^T \left(\mathbf{I} - \mathbf{A}_k^T \mathbf{A}_k \right) \tilde{\mathbf{t}}_k < 0 \tag{2.57}$$

which can only be achieved if

$$\mathbf{I} - \mathbf{A}_k^T \mathbf{A}_k \succ \mathbf{0} \tag{2.58}$$

After some algebraic manipulation, it is possible to rewrite this equation

$$\mathbf{I} - \mathbf{A}_k^T \mathbf{A}_k = 2\mathbf{Q}_k - \mathbf{Q}_k^2 \succ \mathbf{0} \tag{2.59}$$

where

$$\mathbf{Q}_k = -\sum_{i=1}^{N} \alpha_i S(\mathbf{u}_{i_k})^2 \succ \mathbf{0} \tag{2.60}$$

The range of the eigenvalues $\lambda(\mathbf{Q}_k)$ can be easily obtained taking into account the inequalities in (2.59) and (2.60):

$$0 < \lambda(\mathbf{Q}_k) < 2 \tag{2.61}$$

Since

$$\lambda_{min}(\mathbf{Q}_k) \leq \lambda_{max}(\mathbf{Q}_k) < 2 \tag{2.62}$$

and

$$\lambda_{max}(\mathbf{Q}_k) = \sum_{i=1}^{N} \alpha_i \tag{2.63}$$

the following restriction is obtained for the values of α_i

$$\sum_{i=1}^{N} \alpha_i < 2 \qquad \text{for } \alpha_i > 0,\ i = 1, ..., N \tag{2.64}$$

The possibility of giving different weights to each measurement can be very convenient in a situation where one measurement is more accurate than another. Also, the ability to update iteratively the values of α_i depending on some function of the distance between the Target and the Sensor can be taken into account in order to produce better estimation results. However, for the sake of simplicity, a constant value of α is used for every measurement. For instance, if N=2 Sensors, then $\alpha < 1$.

The main limitation of this estimator is the fact that it is constructed under the assumption that $\dot{\mathbf{t}} = 0$ and therefore can only be used with stationary Targets. Nevertheless, it is a globally asymptotically stable estimator that will be used in some scenarios where the Target is standing still.

2.4.4 "Moving Target" Estimator

Another estimator that takes advantage of the cross product properties between the position vectors is the "Moving Target" estimator. This estimator will be used for the second-order system described in Section 3.1, where the control law $\ddot{\mathbf{t}} = \mathbf{u}$ is applied. In this case, the system state space representation is the following

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}_c\mathbf{x} + \mathbf{B}_c\mathbf{u} \\ \mathbf{y} &= \mathbf{C}_c\mathbf{x} \end{aligned} \tag{2.65}$$

where

$$\mathbf{x} = \begin{bmatrix} \mathbf{t} \\ \dot{\mathbf{t}} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \tag{2.66}$$

and

$$\mathbf{A}_c = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \qquad \mathbf{B}_c = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \qquad \mathbf{C}_c = \begin{bmatrix} -S(\mathbf{u}_1)^2 & \mathbf{0} \\ -S(\mathbf{u}_2)^2 & \mathbf{0} \end{bmatrix} \tag{2.67}$$

Matrix $\mathbf{C}_c$ is chosen in such a way that evokes the properties of the Skew-symmetric matrix operator already scrutinized in Section 2.4.3. This form of matrix $\mathbf{C}_c$ will prove to be crucial in order to construct a well-designed estimator, which will be called throughout the text as "Moving Target" estimator.

The estimator chosen has the following structure

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}_c\hat{\mathbf{x}} + \mathbf{B}_c\mathbf{u} - \mathbf{L}\left(\mathbf{y} - \mathbf{C}_c\hat{\mathbf{x}}\right) \tag{2.68}$$

where $\mathbf{L}$ is a gain matrix given by

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_1 & \mathbf{L}_1 \\ \mathbf{L}_2 & \mathbf{L}_2 \end{bmatrix} \tag{2.69}$$

For the sake of simplicity, $\mathbf{L}_1$ and $\mathbf{L}_2$ are considered identity matrices with associated constant positive gains l_1 and l_2, respectively, i.e.

$$\mathbf{L}_1 = \begin{bmatrix} l_1 & 0 & 0 \\ 0 & l_1 & 0 \\ 0 & 0 & l_1 \end{bmatrix} = l_1\mathbf{I} \qquad \text{and} \qquad \mathbf{L}_2 = \begin{bmatrix} l_2 & 0 & 0 \\ 0 & l_2 & 0 \\ 0 & 0 & l_2 \end{bmatrix} = l_2\mathbf{I} \tag{2.70}$$

In this estimator, the role of the corrective term $\mathbf{L}\left(\mathbf{y} - \mathbf{C}_c\hat{\mathbf{x}}\right)$ is to minimize the error between the true and the estimated state vector. Taking into account (2.66), it is possible to expand (2.68), leading to the following expressions

$$\begin{cases} \dot{\hat{\mathbf{x}}}_1 = \hat{\mathbf{x}}_2 + l_1 \sum_{i=1}^{2} S(\mathbf{u}_i)^2(\hat{\mathbf{x}}_1 - \mathbf{x}_1) \\ \dot{\hat{\mathbf{x}}}_2 = \mathbf{u} + l_2 \sum_{i=1}^{2} S(\mathbf{u}_i)^2(\hat{\mathbf{x}}_1 - \mathbf{x}_1) \end{cases} \tag{2.71}$$

Bearing in mind the deduction made in (2.42), this can also be written as

$$\begin{cases} \dot{\hat{\mathbf{x}}}_1 = \hat{\mathbf{x}}_2 + l_1 \sum_{i=1}^{2} S(\mathbf{u}_i)^2(\hat{\mathbf{x}}_1 - \mathbf{s}_i) \\ \dot{\hat{\mathbf{x}}}_2 = \mathbf{u} + l_2 \sum_{i=1}^{2} S(\mathbf{u}_i)^2(\hat{\mathbf{x}}_1 - \mathbf{s}_i) \end{cases} \tag{2.72}$$

which ensures that the estimator is indeed designed without having knowledge of the true location $\mathbf{t}$ of the Target. Instead, it uses the known position coordinates of the Sensors and also the bearing information given by the normalized relative vectors $\mathbf{u}_1$ and $\mathbf{u}_2$.

To choose the values in matrix $\mathbf{L}$, the evolution of the estimation error defined as $\tilde{\mathbf{x}} = \hat{\mathbf{x}} - \mathbf{x}$ needs to be examined

$$\begin{cases} \dot{\tilde{\mathbf{x}}}_1 = \tilde{\mathbf{x}}_2 + l_1 \sum_{i=1}^{2} S(\mathbf{u}_i)^2\tilde{\mathbf{x}}_1 \\ \dot{\tilde{\mathbf{x}}}_2 = l_2 \sum_{i=1}^{2} S(\mathbf{u}_i)^2\tilde{\mathbf{x}}_1 \end{cases} \tag{2.73}$$

A change of coordinates $\xi = \tilde{\mathbf{x}}_1 - \frac{l_1}{l_2}\tilde{\mathbf{x}}_2$ is adopted in order to simplify the future calculations of the $\mathbf{L}$ matrix gains which drive to the estimator's stability. Consequently, the estimation error dynamics can be written as

$$\begin{cases} \dot{\tilde{\mathbf{x}}}_1 = \left(l_1 \sum_{i=1}^{2} S(\mathbf{u}_i)^2 + \frac{l_2}{l_1}\mathbf{I} \right) \tilde{\mathbf{x}}_1 - \frac{l_2}{l_1}\xi \\ \dot{\xi} = \tilde{\mathbf{x}}_2 = \frac{l_2}{l_1}\tilde{\mathbf{x}}_1 - \frac{l_2}{l_1}\xi \end{cases} \tag{2.74}$$

To ensure the stability of the estimator, the Lyapunov theory will be again used, as previously done in Section 2.4.3. A possible Lyapunov function for this case is the following

$$V = \frac{1}{2}\xi^T\xi + \frac{1}{2}\tilde{\mathbf{x}}_1^T\tilde{\mathbf{x}}_1 \tag{2.75}$$

which is a locally positive-definite function, since $V(0) = 0$ and $V(\tilde{\mathbf{x}}_1, \xi) > 0$ in $D \setminus \{(0,0)\}$. Its derivative is then given by

$$\dot{V} = -\frac{l_2}{l_1}\xi^T\xi + \tilde{\mathbf{x}}_1^T \left(l_1 \sum_{i=1}^{2} S(\mathbf{u}_i)^2 + \frac{l_2}{l_1}\mathbf{I} \right) \tilde{\mathbf{x}}_1 \tag{2.76}$$

which should be a negative-definite function in $D \setminus \{(0,0)\}$. The first term on the right side of (2.76) is always negative definite since both l_1 and l_2 are positive gains. To guarantee the negative definiteness of the second term the following holds

$$l_1 \sum_{i=1}^{2} S(\mathbf{u}_i)^2 + \frac{l_2}{l_1}\mathbf{I} \prec \mathbf{0} \tag{2.77}$$

Since

$$\sum_{i=1}^{2} S(\mathbf{u}_i)^2 \preceq \lambda_{max}(\sum_{i=1}^{2} S(\mathbf{u}_i)^2)\mathbf{I} \prec -\frac{l_2}{{l_1}^2}\mathbf{I} \tag{2.78}$$

and

$$\lambda_{max}(\sum_{i=1}^{2} S(\mathbf{u}_i)^2) = 0 \tag{2.79}$$

apparently there is no possible choice of gains l_1 and l_2 that satisfies the conditions imposed in (2.78). However, as mentioned before, the maximum eigenvalue of $\sum_{i=1}^{2} S(\mathbf{u}_i)^2$ is only equal to zero when the $\mathbf{u}_1$ and $\mathbf{u}_2$ vectors are collinear, i.e., when the Sensors-Target geometric configuration is of 180°. This maximum eigenvalue monotonically decreases until the formation assumes a geometric configuration of 90°, starting to increase monotonically until it reaches a hypothetical 0° configuration, where the measurements are also collinear and thus the maximum eigenvalue is again equal to zero. Therefore, a limit can be imposed to the geometric configuration so that an acceptable maximum eigenvalue is obtained. This way, a limit of 140°/40° geometric configuration is a reasonable choice given the results presented later in Subsection 2.5.1. This maximum/minimum angle configuration leads to

$$\lambda_{max(140^o/40^o)}(\sum_{i=1}^{2} S(\mathbf{u}_i)^2) = -0.234 \tag{2.80}$$

For this reason, the restrictions for choosing l_1 and l_2, given a maximum/minimum configuration of $140^o/40^o$, can be rewritten as

$$l_2 < 0.234\ {l_1}^2 \tag{2.81}$$

Note that a more flexible approach to impose the limits on the maximum/minimum angle configuration could be adopted. However, in such situation the maximum eigenvalue would be closer to zero and thus the range of possible l_1 and l_2 values would be smaller.

Even though this estimator is more complex than the "Stationary Target" estimator described in Subsection 2.4.3, its main advantage lies in the fact that it estimates both position and velocity and thus can be used with moving Targets.

2.5 Results

2.5.1 Optimal Geometry Including Physical Constraints

The first step in order to calculate the optimal geometry is to include the physical constraints imposed by the bridge, since the Sensors will not receive their position information through the GPS signal if they are too close to the bridge. Another constraint taken into account for this optimization problem is the fact that the two Sensors are on opposite sides of the bridge.

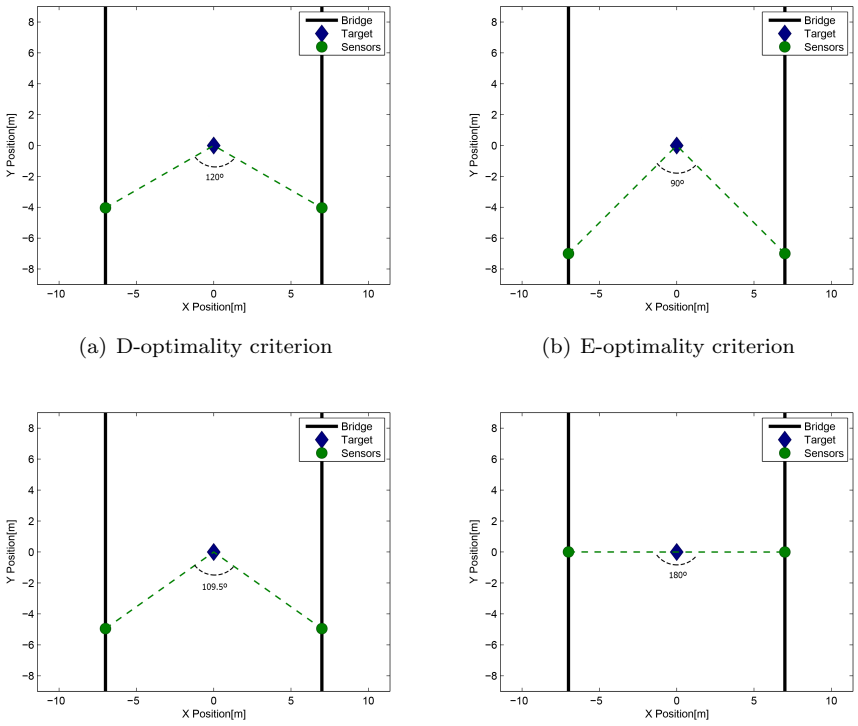
At the beginning, the optimization problem is solved by calculating the position of the two Sensors in a Cartesian coordinate system with the y axis parallel to the limits of the bridge. Using the D-optimality criterion described in Subsection 2.3, one solution is found in which the angle between the Sensors is 120° , instead of the 90° formation deduced by Bishop et al. [4]. This difference can be easily explained by considering the fact that the constraints imposed by the bridge affect the optimal solution. Another result that strikes after several experiments with different initial conditions is that the x value of the Sensors' positions is always the same, being exactly equal to the constraint values representing the edges of the bridge. Therefore, in order to simplify the optimization problem, in the following optimizations the only variables to be optimized are the y value of the Sensors' positions, being x fixed and equal to the boundary values of the constraint imposed by the bridge.

After eliminating the x dependency of the optimization problem, the different criteria described in Subsection 2.3 are tested and the corresponding uncertainty ellipses and areas are calculated to obtain a quantitative comparison. The results obtained can be seen in Table 2.1, where each uncertainty ellipse area is divided by A_D, the area obtained for the D-optimality criterion. Also, the uncertainty ellipses obtained using the D-optimality, A-optimality and E-optimality criteria are represented in Figure 2.4. Since the Sensitivity criterion is associated with a much bigger area of uncertainty, its representation is not included in Figure 2.4. An illustration of the different configurations obtained with the four criteria can be seen in Figure 2.3, where the bridge has a width of 14 meters and the Target is in its middle.

From the results presented in Table 2.1 it is possible to conclude that depending on the chosen criterion, the geometry of the Sensors will change. While for the D-optimality criterion the angle between

Table 2.1: Results obtained after optimization process for the different criteria

Criterion	Angle between Sensors(°)	Uncertainty Ellipse Area($/A_D$)
D-optimality	120°	1
E-optimality	90°	2.84
A-optimality	109.5°	1.14
Sensitivity	180°	1.05×10^{15}



(a) D-optimality criterion

(b) E-optimality criterion

(c) A-optimality criterion

(d) Sensitivity criterion

Figure 2.3: Geometric Configurations obtained with different criteria

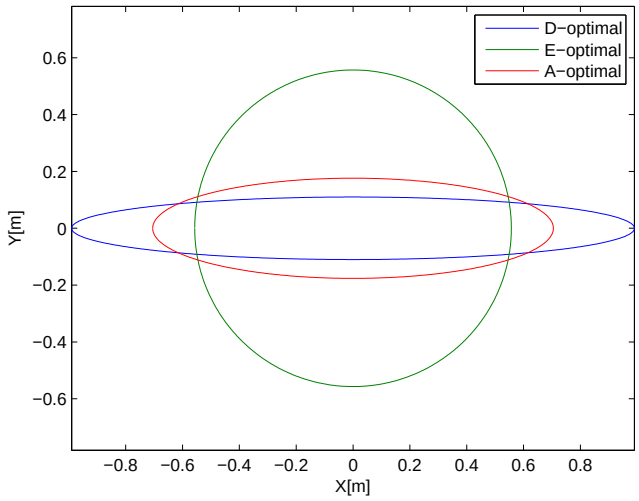


Figure 2.4: Uncertainty ellipses for D-optimality, E-optimality and A-optimality criteria

the Sensors obtained is 120°, as it was concluded before, the E-optimality criterion leads to a 90° configuration. Considering that the optimization process using the E-optimality criterion involves finding the minimum value of the length of the largest axis of the uncertainty ellipse, it is reasonable to think that this criterion will lead to an uncertainty ellipse closer to a circle, as it can be confirmed in Figure 2.4. Even though an ellipse closer to a circle is preferable than a flatter one, because it is undesirable to have a large discrepancy between the two axes of uncertainty, the area of the ellipses must also be taken into account and in this case it is considerably higher than the uncertainty ellipse area obtained with the D-optimality criterion (almost 3 times higher).

The A-optimality criterion is designed to restrain the average variance of the estimates. It can be seen as an intermediate criterion between the D-optimality and E-optimality criteria since the angle between the Sensors obtained is roughly 109.5°. The uncertainty ellipse is less flat but has a slightly higher area when comparing with the D-optimality criterion (about 1.14 times higher).

As far as the Sensitivity criterion is concerned, the configuration obtained is 180°, meaning the Sensors are collinear with the Target. Since the objective function is the negative trace of the FIM, it becomes clear from (2.9) that all the angular information is neglected, being the distance between the Sensors and the Target the only parameter to be minimized. Even though it is the easiest to compute, this criterion is by far the worst and might be used only to initialize an optimization problem.

2.5.2 Optimal Geometry with Stationary Target

In this section, the several criteria described above are tested in an estimation process and compared with a simple case where no geometric optimization is made. In this additional case, a constant y offset between the Target and the Sensors is used, i.e., the Sensors follow the Target from behind, being always in the same horizontal line. This constant offset y is equal to half of the bridge's width, as shown in Figure 2.5.

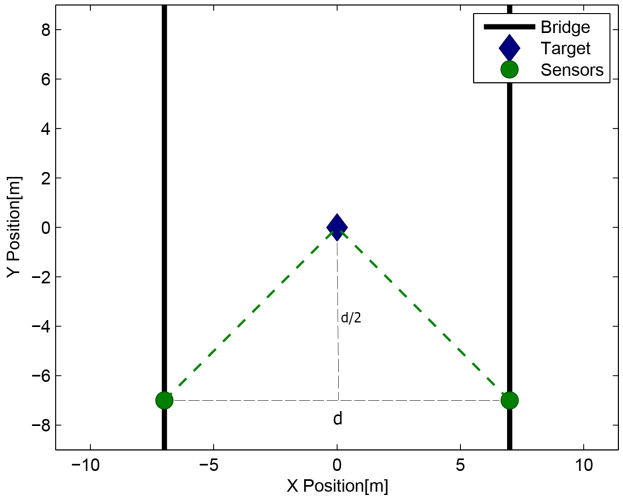


Figure 2.5: Geometric Configuration obtained with "Constant Offset" Criterion

To begin with, a scenario where the Target is standing in a stationary position inside the bridge limits is studied. For this scenario, the position needs to be chosen so that it prevents obtaining the same results for different criterion. For example, when the Target is standing exactly in the middle of the bridge, the E-optimality criterion and the newly introduced "Constant Offset" criterion produce the same 90^{o} angle Sensors-Target configuration, as shown in Figures 2.3(b) and 2.5. However, they are different criteria since the E-optimality criterion always brings a 90^{o} angle configuration no matter the position of the Target, whilst the "Constant Offset" criterion produces various angular configurations depending on the x position of the Target, only guaranteeing that the Sensors are always at a constant y distance behind the Target. For that reason, a scenario where the Target is standing three meters to the right of the centre of the bridge is chosen, so that no criterion overlaps another.

After averaging over 20 Monte-Carlo (MC) runs with different initial estimates of the state, the performance of the the D-optimality, E-optimality, A-optimality and "Constant Offset" criteria are compared based on the Root-Mean-Squared Error (RMSE) between the true and the estimated Target position in Table 2.2. Since the Sensitivity criterion produce far worse results than the other four, it is going to be dismissed from now on. For all the simulations, the "Moving Target" estimator is used. The estimation and optimization algorithms run in sequence and are updated in real-time. The theoretical Cramer-Rao Lower Bounds in the first 2 seconds of the estimation process are shown in Figure 2.6 for the 4 criteria.

As far as the CRLBs are concerned, it is possible to conclude that the CRLB for the D-optimality criterion is the smallest of all in the y axis but it is the highest in the x axis. This is in accordance with the uncertainty ellipses shown in Figure 2.4, in which the D-optimality criterion ellipse is the flattest, meaning that it has a high uncertainty in one of the axis and a very low uncertainty on the other. Since the E-optimality criterion leads to a rounder ellipse of uncertainty, it is plausible that the Cramer-Rao lower bounds on x and y axes are similar, as it can be seen in Figure 2.6. The A-optimality criterion presents low and not very discrepant Cramer-Rao lower bounds when comparing with, for example, the D-optimality criterion. Finally, the "Constant Offset" criterion presents the highest Cramer-Rao lower bound in the y axis and the lowest in the x axis, along with the A-optimality criterion.

In addition to this, a relation between the geometric configurations depicted in Figure 2.3 and the theoretical CRLB can be made. For instance, the fact that the D-optimality criterion leads to a 120^{o} angle configuration suggests that the uncertainty and thus the CRLB in the x axis will be greater when comparing with other configurations closer to orthogonality, such as the ones obtained using the E-optimality criterion. On the other hand, the uncertainty in the y axis will be smaller, as confirmed in Figure 2.6. Nevertheless, besides the configuration angle, the distance is also an important factor which affects the CRLB in every criterion and must be taken into account when analysing the results obtained. For example, the fact that the A-optimality criterion leads to a configuration of 109.5^{o} and the E-optimality to an orthogonal geometry would hint that the CRLB in the x axis of the 109.5^{o} configuration would be higher. However, due to the combined effect of distance and angular configuration, the CRLB is indeed lower for the A-optimality criterion possibly due to the smaller distance between the Sensors and the Target, as corroborated by Figure 2.6.

Comparing the theoretical CRLB in Figure 2.6 and the RMSE obtained in Table 2.2, it is possible

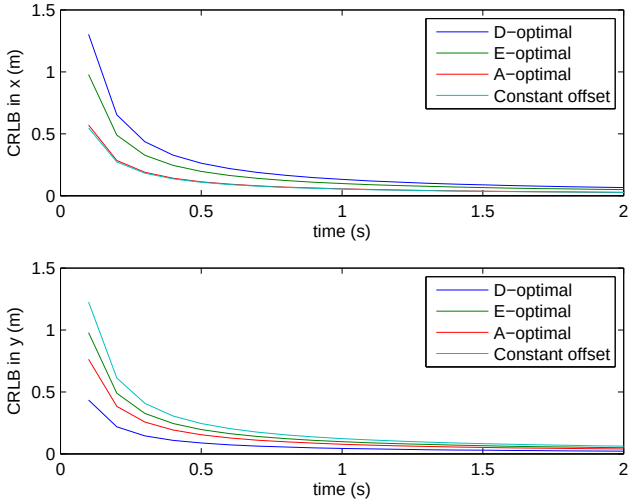


Figure 2.6: Cramer-Rao Lower Bounds (CRLB) in x (on top) and y (on the bottom) for different criteria

Table 2.2: Root-Mean-Squared Error (RMSE) in x and y for different criteria

Criterion	RMSE in x (mm)	RMSE in y (mm)
D-optimality	34.5282	18.6838
E-optimality	19.9570	20.4307
A-optimality	23.8740	17.7977
Constant Offset	19.5138	23.2420

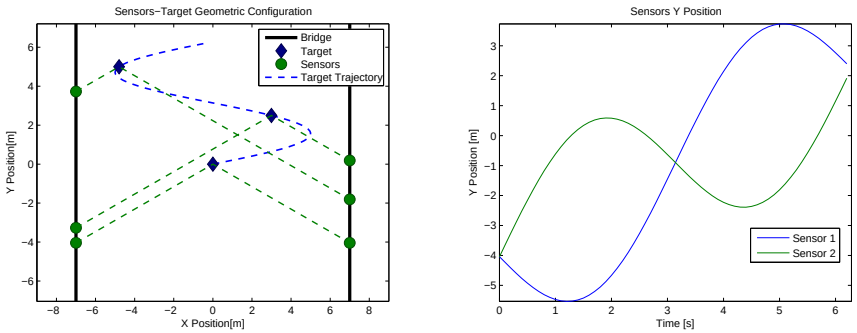
to find close resemblances which point out that the CRLB is indeed a good indicator to evaluate the performance of each criterion in the estimation process. For instance, the higher RMSE of the D-optimality and "Constant Offset" criteria in x and y axes, respectively, are also verified in terms of CRLB, in Figure 2.6. In Table 2.2, it is clear that the "Constant Offset" and the E-optimality criteria lead to the lowest RMSE in the x axis, while the D- and A-optimality criteria produce better results in the y axis. The RMSE can also be explained by analysing the different angle configurations obtained for each criterion. For example, the E-optimality criterion leads to an orthogonal measurement configuration which gives equal importance to the uncertainty on the x and y axis, as previously pictured in Figure 1.2(b). This justifies the closeness between the RMSE in the two axes for this criterion. On the other hand, the D- and A-optimality criteria lead to non-orthogonal angle configurations which contribute to a lower uncertainty in a certain axis, in this case the y axis, and a higher in the other one, as shown in Figure 1.2(a). The RMSE obtained in Table 2.2 for these two criteria are clearly consistent with this, as they are higher in the x axis than the other criteria and the lowest in the y axis. Since the "Constant Offset" criterion produces a non-orthogonal configuration with an angle below 90°, it has a lower uncertainty in the x axis and a higher one in the y axis when comparing with the other criteria, as confirmed by the RMSE presented in Table 2.2. All things considered, it is hard to come with an overall best criterion given the conflicting effects on the x and y axes. Nevertheless, it is irrefutable that the A-optimality produces better results than the D-optimality criterion in both axes. Also, the E-optimality seems to be advantageous relative to the "Constant Offset" criterion when comparing the differences between their

RMSE in both axes.

2.5.3 Optimal Geometry with Moving Target

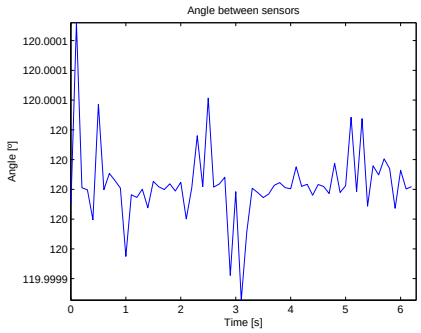
In this Subsection, a sinusoidal trajectory of the Target is designed and the several criteria already described in Subsection 2.3 are tested, being the results illustrated in Figures 2.7, 2.8, 2.9, 2.10 and 2.11 for the optimal and the "Constant Offset" criteria. The sinusoidal trajectory is designed so that the Target never gets too close to the borders of the bridge, having a two meter safety distance. The simulation ends after the Target completes a full sinusoidal cycle, during 2π seconds. The Sensors-Target geometric configurations at 3 time instants are shown for each criterion in Figures 2.7(a), 2.8(a), 2.9(a), 2.10(a) and 2.11(a). In Figures 2.7(b), 2.8(b), 2.9(b), 2.10(b) and 2.11(b) the y positions of both Sensor 1 (on the left side of the bridge) and Sensor 2 (on the right side) are represented for the correspondent criterion. Finally, in Figures 2.7(c), 2.8(c), 2.9(c), 2.10(c) and 2.11(c) the angles subtended at the Target by the two Sensors during the simulation time are depicted. The small variance visible on some of the angle plots is due to computational errors in the optimization process.

As far as the D-optimality criterion is concerned, it can be seen in Figure 2.7(b) that both Sensors need to go back and forth in the y axis in order to maintain the 120^o geometric configuration shown in Figure 2.7(c). A similar situation also happens for the E-optimality criterion in Figure 2.8, in which the Sensors maintain a constant 90^o geometric configuration by adjusting their y position as the Target moves in a sinusoidal trajectory. The A-optimality criterion produces a more flexible geometric configuration since the angle subtended at the Target by the two Sensors varies from 102.5^o to 109.5^o depending on the x location of the Target, as shown in Figure 2.8(c). The Sensitivity criterion leads to a geometric configuration where the Sensors follow in line with the Target in a 180^o geometric configuration, having always the same y coordinates of the Target, as illustrated in Figure 2.10. Lastly, the "Constant Offset" criterion, which, contrarily to the other criteria, does not involve optimizing any function derived from the FIM, produces a geometric configuration in which the Sensors follow the Target at a fixed y distance from it. Figure 2.11(c) shows that the angle of the Sensors-Target configuration varies between approximately 76^o and 90^o.



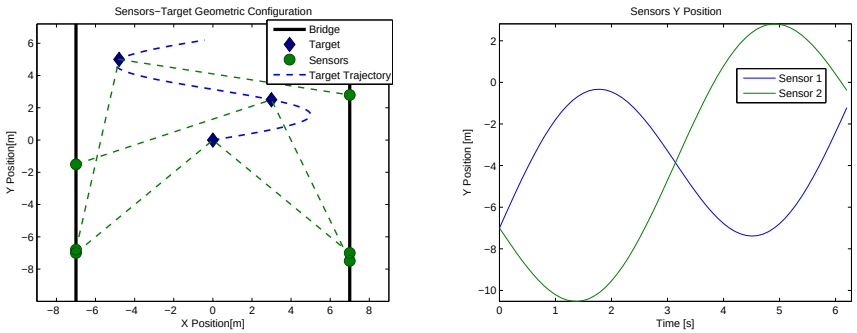
(a) Configuration as the Target moves

(b) Sensors y position



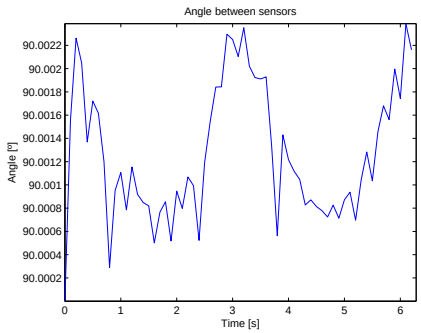
(c) Angle between Sensors and Target

Figure 2.7: Optimal Geometry obtained with the D-optimality criterion for a Target Sinusoidal Trajectory



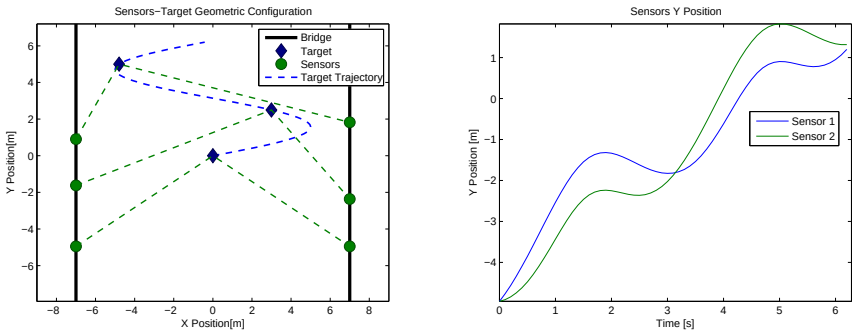
(a) Configuration as the Target moves

(b) Sensors y position



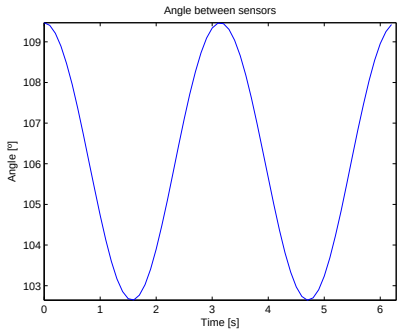
(c) Angle between Sensors and Target

Figure 2.8: Optimal Geometry obtained with the E-optimality criterion for a Target Sinusoidal Trajectory



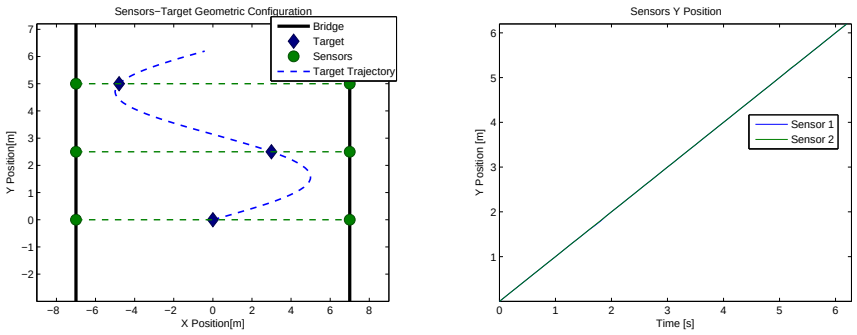
(a) Configuration as the Target moves

(b) Sensors y position



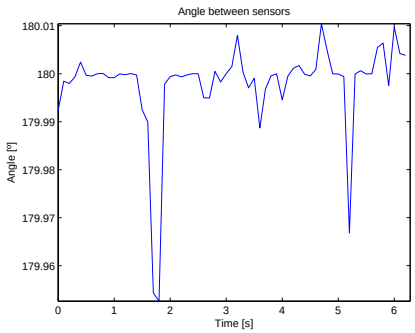
(c) Angle between Sensors and Target

Figure 2.9: Optimal Geometry obtained with the A-optimality criterion for a Target Sinusoidal Trajectory



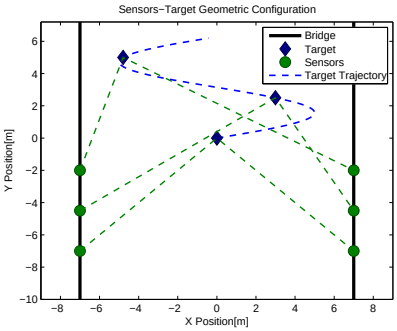
(a) Configuration as the Target moves

(b) Sensors y position

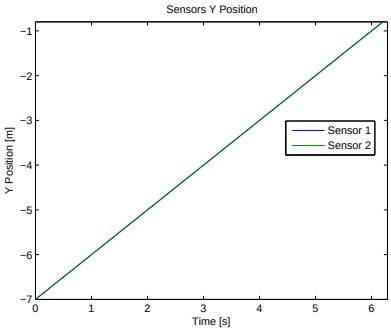


(c) Angle between Sensors and Target

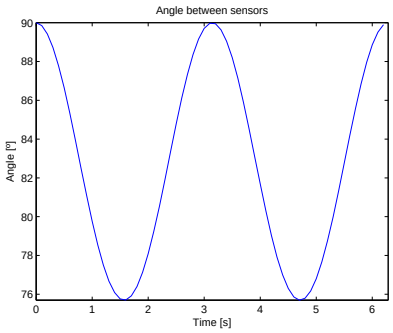
Figure 2.10: Optimal Geometry obtained with the Sensitivity criterion for a Target Sinusoidal Trajectory



(a) Configuration as the Target moves



(b) Sensors y position



(c) Angle between Sensors and Target

Figure 2.11: Optimal Geometry obtained with the "Constant Offset" criterion for a Target Sinusoidal Trajectory

Chapter 3

Target Trajectory Control

In this Chapter, a controller is combined with the nonlinear estimator previously proposed to drive the target vehicle along a desired trajectory. First, the system and estimator dynamics are explained and implemented in a Simulink model along with a controller in order to control the Target's position over time. Then, the results obtained both for a stationary and a moving Target are presented and carefully analysed.

3.1 Target Position Control

Assuming that the target dynamics is described by a double integrator driven by an input $\mathbf{u}$, target trajectory tracking can be accomplished with a second order control system. The continuous system state space representation can be written as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}_c\mathbf{x} + \mathbf{B}_c\mathbf{u} \\ \mathbf{y} &= \mathbf{C}_c\mathbf{x} \end{aligned} \tag{3.1}$$

where

$$\mathbf{x} = \begin{bmatrix} \mathbf{t} \\ \dot{\mathbf{t}} \end{bmatrix} \tag{3.2}$$

and

$$\mathbf{A}_c = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \qquad \mathbf{B}_c = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} \qquad \mathbf{C}_c = \begin{bmatrix} -S(\mathbf{u}_1)^2 & \mathbf{0} \\ -S(\mathbf{u}_2)^2 & \mathbf{0} \end{bmatrix} \tag{3.3}$$

The output matrix $\mathbf{C}_c$ uses the Skew operator $S(\cdot)$ of the normalized relative vectors between the Target and the 2 Sensors ($\mathbf{u}_1$ and $\mathbf{u}_2$). Since the system is a non-linear one, it is not possible to use a Kalman Filter and implementing an Extended Kalman Filter requires more computational resources and only guarantees local convergence, thus being subject to the quality of initial estimates. For that reason,

the estimator which will be used is the "Moving Target" estimator, already described in Subsection 2.4.4.

The "Moving Target" estimator has the following structure

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}_c\hat{\mathbf{x}} + \mathbf{B}_c\mathbf{u} - \mathbf{L}\left(\mathbf{y} - \mathbf{C}_c\hat{\mathbf{x}}\right) \tag{3.4}$$

where $\mathbf{L}$ is a gain matrix given by

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_1 & \mathbf{L}_1 \\ \mathbf{L}_2 & \mathbf{L}_2 \end{bmatrix} \tag{3.5}$$

where $\mathbf{L}_1 = l_1\mathbf{I}$, $\mathbf{L}_2 = l_2\mathbf{I}$ and with $l_2 < 0.234\ {l_1}^2$.

If direct measures of $\mathbf{t}$ and $\dot{\mathbf{t}}$ were available, a simple proportional derivative controller together with a feedforward term $\ddot{\mathbf{t}}^*$ would accomplish the trajectory tracking goal, yielding

$$\ddot{\mathbf{t}} = \mathbf{u} = -\mathbf{K}_P(\mathbf{t} - \mathbf{t}^*) - \mathbf{K}_D(\dot{\mathbf{t}} - \dot{\mathbf{t}}^*) + \ddot{\mathbf{t}}^* \tag{3.6}$$

Substituting $\ddot{\mathbf{t}}^* = \mathbf{u}^*$, (3.6) can be written in state space form as

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\mathbf{x} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix}\left(-\begin{bmatrix} \mathbf{K}_P & \mathbf{K}_D \end{bmatrix}(\mathbf{x} - \mathbf{x}^*) + \mathbf{u}^*\right) \tag{3.7}$$

which results in a error dynamics

$$\dot{\mathbf{x}}_e = (\mathbf{A}_c - \mathbf{B}_c\mathbf{K}_{LQR})\mathbf{x}_e \tag{3.8}$$

where $\mathbf{x}_e = \mathbf{x} - \mathbf{x}^*$ and $\mathbf{K}_{LQR} = \begin{bmatrix} \mathbf{K}_P & \mathbf{K}_D \end{bmatrix}$.

Then, a LQR problem can be solved to obtain an optimal gain matrix $\mathbf{K}_{LQR}$ that minimizes the cost functional

$$\varphi = \int_0^T \left((\mathbf{x}(\tau) - \mathbf{x}^*(\tau))^T\mathbf{Q}_{LQR}(\mathbf{x}(\tau) - \mathbf{x}^*(\tau)) + (\mathbf{u}(\tau) - \mathbf{u}^*(\tau))^T\mathbf{R}_{LQR}(\mathbf{u}(\tau) - \mathbf{u}^*(\tau))\right)\mathrm{d}\tau \tag{3.9}$$

Matrices $\mathbf{Q}_{LQR}$ and $\mathbf{R}_{LQR}$ can be seen as the weights given on the states and on the control inputs, respectively, when minimizing the cost function. The MATLAB function `lqr`($\mathbf{A}_c$,$\mathbf{B}_c$,$\mathbf{Q}_{LQR}$,$\mathbf{R}_{LQR}$) calculates straightforward the optimal gain matrix $\mathbf{K}_{LQR}$ and it is used in this work.

Given that only estimates are available, the control law is redefined as

$$\ddot{\mathbf{t}} = \mathbf{u} = -\mathbf{K}_P(\hat{\mathbf{t}} - \mathbf{t}^*) - \mathbf{K}_D(\dot{\hat{\mathbf{t}}} - \dot{\mathbf{t}}^*) + \ddot{\mathbf{t}}^* \tag{3.10}$$

so that (3.8) becomes

$$\dot{\mathbf{x}}_e = (\mathbf{A}_c - \mathbf{B}_c\mathbf{K}_{LQR})\mathbf{x}_e - \mathbf{B}_c\mathbf{K}_{LQR}\tilde{\mathbf{x}} \tag{3.11}$$

which, together with estimation error dynamics, yields the closed-loop system

$$\begin{aligned}\dot{\mathbf{x}}_e &= (\mathbf{A}_c - \mathbf{B}_c\mathbf{K}_{LQR})\mathbf{x}_e - \mathbf{B}_c\mathbf{K}_{LQR}\tilde{\mathbf{x}} \\ \dot{\tilde{\mathbf{x}}} &= (\mathbf{A}_c - \mathbf{L}\mathbf{C}_c)\tilde{\mathbf{x}}\end{aligned} \tag{3.12}$$

The stability of the "Moving Target" estimator was already proven in Subsection 2.4.4 and the control error equation can be seen as a linear stable system, given a specific $\mathbf{K}_{LQR}$, with input $\tilde{\mathbf{x}}$. Since $\tilde{\mathbf{x}}$ converges to zero, the equation will also converge to zero and so the overall system is stable.

In Figure 3.1, a Simulink scheme of the Target position control is portrayed. The `x desired` block contains the position and velocity desired $\mathbf{x}^* = \begin{bmatrix}\mathbf{t}^* & \dot{\mathbf{t}}^*\end{bmatrix}^T$ of the Target in the 3-dimensional space. Then, the desired state vector $\mathbf{x}^*$ enters the `Controller` block along with the current estimate of the state, $\hat{\mathbf{x}}_k$. In this block, the optimal matrix gain $\mathbf{K}_{LQR}$ is applied to the difference between $\mathbf{x}^*$ and $\hat{\mathbf{x}}$. The desired acceleration $\ddot{\mathbf{t}}^* = u^*$ is then added, as established by the control law in (3.6). The input $\mathbf{u}$ is then forwarded to the `System Dynamics` block along with the positions of the two Sensors, necessary to calculate the normalized relative vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ used in the output matrix $\mathbf{C}$. The `System Dynamics` block contains a continuous S-function which reproduce the system dynamics expressed in (3.1). Besides $\mathbf{y}$, the outputs of this block are the output matrix $\mathbf{C}$ which will be needed in the `Estimator` block and the true $\mathbf{x}$ which is not directly available, but in this simulation it is claimed for plotting reasons only. Both the estimated state vector $\hat{\mathbf{x}}$ and the optimal Sensor positions for estimating the Target's position are determined in the `Estimator` block. Here, a continuous S-function recreates the estimation process described in (3.4). Also in this block, the optimization process of the Sensors-Target geometry is carried out by a S-function where the positions of the Sensors in relation to the Target's estimated position are updated taking into account the maximization of the Fisher Information Matrix . Finally, the `Plots` block consists of graphs comparing the desired state vector $\mathbf{x}^*$, the true state vector $\mathbf{x}$ and the estimated state vector $\hat{\mathbf{x}}$ in all the 3 axes of the 3-dimensional space.

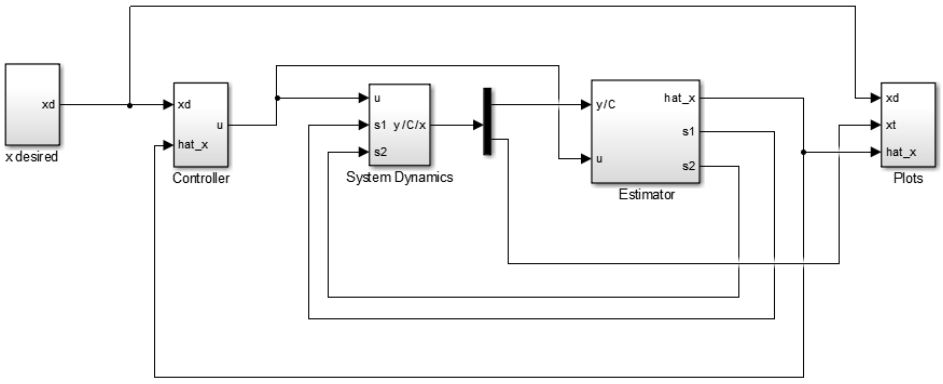


Figure 3.1: Simulink Model of the Target Position Control with the LQR controller

3.2 Results

In this subsection, two scenarios will be studied: firstly a situation where the Target is stationary and secondly a case where the Target is given a specific reference to follow, more particularly a sinusoidal trajectory similar to the one described in Subsection 2.5.3.

3.2.1 Control of Stationary Target

For the stationary Target case the Target state is assumed invariant, having a process noise $\mathbf{Q} = \mathbf{0}$. As in the other cases, the sensor noise is assumed to have a constant standard deviation $\sigma = 5^{o}$. To analyse the combination of the Target estimation performance with the controller performance, a Simulink model is created as depicted in Figure 3.1. Note that, in this case, the state vector $\mathbf{x}$ is composed by the three position components and also by the three velocity components. The following values are used to initialize the simulation

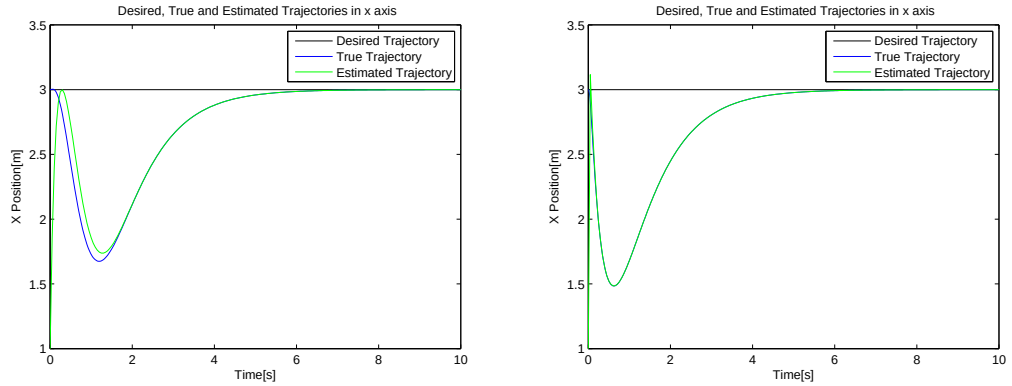
$$\mathbf{x}_0 = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \qquad \hat{\mathbf{x}}_0 = \begin{bmatrix} 2 & 1 & -1 & 0 & 0 & 0 \end{bmatrix}^T \tag{3.13}$$

Also, the Sensors' initial positions are defined according to the initial estimates of the Target's position. Since the problem of finding the optimal geometry in Subsection 2.5.1 is solved in 2-dimensional space and now the problem is transposed to 3D, it is necessary to impose limits to the z coordinates of Sensors 1 and 2 as well, by keeping them in the same horizontal plane as the Target, i.e., by setting Sensors 1 and 2 z position equal to the current z estimate of the Target. This way, the different angular Sensors-Target geometries studied in previous sections will remain with the same configurations and the optimization problem will be kept simple and uncomplicated by continuing to only optimize the y values of the Sensors' positions. Therefore, the Sensors are initially placed in each side of the bridge where they obtain GPS data, with z coordinates equal to the respective Target's estimated z coordinate and y coordinates at a certain distance behind the Target, depending on the criterion used. Note that it is important to guarantee that the Sensors are initially placed at a distance that ensures the real configuration (not the estimated) is between the limits of $140^{o}/40^{o}$ imposed when constructing the "Moving Target" estimator. In principle, this requirement is feasible since the quadrotors are supposed to initiate their task from the ground or a platform where one can easily identify if the initial configuration is within these limits. After that, the estimator will converge to the real position of the Target and the configuration will never exceed these limits when using the various criteria described before, apart from the Sensitivity criterion which leads to a 180^{o} geometric configuration and that was already dismissed in previous Sections due to its poor performance.

In terms of the LQR controller, after testing several pairs of LQR matrices $\mathbf{Q}_{LQR}$ and $\mathbf{R}_{LQR}$, the best values found for this simulation are $\mathbf{Q}_{LQR} = 10\mathbf{I}_6$ and $\mathbf{R}_{LQR} = \mathbf{I}_3$. These values can also be explained by bearing in mind that a larger value in matrix $\mathbf{Q}_{LQR}$ implies a greater penalization on the state variation. Since these gains are chosen for the stationary Target case, where small variation of the state is expected, it is understandable that matrix $\mathbf{Q}_{LQR}$ has higher values than matrix $\mathbf{R}_{LQR}$. As mentioned in Subsection 3.1, the optimal LQR matrix gain $\mathbf{K}_{LQR}$ is calculated using the MATLAB

function `lqr`($\mathbf{A}_c$,$\mathbf{B}_c$,$\mathbf{Q}_{LQR}$,$\mathbf{R}_{LQR}$).

The estimator's $\mathbf{L}$ matrix gains, $\mathbf{L}_1$ and $\mathbf{L}_2$, must respect the condition $\mathbf{L}_2 \prec 0.234\ {\mathbf{L}_1}^2$, as explained in Subsection 2.4.4. This means that there are infinite possible combinations of these two gains, since the gains can be as high as one wishes. Nevertheless, a first approach to calculate these gains is to fix an equality between $\mathbf{L}_1$ and $\mathbf{L}_2$. An acceptable first guess which gives a certain margin is $\mathbf{L}_2 = 0.2\ {\mathbf{L}_1}^2 \prec 0.234\ {\mathbf{L}_1}^2$. After having defined this equality, various values were assigned and tested for $\mathbf{L}_1$ and $\mathbf{L}_2$. The results of the trajectories obtained in the x axis for two simulations ran under the same conditions but with different $\mathbf{L}$ matrix gains are presented in Figures 3.2(a) and 3.2(b).



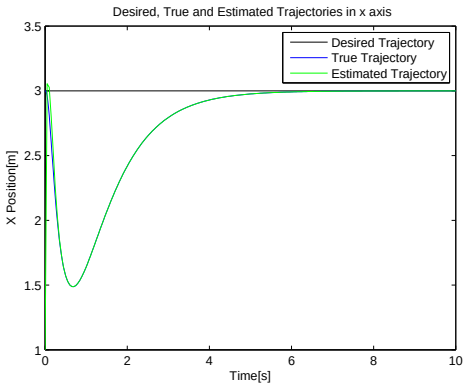
(a) $\mathbf{L}_1 = 10\mathbf{I}$ and $\mathbf{L}_2 = 20\mathbf{I}$ (b) $\mathbf{L}_1 = 100\mathbf{I}$ and $\mathbf{L}_2 = 2000\mathbf{I}$

Figure 3.2: Effects of changing $\mathbf{L}_1$ with $\mathbf{L}_2 = 0.2{\mathbf{L}_1}^2$ in the True and Estimated x Position

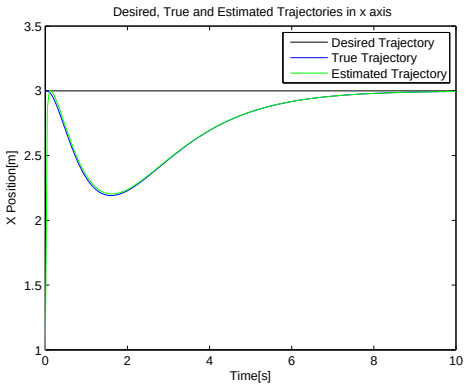
It is clear that smaller gains lead to a response with less overshoot. However, when comparing the blue lines with the green ones in each Figure, it can be concluded that the estimate takes less time to converge to the real x position when the gains are greater. Taking into account these two factors, it is necessary to find a balance between the fastness of the estimator response and the overshoot induced by this response on the system. For this reason, an intermediate pair of values $\mathbf{L}_1 = 50\mathbf{I}$ and $\mathbf{L}_2 = 250\mathbf{I}$ is chosen to proceed with the selection of the $\mathbf{L}$ matrix gains. In order to reduce the response's overshoot, the $\mathbf{L}_2$ gain will now be changed iteratively, with a fixed value for $\mathbf{L}_1 = 50\mathbf{I}$. The results of two of those experiments where $\mathbf{L}_2 = 0.2{\mathbf{L}_1}^2$ and $\mathbf{L}_2 = 0.015{\mathbf{L}_1}^2$ are portrayed in Figures 3.3(a) and 3.3(b), respectively.

Even though in Figure 3.3(b) the system takes more time to converge to the desired x position, the overshoot is clearly reduced to half the extent when comparing with Figure 3.3(a). Besides, it is also evident that reducing $\mathbf{L}_2$ does not affect significantly the convergence of the estimate to the true x position. For that reason, the final values chosen for the estimator's $\mathbf{L}$ matrix are $\mathbf{L}_1 = 50\mathbf{I}$ and $\mathbf{L}_2 = 37.5\mathbf{I}$.

After having defined the controller's $\mathbf{K}_{LQR}$ matrix gain and the estimator's $\mathbf{L}$ matrix, a simulation using a LQR controller, the "Moving Target" estimator to estimate the Target's position and the E-optimality criterion to determine the best Sensors-Target geometric configuration is performed. In Figures 3.4(a), 3.4(b) and 3.4(c) the x, y and z positions are plotted and in Figures 3.5(a), 3.5(b) and 3.5(c) the x, y and z velocities, respectively.

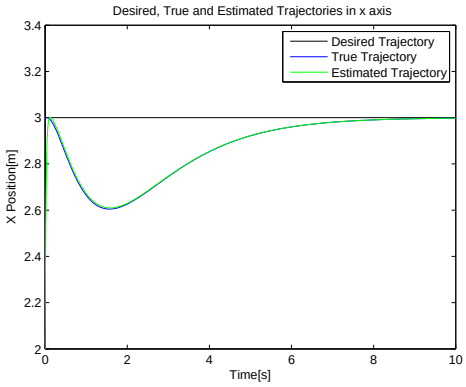


(a) $\mathbf{L}_1 = 50\mathbf{I}$ and $\mathbf{L}_2 = 500\mathbf{I}$

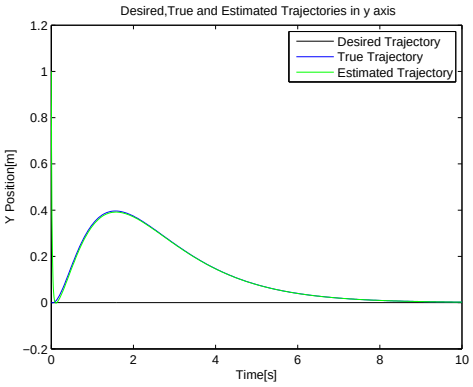


(b) $\mathbf{L}_1 = 50\mathbf{I}$ and $\mathbf{L}_2 = 37.5\mathbf{I}$

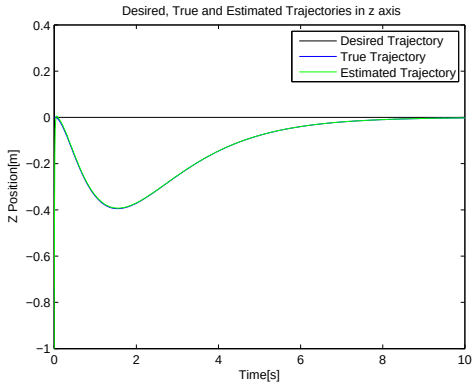
Figure 3.3: Effects of changing $\mathbf{L}_2$ with $\mathbf{L}_1 = 50\mathbf{I}$ in the True and Estimated x Position



(a) Desired, True and Estimated x Position



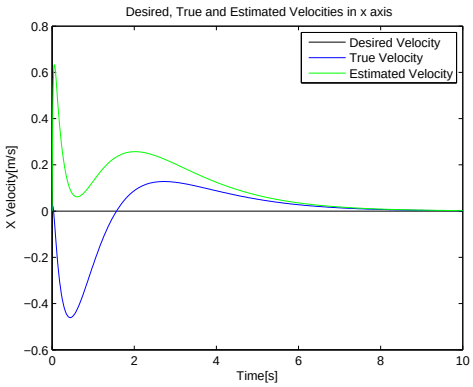
(b) Desired, True and Estimated y Position



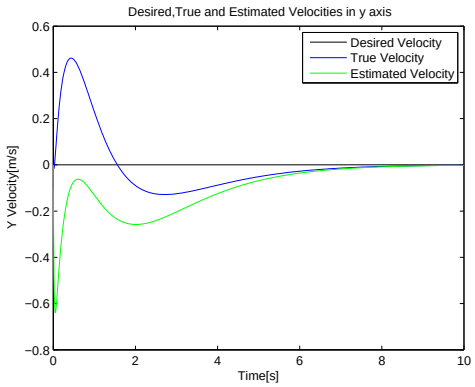
(c) Desired, True and Estimated z Position

Figure 3.4: Desired, True and Estimated Position of Stationary Target

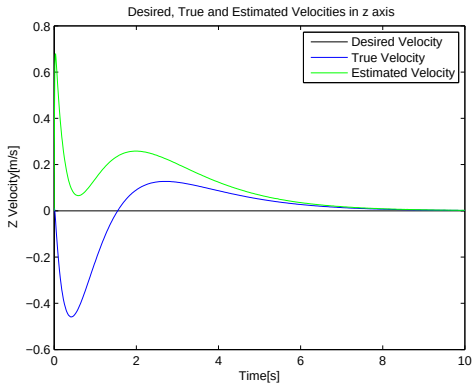
It is necessary to analyse each figure taking into account two crucial aspects, the estimation process and the controller response. First, as far as the "Moving Target" estimator is concerned, it is observed that the initial position estimates in all 3 axes, represented in green, are distant from the true position but quickly settle and converge to the blue lines representing the true position values. However, this fast response in the estimates of the position will have a direct effect in the estimated velocities and consequently in the true velocities. For instance, the initial estimate of the Target's position in the x axis



(a) Desired, True and Estimated x Velocity



(b) Desired, True and Estimated y Velocity



(c) Desired, True and Estimated z Velocity

Figure 3.5: Desired, True and Estimated Velocity of Stationary Target

is lower than its true value which means that the estimated velocity will be positive so that the estimated position quickly reaches the true position. This explains the initial positive x estimated velocity shown in Figure 3.5(a). Then, since the LQR controller compares the estimated with the desired state vector, an input to mitigate this difference will be applied to the real system, inducing a negative velocity and consequently decreasing the x true position. For that reason, the Target experiences a small overshoot which can be seen in Figure 3.4(a). Nevertheless, the Target true state vector promptly converges to the desired position and velocity, reaching it about 6 seconds later.

In Subsection 2.5.2, the different criteria are compared based on the Root-Mean-Squared Error (RMSE) between the estimate given by the "Moving Target" estimator, described in Subsection 2.4.4, and the true value of the Target's position. Now, this criteria comparison will be performed with a LQR controller on top. Therefore, for each criterion, the true position will be compared not only with the estimated value, but also with the desired one.

After averaging over 20 Monte-Carlo (MC) runs with different initial conditions, the performance of the 4 criteria are compared based on the Root-Mean-Squared Error (RMSE). In Table 3.1 are listed the RMSE for the 3-dimensional axes between the estimated and the true Target's position and in Table 3.2 the RMSE between the true and the desired Target's position. For all the simulations, the "Moving Target" estimator is used. The theoretical Cramer-Rao Lower Bounds in the first 2 seconds of the

estimation process are shown in Figure 3.6 for the 4 criteria.

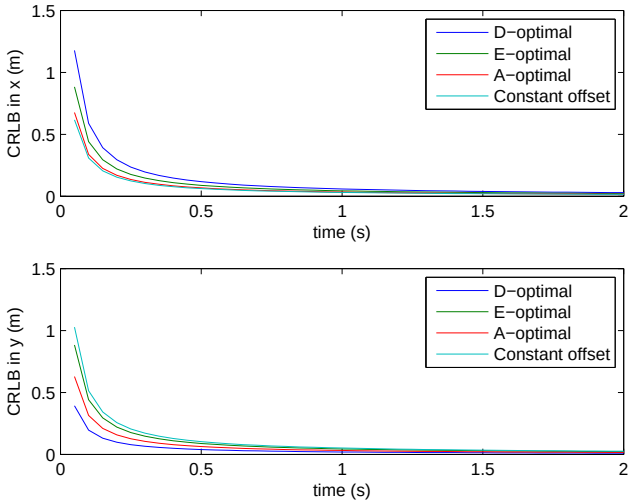


Figure 3.6: Cramer-Rao Lower Bounds (CRLB) in x (on top) and y (on the bottom) for different criteria

Table 3.1: Root-Mean-Squared Error (RMSE) between the Estimated and the True Target's Position in x, y and z for different criteria, with a Stationary Target

Criterion	RMSE in x (mm)	RMSE in y (mm)	RMSE in z (mm)
D-optimality	50.4461	25.3811	15.2849
E-optimality	30.0752	26.9098	14.2339
A-optimality	38.6422	26.0345	15.0239
Constant Offset	28.6261	30.8578	14.7179

Table 3.2: Root-Mean-Squared Error (RMSE) between the True and the Desired Target's Position in x, y and z for different criteria, with a Stationary Target

Criterion	RMSE in x (mm)	RMSE in y (mm)	RMSE in z (mm)
D-optimality	129.3776	105.1350	79.2338
E-optimality	126.6488	103.9396	77.9351
A-optimality	128.9813	105.2138	79.0412
Constant Offset	128.1783	114.0077	81.0234

Comparing the four criteria presented in Table 3.1, it is possible to see that the conclusions obtained in Subsection 2.5.2 when analysing Table 2.2 can be applied here too, since the "Constant Offset" and the E-optimality criteria lead to the lowest RMSE in the x axis, while the D- and A-optimality criteria produce better results in the y axis. The RMSE in z axis is also presented in Table 3.1 and, as expected, there is no significant difference between the various criteria. The E-optimality criterion involves finding a configuration that produces an uncertainty ellipse closer to a circle and indeed, amongst the three optimality criteria, it is the one which has less discrepancy between the RMSE in x and y axes. The D-optimality criterion might have the lowest RMSE in y, but it also has the highest one in the x axis, thus having an undesirable disparity between the two RMSE. The A-optimality criterion seems to produce RMSE in y similar to the E-optimality but has a higher RMSE in x. Still, it is considerably better than

the D-optimality criterion mainly in the x axis. Finally, even though the "Constant Offset" criterion has the lowest RMSE in x of all the criteria, it is evident that it also has the highest RMSE in y. Also, a close resemblance between the CRLB in Figure 3.6 and the RMSE can be seen, since the highest CRLBs correspond to the lowest RMSE and vice-versa, implying that the CRLB is a good indicator of the estimator performance.

In Table 3.2 it is possible to see and analyse the effect that each criterion has on the difference between the true and the desired Target's position. The first result that strikes is the discrepancy between the "Constant Offset" and the other three criteria as far as the RMSE in y is concerned. The E-optimality has the lowest RMSE in all three axes and consequently can be selected as the best criterion for this scenario. The A-optimality seems to be the second best followed by the D-optimality criterion. Despite the fact that the "Constant Offset" leads to a lower error in x than the A- and D-optimality criterion, it is undeniable that the RMSE in y is much higher and for that reason the "Constant Offset" criterion is considered the worst regarding the RMSE between the true and the desired Target's position for a scenario with a Stationary Target.

All things considered and taking into account the results obtained in both Tables 3.1 and 3.2, it is visible that, in general, the E-optimality criterion leads to better estimation and control results in a scenario where the Target is stationary. It can also be stated that the "Constant Offset" criterion, which is the only one that does not involve optimizing the Fisher Information Matrix, produces worse results mainly in the y axis when comparing with the optimality criteria.

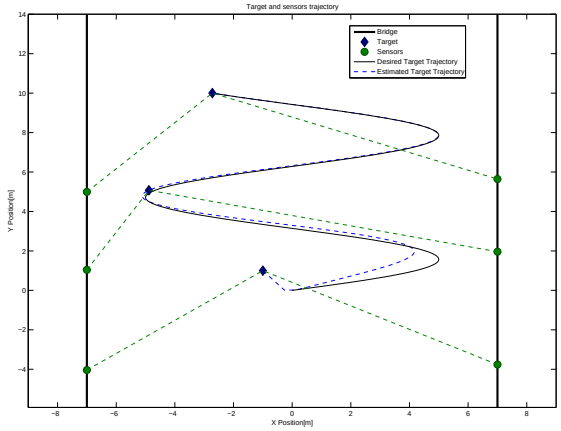
3.2.2 Control of Moving Target

The second scenario chosen for this section is the situation where the Target performs a sweeping action under the bridge's deck with a sinusoidal trajectory, in order to investigate the existence of any defect. The bridge's width is again considered to be 14m and the bridge's height equal to 10m. The origin of the coordinated system is considered to be in the middle of the bridge at an altitude slightly lower than the bridge deck to allow the Targets to perform their structural investigation tasks. The Target starts in the ground at a point in the middle of the bridge and the Sensors are initially placed in opposite sides of the bridge at the same estimated z value of the Target and at a y distance dependent on the criterion used. Since both Target and Sensors begin the simulation from a flat ground/platform situated under the bridge, their z coordinate is known and so the initial estimate of z can be set equal to the true Target's z value. Thus, the initial conditions are given by

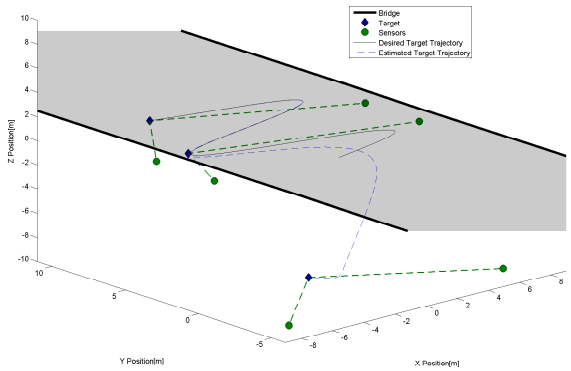
$$\mathbf{x}_0 = \begin{bmatrix} 0 & 0 & -10 & 0 & 0 & 0 \end{bmatrix}^T \qquad \hat{\mathbf{x}}_0 = \begin{bmatrix} -1 & 1 & -10 & 0 & 0 & 0 \end{bmatrix}^T \tag{3.14}$$

In Figure 3.7, a simulation of the scenario above explained is depicted in two perspectives: in Figure 3.7(a) a perspective from above and similar to the ones used in the 2D problems and in Figure 3.7(b) a 3D plot where the trajectory in the z axis is more elucidative. For this simulation, the criterion used in the maximization of the FIM is the A-optimality criterion.

Figure 3.8 shows the desired, true and estimated position of the Target following a sinusoidal trajectory



(a) Sensors-Target Geometry seen from above

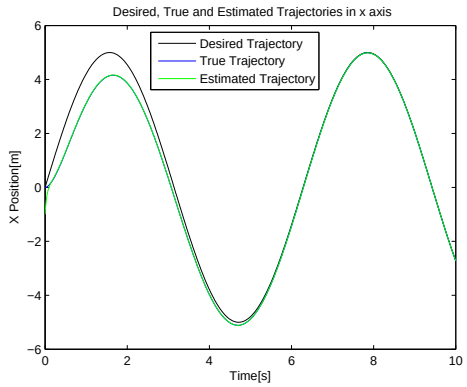


(b) 3D representation of Sensors-Target Geometry

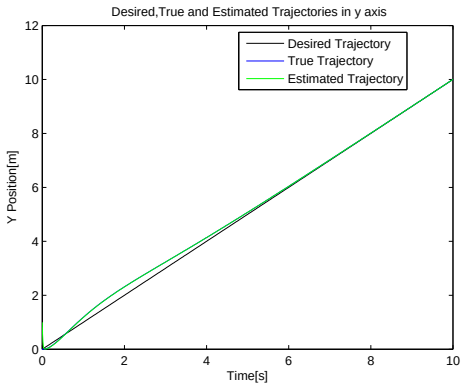
Figure 3.7: Target following a Sinusoidal Trajectory

and Figure 3.9 presents the respective velocity. The difference between the estimated and true position values mitigates very quickly and for that reason the green line representing the estimated trajectory is overlapping the true trajectory blue line in all the plots. As shown in Figures 3.9(a) and 3.9(b), the Target starts with true and estimated null velocities in the x and y axes which converge to the non-null desired velocities about 4 seconds after the simulation started. In the z axis, even though the desired velocity is null, both the estimated and true velocities need to be positive at the beginning in order to correct the Target's z position. As the position in the z axis approximates to the desired value, the correspondent velocity also decreases to the desired null value. The initial misalignment between the Target's true and desired position shown in Figures 3.8(a) and 3.8(b) is easily explained not only by the difference between the initial estimate and the true position, but also due to the fact that the Target starts the simulation from a stationary position, requiring a short period of time to attain the desired velocity. Still, the Target's x and y true positions converge to their desired values, taking approximately 3 and 4 seconds, respectively, to reach them. After analysing Figure 3.8(c), it can be clearly seen the convergence of the true z position to the desired value, 10 meters above from the initial position. The

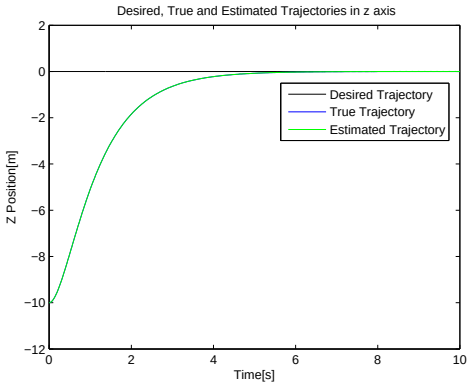
Target reaches its desired z position around 4 seconds after the simulation started. Another aspect that is important to mention is that the z response has no overshoot, which is crucial in order to avoid undesired situations such as the Target colliding with the bridge's deck from below.



(a) Desired, True and Estimated x Position



(b) Desired, True and Estimated y Position



(c) Desired, True and Estimated z Position

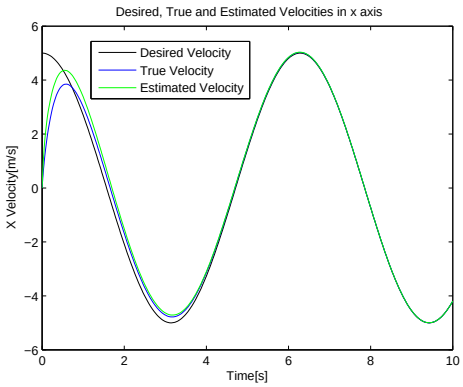
Figure 3.8: Desired, True and Estimated Position of Target following a Sinusoidal Trajectory

After averaging over 20 Monte-Carlo (MC) runs with different initial conditions, the performance of the 4 criteria are compared based on the Root-Mean-Squared Error (RMSE). In Table 3.3 are listed the RMSE for the 3-dimensional axes between the estimated and the true Target's position and in Table 3.4 the RMSE between the true and the desired Target's position. The theoretical Cramer-Rao Lower Bounds in the first 2 seconds of the estimation process are shown in Figure 3.10 for the 4 criteria.

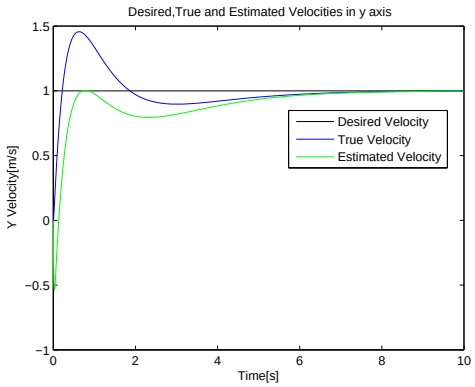
Table 3.3: Root-Mean-Squared Error (RMSE) between the Estimated and the True Target's Position in x, y and z for different criteria, when performing a sinusoidal trajectory

Criterion	RMSE in x (mm)	RMSE in y (mm)	RMSE in z (mm)
D-optimality	48.6215	26.1653	32.2716
E-optimality	34.7550	31.6491	33.3210
A-optimality	46.5583	27.5222	33.1201
Constant Offset	32.0687	33.8432	32.3295

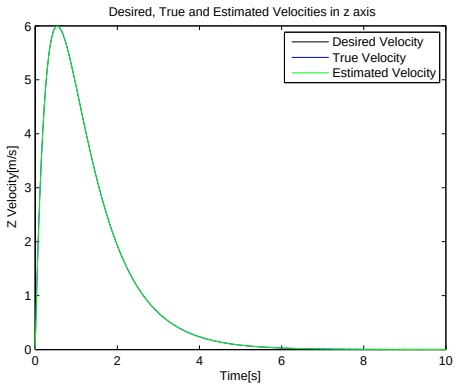
Comparing the four criteria presented in Table 3.3, the same conclusions made in Subsection 3.2.1 for the Stationary Target scenario can be applied for this scenario where the Target is performing a sweep action with a sinusoidal trajectory. The D-optimality criterion continues to present the highest RMSE



(a) Desired, True and Estimated x Velocity



(b) Desired, True and Estimated y Velocity



(c) Desired, True and Estimated z Velocity

Figure 3.9: Desired, True and Estimated Velocity of Target following a Sinusoidal Trajectory

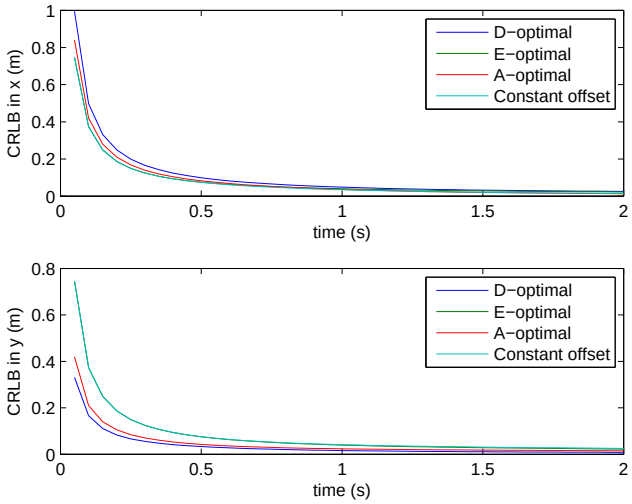


Figure 3.10: Cramer-Rao Lower Bounds (CRLB) in x (on top) and y (on the bottom) for different criteria

in x and the lowest in y. The E-optimality tries to have similar uncertainties in both axes and this is endorsed by the values obtained in Table 3.3. The A-optimality leads to the second worst RMSE in x but also the second best in y. Finally, the "Constant Offset" criterion can be seen as the opposite of the D-optimality, since it has the lowest RMSE in x but, on the other hand, it produces the highest RMSE

Table 3.4: Root-Mean-Squared Error (RMSE) between the True and the Desired Target's Position in x, y and z for different criteria, when performing a sinusoidal trajectory

Criterion	RMSE in x (mm)	RMSE in y (mm)	RMSE in z (mm)
D-optimality	213.4988	118.1488	1839.7958
E-optimality	209.2227	117.7496	1777.3227
A-optimality	206.5138	116.4500	1785.2439
Constant Offset	212.0066	121.5137	1791.9211

in y of all the considered criteria. As expected, the RMSE in z axis does not have a significant difference between the various criteria. Moreover, the theoretical CRLBs in Figure 3.10 are again in agreement with the estimation results obtained, since the highest CRLBs correspond to the lowest RMSE and vice-versa. Note that in Figure 3.10 the green and light blue lines representing the E-optimality and "Constant Offset" criteria, respectively, are very close to each other and therefore are overlapped in the plot, also explaining the proximity between the correspondent RMSE in both x and y axes.

In Table 3.4 it is possible to see and analyse the effect that each criterion has on the difference between the true and the desired Target's position when performing a sinusoidal trajectory. In this case, the criterion that seems to produce better results is the A-optimality since it leads to the lowest RMSE in x and y axes. The E-optimality can be considered the second best followed by the D-optimality criterion. The "Constant Offset" is again the worst mainly due to the considerably higher error in y. The fact that the RMSE in z is much higher than in the other axes is easily explained by the fact that the initial true z position of the Target is 10 meters below the desired one.

To conclude, the results shown in both Tables 3.3 and 3.4 indicate that, although there is no clear evidence of a criterion that has undoubtedly the lowest RMSE between the estimated and true Target's position, the same does not happen when comparing the true with the desired position. In this case, the A-optimality leads to better results whereas the "Constant Offset", which is the only non-optimal of all the considered criteria, produce worse results mainly in the y axis.

Chapter 4

Network Localization

In this Chapter, the concepts of network, point formations and rigidity in directed graphs are introduced. Then, a possible construction method for directed rigid formations is described and the convergence and uniqueness of the stationary solution constructed with that method is demonstrated. It ends with the implementation of the theory previously developed in order to construct a solid network formation.

4.1 Network and Point Formation Definitions

In this section, the problem of localizing a network of Targets and Sensors is handled. A network $\mathbf{N}$ of n nodes needs to be defined in a 2D space. From those n nodes, s represent "anchor" nodes which have GPS capabilities and therefore know their exact position, i.e, the Sensors. The remaining $n-s=t$ nodes are the so-called ordinary nodes which represent the Targets that do not know their exact location and can only infer their position from bearings measurements. The aim is to determine the unknown positions of the t Targets given the known locations of the s Sensors and the required bearing measurements. A network $\mathbf{N}$ can be described by an undirected graph $\mathbf{G_N} = (V, E)$ where V is the set of nodes or vertices and E is the set of unordered pair of vertices that have a link between them, usually called edges.

To study the solvability of this network localization problem, it is necessary to reformulate and use the concept of point formation. A point formation has a similar graph associated with the corresponding network,with the only difference that in a point formation there are extra edges that link all the anchor nodes. It is obvious that the anchor nodes do not need any incoming link to localize themselves since they have full knowledge of their position. Nevertheless, these links are necessary to form the grounded graph $\hat{\mathbf{G}}_\mathbf{N}$ of the network $\mathbf{N}$ which is crucial to prove the solvability of the network localization problem using bearings measurements.

4.2 Rigidity of Directed Graphs

A point formation is called rigid if the distance between each pair of nodes remains constant over time. Rigidity is a property of the graph that can be used to prove the uniqueness of network localization

solutions using bearings-only measurements. For other situations like, for instance, networks using only distance information, stronger concepts of rigidity are necessary to guarantee a unique network solution, as discussed in [1].

From Laman's Theorem [13], an undirected graph $\mathbf{G_N} = (V, E)$ with n vertices is a rigid graph in $\mathbb{R}^2$ if and only if there is a subset $E' \subseteq E$ satisfying the following two conditions:

1. $|E'| = 2|V| - 3$;
2. for all $E'' \subseteq E'$, $E'' \neq \emptyset$, $|E''| \leq 2|V(E'')| - 3$ where $|V(E'')|$ is the number of vertices that are end-vertices of the edges in E''.

A network $\mathbf{N}$ can be described by a directed graph $\mathbf{G_N} = (V, D)$ where V is the set of nodes or vertices and D is the set of ordered pair of vertices called directed edges. A directed graph is a graph where the edges have a direction associated with them. A directed edge is written with an ordered pair of vertices (i, j) represented with an arrow from i to j. The number of edges directed into a certain vertex i in a directed graph $\mathbf{G}$ is called the *in-degree* of the vertex and is denoted by $d^-_{\mathbf{G}}(i)$, whereas the number of edges directed out from vertex i is called the *out-degree* and is denoted by $d^+_{\mathbf{G}}(i)$.

As it was already suggested, the vertices or nodes of the graph represent the Sensors and Targets in the network. A directed edge (i, j) represents a bearing measurement, i.e., the azimuth angle and the elevation angle between node i and node j. The direction of the edge determines which node will use the bearing information for localizing itself. For instance, in a directed edge (i, j) represented with an arrow from i to j, only node j will use the bearing information between i and j in the localization process.

When an undirected graph is transformed into a directed one, rigidity is not necessarily maintained as a poor selection of the directions may lead, for instance, to a in-degree of 1 in a certain node and a out-degree of 3 in another one. Since a minimum in-degree of 2 is necessary to localize a node, in this case the formation would not be rigid which confirms that undirected rigidity is a necessary but not sufficient condition for directed rigidity.

According to Eren [5], a directed graph $\mathbf{G_N} = (V, D)$ is rigid if and only if there is a subset $D' \subseteq D$ satisfying the following conditions:

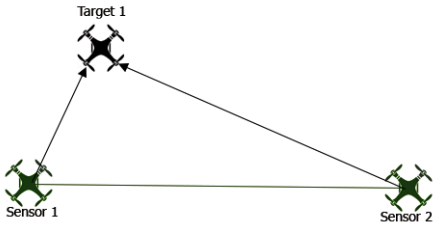
1. $|D'| = 2|V| - 3$;
2. for all $D'' \subseteq D'$, $D'' \neq \emptyset$, $|D''| \leq 2|V(D'')| - 3$ where $|V(D'')|$ is the number of vertices that are end-vertices of the edges in D'';
3. $\mathbf{G}' = (V, D')$ is 2-directed, meaning that, for all $i \in V$, $d^-_{\mathbf{G}'}(i) \leq 2$.

4.3 Construction Method For Directed Rigid Formations

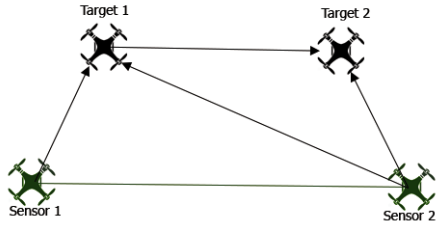
One construction method that preserves the undirected rigidity of point formations is the so-called Henneberg construction method which is described by Eren et al. [6]. A similar approach can be made when constructing directed graphs, bearing in mind that the direction of the edges is now critical and must be taken into account. Even though the existence of one anchor node rules out translation, scaling

is still possible and therefore, a minimum number of 2 anchor nodes is mandatory in order to ensure the rigidity of the network formation. Thus, this method starts with a single link between the 2 anchors at the first step to initialize the construction of the grounded graph. In Figure 4.1(a) it is possible to see an illustration of this initial phase, where the two Sensors 1 and 2 represent the anchor nodes and have full knowledge of their position. Then, there are mainly two possible operations that will be explored in the extension process of the rigid directed graph. First, there is the directed vertex addition (DVA), in which a new vertex is added with two edges directed from two other vertices in the rigid graph to the new vertex. The DVA step is illustrated in Figure 4.1(b) where the vertex representing Target 1 is added with two directed edges coming from Sensor 1 and Sensor 2, and also in Figure 4.1(c), where a new vertex representing Target 2 is added with two directed edges coming from two other vertices from the graph, which in this case are Sensor 1 and Target 1. The second operation that preserves the rigidity of the directed graph is the directed edge splitting (DES) in which an existing directed edge (j, k) is removed and a new vertex i is added with two directed edges (j, i), (i, k) and a third directed edge linking another vertex from the graph (other than j and k) to the new vertex i. The DES operation is portrayed in Figure 4.1(d) where the existing directed edge connecting Sensor 2 with Target 1 is removed and a new vertex representing Target 3 is added with two directed edges from Sensor 2 to Target 3 and from Target 3 to Target 1 and a third directed edge linking another vertex from the graph (other than Target 1 and Sensor 2) to Target 3. In this case, the vertex chosen was the one representing Target 2, but another possible choice would be Sensor 1.

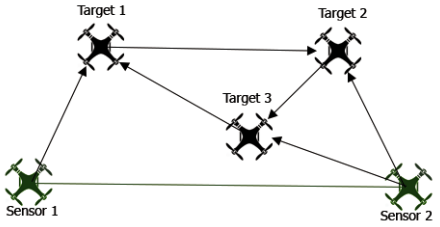
(a) Initialisation of the grounded graph with two anchors (Sensors 1 and 2)



(b) Directed vertex addition step (Target 1)



(c) Directed vertex addition step (Target 2)



(d) Directed edge splitting step (Target 3)

Figure 4.1: Steps in the Construction Method For Directed Rigid Formations

4.4 Convergence and Uniqueness of the Stationary Solution

After having designed a rigid formation with the construction method presented above, the next step in the network localization problem is to prove the convergence to a unique stationary solution using a well-designed estimator. For this section, the estimator used will be the "Stationary Target" estimator described in Subsection 2.4.3.

A simple approach to study the uniqueness of the stationary solution obtained can be introduced by first considering the simplest case where one Target is localized using the bearing measurements from two Sensors, as depicted in Figure 4.1(b). The estimation equation would then be

$$\hat{\mathbf{t}}_1^+ = \hat{\mathbf{t}}_1 + \alpha \left(S(\mathbf{u}_{s_1})^2(\hat{\mathbf{t}}_1 - \mathbf{s}_1) + S(\mathbf{u}_{s_2})^2(\hat{\mathbf{t}}_1 - \mathbf{s}_2) \right) \tag{4.1}$$

where, for the sake of simplicity, the notation $\hat{\mathbf{t}}_1 \equiv \hat{\mathbf{t}}_1(t)$ and $\hat{\mathbf{t}}_1^+ \equiv \hat{\mathbf{t}}_1(t+1)$ was adopted. When the estimation converges, $\hat{\mathbf{t}}_1^+ = \hat{\mathbf{t}}_1$, and the equation becomes

$$S(\mathbf{u}_{s_1})^2(\hat{\mathbf{t}}_1 - \mathbf{s}_1) + S(\mathbf{u}_{s_2})^2(\hat{\mathbf{t}}_1 - \mathbf{s}_2) = 0 \tag{4.2}$$

After rearranging (4.2)

$$\hat{\mathbf{t}}_1(S(\mathbf{u}_{s_1})^2 + S(\mathbf{u}_{s_2})^2) = S(\mathbf{u}_{s_1})^2\mathbf{s}_1 + S(\mathbf{u}_{s_2})^2\mathbf{s}_2 \tag{4.3}$$

Since $(S(\mathbf{u}_{s_1})^2 + S(\mathbf{u}_{s_2})^2)$ is, as shown in Subsection 2.4.3, a negative definite matrix unless $\mathbf{u}_{s_1}$ and $\mathbf{u}_{s_2}$ are collinear, then it is also invertible, meaning that the solution to (4.2) is unique and satisfies

$$\begin{cases} S(\mathbf{u}_{s_1})^2(\hat{\mathbf{t}}_1 - \mathbf{s}_1) = 0 \\ S(\mathbf{u}_{s_2})^2(\hat{\mathbf{t}}_1 - \mathbf{s}_2) = 0 \end{cases} \tag{4.4}$$

Since, for instance

$$S(\mathbf{u}_{s_1})^2(\hat{\mathbf{t}}_1 - \mathbf{s}_1) = \frac{\mathbf{t}_1 - \mathbf{s}_1}{\|\mathbf{t}_1 - \mathbf{s}_1\|} \times \left(\frac{\mathbf{t}_1 - \mathbf{s}_1}{\|\mathbf{t}_1 - \mathbf{s}_1\|} \times (\hat{\mathbf{t}}_1 - \mathbf{s}_1) \right) = 0 \tag{4.5}$$

then $\mathbf{u}_{s_1}$ and $(\hat{\mathbf{t}}_1 - \mathbf{s}_1)$ must be collinear vectors and thus the only solution corresponds exactly to $\hat{\mathbf{t}}_1 = \mathbf{t}_1$.

The convergence to a unique stationary solution can be proved for each construction step (DVA and DES) described in Subsection 4.3. By proving it for each step, it can then be concluded that this construction method creates a network formation where a convergent estimation always leads to a unique solution corresponding to the real one.

4.4.1 Directed Vertex Addition Step

Considering first the DVA step illustrated in Figure 4.1(c), the estimation equations of the two Targets are

$$\begin{cases} \hat{\mathbf{t}}_1^+ = \hat{\mathbf{t}}_1 + \alpha \left(S(\mathbf{u}_{s_1 1})^2 (\hat{\mathbf{t}}_1 - \mathbf{s}_1) + S(\mathbf{u}_{s_2 1})^2 (\hat{\mathbf{t}}_1 - \mathbf{s}_2) \right) \\ \hat{\mathbf{t}}_2^+ = \hat{\mathbf{t}}_2 + \alpha \left(S(\mathbf{u}_{12})^2 (\hat{\mathbf{t}}_2 - \hat{\mathbf{t}}_1) + S(\mathbf{u}_{s_2 2})^2 (\hat{\mathbf{t}}_2 - \mathbf{s}_2) \right) \end{cases} \tag{4.6}$$

which can be transformed into matrix form like

$$\hat{\mathbf{T}}_2^+ = \hat{\mathbf{T}}_2 + \alpha(\mathbf{F} \cdot \hat{\mathbf{T}}_2 - \mathbf{G} \cdot \mathbf{S}_2) \tag{4.7}$$

where $\hat{\mathbf{T}}_2 = \begin{bmatrix} \hat{\mathbf{t}}_1 & \hat{\mathbf{t}}_2 \end{bmatrix}^T$, $\mathbf{S}_2 = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 \end{bmatrix}^T$ and

$$\mathbf{F} = \begin{bmatrix} S(\mathbf{u}_{s_1 1})^2 + S(\mathbf{u}_{s_2 1})^2 & \mathbf{0} \\ -S(\mathbf{u}_{12})^2 & S(\mathbf{u}_{s_2 2})^2 + S(\mathbf{u}_{12})^2 \end{bmatrix}, \qquad \mathbf{G} = \begin{bmatrix} S(\mathbf{u}_{s_1 1})^2 & S(\mathbf{u}_{s_2 1})^2 \\ \mathbf{0} & S(\mathbf{u}_{s_2 2})^2 \end{bmatrix} \tag{4.8}$$

Since the true Target positions also satisfy (4.7), the error system with state given by $\tilde{\mathbf{T}}_2 = \hat{\mathbf{T}}_2 - \mathbf{T}_2$ can be written as

$$\tilde{\mathbf{T}}_2^+ = (\mathbf{I} + \alpha \mathbf{F}) \tilde{\mathbf{T}}_2 \tag{4.9}$$

meaning that $\tilde{\mathbf{T}}_2$ will converge to zero if $(\mathbf{I} + \alpha \mathbf{F})^T (\mathbf{I} + \alpha \mathbf{F}) \prec \mathbf{I}$ or $\rho(\mathbf{I} + \alpha \mathbf{F}) < 1$, being $\rho(\cdot)$ the spectral radius of a matrix.

It can be proven that the eigenvalues of $\mathbf{F}$ correspond to the eigenvalues of its diagonal block matrices if the following verifies

$$\begin{bmatrix} S(\mathbf{u}_{s_1 1})^2 + S(\mathbf{u}_{s_2 1})^2 & \mathbf{0} \\ -S(\mathbf{u}_{12})^2 & S(\mathbf{u}_{s_2 2})^2 + S(\mathbf{u}_{12})^2 \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{V}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{V}_2 \end{bmatrix} \mathbf{D}_2 \tag{4.10}$$

where $\mathbf{D}_2$ is a diagonal matrix containing the eigenvalues of $S(\mathbf{u}_{s_2 2})^2 + S(\mathbf{u}_{12})^2$ and $\mathbf{V}_2$ is a matrix whose columns are the corresponding eigenvectors and also if

$$\begin{bmatrix} S(\mathbf{u}_{s_1 1})^2 + S(\mathbf{u}_{s_2 1})^2 & \mathbf{0} \\ -S(\mathbf{u}_{12})^2 & S(\mathbf{u}_{s_2 2})^2 + S(\mathbf{u}_{12})^2 \end{bmatrix} \begin{bmatrix} \mathbf{V}_1 \\ * \end{bmatrix} = \begin{bmatrix} \mathbf{V}_1 \\ * \end{bmatrix} \mathbf{D}_1 \tag{4.11}$$

where $\mathbf{D}_1$ is a diagonal matrix containing the eigenvalues of $S(\mathbf{u}_{s_1 1})^2 + S(\mathbf{u}_{s_2 1})^2$ and $\mathbf{V}_1$ is a matrix whose columns are the corresponding eigenvectors. The $*$ represents a block matrix which is not relevant to demonstrate that the eigenvalues of $\mathbf{F}$ are equal to the ones of the submatrices in the main diagonal of $\mathbf{F}$. It can be easily seen that (4.10) and (4.11) are indeed verified and so it can be concluded that the eigenvalues of matrix $\mathbf{F}$ are equal to the eigenvalues of all of its main diagonal submatrices.

This result is very useful not only to prove that $\mathbf{F}$ is invertible, but also to prove that $\rho(\mathbf{I} + \alpha \mathbf{F}) < 1$

$$\rho(\mathbf{I} + \alpha \mathbf{F}) = \rho \left(\begin{bmatrix} \mathbf{I} + \alpha(S(\mathbf{u}_{s_1 1})^2 + S(\mathbf{u}_{s_2 1})^2) & \mathbf{0} \\ -S(\mathbf{u}_{12})^2 & \mathbf{I} + \alpha(S(\mathbf{u}_{s_2 2})^2 + S(\mathbf{u}_{12})^2) \end{bmatrix} \right) < 1 \tag{4.12}$$

Similarly to $\mathbf{F}$, the eigenvalues of $\mathbf{I}+\alpha\mathbf{F}$ also correspond to the eigenvalues of the submatrices in its main diagonal. Thus, $\rho(\mathbf{I}+\alpha\mathbf{F})<1$ can be decomposed into two conditions

$$\begin{cases}\rho(\mathbf{I}+\alpha(S(\mathbf{u}_{s_11})^2+S(\mathbf{u}_{s_21})^2))<1\\ \rho(\mathbf{I}+\alpha(S(\mathbf{u}_{s_22})^2+S(\mathbf{u}_{12})^2))<1\end{cases} \tag{4.13}$$

It can be noted that the terms inside $\rho(\cdot)$ in (4.13) have a similar structure to (2.56). Therefore, the deductions made in Subsection 2.4.3 can also be applied here, leading to the same condition where $\alpha<1$ in order to have a convergent estimation process.

Since the estimator converges, $\hat{\mathbf{t}}_i^+=\hat{\mathbf{t}}_i$ for $i=1,2$, and (4.6) becomes

$$\begin{cases}\big(S(\mathbf{u}_{s_11})^2(\hat{\mathbf{t}}_1-\mathbf{s}_1)+S(\mathbf{u}_{s_21})^2(\hat{\mathbf{t}}_1-\mathbf{s}_2)\big)=0\\ \big(S(\mathbf{u}_{12})^2(\hat{\mathbf{t}}_2-\hat{\mathbf{t}}_1)+S(\mathbf{u}_{s_22})^2(\hat{\mathbf{t}}_2-\mathbf{s}_2)\big)=0\end{cases} \tag{4.14}$$

or in matrix form

$$\mathbf{F}\cdot\hat{\mathbf{T}}_2=\mathbf{G}\cdot\mathbf{S}_2 \tag{4.15}$$

Analysing matrix $\mathbf{F}$ in (4.8), since $(S(\mathbf{u}_i)^2+S(\mathbf{u}_j)^2)$ is, as shown in Subsection 2.4.3, a negative definite matrix unless $\mathbf{u}_i$ and $\mathbf{u}_j$ are collinear, then the eigenvalues of $\mathbf{F}$ are all negative, thus being an invertible matrix. This means that (4.15) has a unique solution where the following holds

$$\begin{cases}S(\mathbf{u}_{s_11})^2(\hat{\mathbf{t}}_1-\mathbf{s}_1)=0\\ S(\mathbf{u}_{s_21})^2(\hat{\mathbf{t}}_1-\mathbf{s}_2)=0\\ S(\mathbf{u}_{12})^2(\hat{\mathbf{t}}_2-\hat{\mathbf{t}}_1)=0\\ S(\mathbf{u}_{s_22})^2(\hat{\mathbf{t}}_2-\mathbf{s}_2)=0\end{cases} \tag{4.16}$$

becoming clear that $\hat{\mathbf{T}}_2=\mathbf{T}_2$ is the only solution.

Expanding this to N targets, the addition of another Target using the DVA step will lead to

$$\hat{\mathbf{T}}_{N+1}^+=\hat{\mathbf{T}}_{N+1}+\alpha(\mathbf{F}_{N+1}\cdot\hat{\mathbf{T}}_{N+1}-\mathbf{G}_{N+1}\cdot\mathbf{S}_2) \tag{4.17}$$

with

$$\mathbf{F}_{N+1}=\begin{bmatrix}\mathbf{F}_N & \mathbf{0}\\ \mathbf{C}_{N+1} & S(\mathbf{u}_{iN+1})^2+S(\mathbf{u}_{jN+1})^2\end{bmatrix} \tag{4.18}$$

where $\mathbf{F}_N$ has negative eigenvalues and $\mathbf{C}_{N+1}$ and $S(\mathbf{u}_{iN+1})^2+S(\mathbf{u}_{jN+1})^2$ are blocks representing the new connections due to the addition of a new Target.

Since the true Target positions also satisfy (4.17), the error system with state given by $\tilde{\mathbf{T}}_{N+1}=\hat{\mathbf{T}}_{N+1}-\mathbf{T}_{N+1}$ can be written as

$$\tilde{\mathbf{T}}_{N+1}^+=(\mathbf{I}+\alpha\mathbf{F}_{N+1})\tilde{\mathbf{T}}_{N+1} \tag{4.19}$$

meaning that $\tilde{\mathbf{T}}_{N+1}$ will converge to zero if $(\mathbf{I}+\alpha\mathbf{F}_{N+1})^T(\mathbf{I}+\alpha\mathbf{F}_{N+1})\prec\mathbf{I}$ or $\rho(\mathbf{I}+\alpha\mathbf{F}_{N+1})<1$.

Similarly to the previous case, it can be verified that

$$\mathbf{F}_{N+1}\begin{bmatrix}\mathbf{0}\\ \mathbf{V}_{N+1}\end{bmatrix} = \begin{bmatrix}\mathbf{0}\\ \mathbf{V}_{N+1}\end{bmatrix}\mathbf{D}_{N+1} \tag{4.20}$$

where $\mathbf{D}_{N+1}$ is a diagonal matrix containing the eigenvalues of $S(\mathbf{u}_{iN+1})^2 + S(\mathbf{u}_{jN+1})^2$ and $\mathbf{V}_{N+1}$ is a matrix whose columns are the corresponding eigenvectors and also that

$$\mathbf{F}_{N+1}\begin{bmatrix}\mathbf{V}_{N}\\ *\end{bmatrix} = \begin{bmatrix}\mathbf{V}_{N}\\ *\end{bmatrix}\mathbf{D}_{N} \tag{4.21}$$

where $\mathbf{D}_N$ is a diagonal matrix containing the eigenvalues of $\mathbf{F}_N$ and $\mathbf{V}_N$ is a matrix whose columns are the corresponding eigenvectors. The $*$ represents a block matrix which is not relevant to demonstrate that the eigenvalues of $\mathbf{F}_{N+1}$ are equal to the ones of the submatrices in its main diagonal.

This result is very useful not only to prove that $\mathbf{F}_{N+1}$ is invertible, but also to prove that $\rho(\mathbf{I} + \alpha\mathbf{F}_{N+1}) < 1$

$$\rho(\mathbf{I} + \alpha\mathbf{F}_{N+1}) = \rho\left(\begin{bmatrix}\mathbf{I} + \alpha\mathbf{F}_N & \mathbf{0}\\ \mathbf{C}_{N+1} & \mathbf{I} + \alpha(S(\mathbf{u}_{iN+1})^2 + S(\mathbf{u}_{jN+1})^2)\end{bmatrix}\right) < 1 \tag{4.22}$$

Similarly to $\mathbf{F}_{N+1}$, the eigenvalues of $\mathbf{I}+\alpha\mathbf{F}_{N+1}$ also correspond to the eigenvalues of the submatrices in its main diagonal. Thus, $\rho(\mathbf{I} + \alpha\mathbf{F}_{N+1}) < 1$ can be decomposed into 2 conditions

$$\begin{cases}\rho(\mathbf{I} + \alpha\mathbf{F}_N) < 1\\ \rho(\mathbf{I} + \alpha(S(\mathbf{u}_{iN+1})^2 + S(\mathbf{u}_{jN+1})^2)) < 1\end{cases} \tag{4.23}$$

Since $\mathbf{F}_N$ has only negative eigenvalues, the first condition is met if $\alpha < 1$. Moreover, it can be noted that the term inside $\rho(\cdot)$ in the second condition in (4.23) has a similar structure to (2.56). Therefore, the deductions made in Subsection 2.4.3 can also be applied here, leading to the same condition where $\alpha < 1$ in order to have a convergent estimation process.

Since the estimator converges, $\hat{\mathbf{t}}_i^+ = \hat{\mathbf{t}}_i$ for $i = 1, 2, ..., N+1$, and the following is obtained

$$\mathbf{F}_{N+1} \cdot \hat{\mathbf{T}}_{N+1} = \mathbf{G}_{N+1} \cdot \mathbf{S}_2 \tag{4.24}$$

Analysing matrix $\mathbf{F}_{N+1}$ in (4.18), since $(S(\mathbf{u}_i)^2 + S(\mathbf{u}_j)^2)$ is, as shown in Subsection 2.4.3, a negative definite matrix unless $\mathbf{u}_i$ and $\mathbf{u}_j$ are collinear, then the eigenvalues of $\mathbf{F}_{N+1}$ are all negative, thus being an invertible matrix. This means that (4.24) has a unique solution corresponding to $\hat{\mathbf{T}}_{N+1} = \mathbf{T}_{N+1}$.

4.4.2 Directed Edge Splitting Step

Now considering the DES step illustrated in Figure 4.1(d), the estimation equations of the three Targets are

$$\begin{cases} \hat{\mathbf{t}}_1^+ = \hat{\mathbf{t}}_1 + \alpha \left(S(\mathbf{u}_{s_1 1})^2 (\hat{\mathbf{t}}_1 - \mathbf{s}_1) + S(\mathbf{u}_{31})^2 (\hat{\mathbf{t}}_1 - \hat{\mathbf{t}}_3) \right) \\ \hat{\mathbf{t}}_2^+ = \hat{\mathbf{t}}_2 + \alpha \left(S(\mathbf{u}_{12})^2 (\hat{\mathbf{t}}_2 - \hat{\mathbf{t}}_1) + S(\mathbf{u}_{s_2 2})^2 (\hat{\mathbf{t}}_2 - \mathbf{s}_2) \right) \\ \hat{\mathbf{t}}_3^+ = \hat{\mathbf{t}}_3 + \alpha \left(S(\mathbf{u}_{23})^2 (\hat{\mathbf{t}}_3 - \hat{\mathbf{t}}_2) + S(\mathbf{u}_{s_2 3})^2 (\hat{\mathbf{t}}_3 - \mathbf{s}_2) \right) \end{cases} \tag{4.25}$$

which can be written in matrix form as

$$\hat{\mathbf{T}}_3^+ = \hat{\mathbf{T}}_3 + \alpha(\mathbf{F} \cdot \hat{\mathbf{T}}_3 - \mathbf{G} \cdot \mathbf{S}_2) \tag{4.26}$$

where $\hat{\mathbf{T}}_3 = \begin{bmatrix} \hat{\mathbf{t}}_1 & \hat{\mathbf{t}}_2 & \hat{\mathbf{t}}_3 \end{bmatrix}^T$, $\mathbf{S}_2 = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 \end{bmatrix}^T$ and

$$\mathbf{G} = \begin{bmatrix} S(\mathbf{u}_{s_1 1})^2 & \mathbf{0} \\ \mathbf{0} & S(\mathbf{u}_{s_2 2})^2 \\ \mathbf{0} & S(\mathbf{u}_{s_2 3})^2 \end{bmatrix} \tag{4.27}$$

$$\mathbf{F} = \begin{bmatrix} S(\mathbf{u}_{s_1 1})^2 + S(\mathbf{u}_{31})^2 & \mathbf{0} & -S(\mathbf{u}_{31})^2 \\ -S(\mathbf{u}_{12})^2 & S(\mathbf{u}_{s_2 2})^2 + S(\mathbf{u}_{12})^2 & \mathbf{0} \\ \mathbf{0} & -S(\mathbf{u}_{23})^2 & S(\mathbf{u}_{s_2 3})^2 + S(\mathbf{u}_{23})^2 \end{bmatrix} = \begin{bmatrix} \mathbf{A_F} & \mathbf{B_F} \\ \mathbf{C_F} & \mathbf{D_F} \end{bmatrix} \tag{4.28}$$

with

$$\mathbf{A_F} = \begin{bmatrix} S(\mathbf{u}_{s_1 1})^2 + S(\mathbf{u}_{31})^2 & \mathbf{0} \\ -S(\mathbf{u}_{12})^2 & S(\mathbf{u}_{s_2 2})^2 + S(\mathbf{u}_{12})^2 \end{bmatrix}, \quad \mathbf{B_F} = \begin{bmatrix} -S(\mathbf{u}_{31})^2 \\ \mathbf{0} \end{bmatrix}$$
$$\mathbf{C_F} = \begin{bmatrix} \mathbf{0} & -S(\mathbf{u}_{23})^2 \end{bmatrix}, \qquad \mathbf{D_F} = \begin{bmatrix} S(\mathbf{u}_{s_2 3})^2 + S(\mathbf{u}_{23})^2 \end{bmatrix} \tag{4.29}$$

Taking into account that

$$\det(\mathbf{F}) = \prod_{i=1}^{9} \lambda_i \tag{4.30}$$

where λ_i represent the eigenvalues of $\mathbf{F}$ and

$$\det(\mathbf{F}) = \det(\mathbf{A_F}) \det(\mathbf{D_F} - \mathbf{C_F} {\mathbf{A_F}}^{-1} \mathbf{B_F}) = \det(\mathbf{D_F}) \det(\mathbf{A_F} - \mathbf{B_F} {\mathbf{D_F}}^{-1} \mathbf{C_F}) \tag{4.31}$$

it is possible to conclude that both $\mathbf{A_F}$ and $\mathbf{D_F}$ have negative eigenvalues and $\det(\mathbf{A_F}) \neq 0$ and $\det(\mathbf{D_F}) \neq 0$, which are necessary but not sufficient conditions for matrix $\mathbf{F}$ to have only negative eigenvalues. It remains to be proven that $\det(\mathbf{D_F} - \mathbf{C_F} {\mathbf{A_F}}^{-1} \mathbf{B_F}) \neq 0$ or $\det(\mathbf{A_F} - \mathbf{B_F} {\mathbf{D_F}}^{-1} \mathbf{C_F}) \neq 0$ to ensure that $\det(\mathbf{F}) \neq 0$. Further investigation is required to determine the sufficient conditions for $\mathbf{F}$ to have all its eigenvalues negative.

Since the true Target positions also satisfy (4.26), the error system with state given by $\tilde{\mathbf{T}}_3 = \hat{\mathbf{T}}_3 - \mathbf{T}_3$ can be written as

$$\tilde{\mathbf{T}}_3^+ = (\mathbf{I} + \alpha \mathbf{F}) \tilde{\mathbf{T}}_3 \tag{4.32}$$

meaning that $\tilde{\mathbf{T}}_3$ will converge to zero if $(\mathbf{I} + \alpha \mathbf{F})^T (\mathbf{I} + \alpha \mathbf{F}) \prec \mathbf{I}$ or $\rho(\mathbf{I} + \alpha \mathbf{F}) < 1$, being $\rho(\cdot)$ the spectral

radius of a matrix. Given that

$$\mathbf{F}\mathbf{v}_i = \lambda_i \mathbf{v}_i \tag{4.33}$$

and so

$$(\mathbf{I} + \alpha\mathbf{F})\mathbf{v}_i = (1 + \alpha\lambda_i)\mathbf{v}_i \tag{4.34}$$

the spectral radius can then be written as

$$\rho(\mathbf{I} + \alpha\mathbf{F}) = \max_{\lambda_i} |1 + \alpha\lambda_i| < 1 \tag{4.35}$$

which gives the following conditions

$$-2 < \alpha\lambda_i < 0 \tag{4.36}$$

Assuming that $\mathbf{F}$ has only negative eigenvalues and by implying that $\rho(\mathbf{I} + \alpha\mathbf{F}) < 1$, for sufficiently small α, the convergence of the estimation process is guaranteed.

Under the assumption that the estimator converges, $\hat{\mathbf{t}}_i^+ = \hat{\mathbf{t}}_i$ for $i = 1, 2, 3$, and (4.25) becomes

$$\begin{cases} \big(S(\mathbf{u}_{s_1 1})^2(\hat{\mathbf{t}}_1 - \mathbf{s}_1) + S(\mathbf{u}_{31})^2(\hat{\mathbf{t}}_1 - \hat{\mathbf{t}}_3)\big) = 0 \\ \big(S(\mathbf{u}_{12})^2(\hat{\mathbf{t}}_2 - \hat{\mathbf{t}}_1) + S(\mathbf{u}_{s_2 2})^2(\hat{\mathbf{t}}_2 - \mathbf{s}_2)\big) = 0 \\ \big(S(\mathbf{u}_{23})^2(\hat{\mathbf{t}}_3 - \hat{\mathbf{t}}_2) + S(\mathbf{u}_{s_2 3})^2(\hat{\mathbf{t}}_3 - \mathbf{s}_2)\big) = 0 \end{cases} \tag{4.37}$$

or in matrix form

$$\mathbf{F} \cdot \hat{\mathbf{T}}_3 = \mathbf{G} \cdot \mathbf{S}_2 \tag{4.38}$$

If the solution is unique and corresponds to the correct Target positions, then the following must hold

$$\begin{cases} S(\mathbf{u}_{s_1 1})^2(\hat{\mathbf{t}}_1 - \mathbf{s}_1) = 0 \\ S(\mathbf{u}_{31})^2(\hat{\mathbf{t}}_1 - \hat{\mathbf{t}}_3) = 0 \\ S(\mathbf{u}_{12})^2(\hat{\mathbf{t}}_2 - \hat{\mathbf{t}}_1) = 0 \\ S(\mathbf{u}_{s_2 2})^2(\hat{\mathbf{t}}_2 - \mathbf{s}_2) = 0 \\ S(\mathbf{u}_{23})^2(\hat{\mathbf{t}}_3 - \hat{\mathbf{t}}_2) = 0 \\ S(\mathbf{u}_{s_2 3})^2(\hat{\mathbf{t}}_3 - \mathbf{s}_2) = 0 \end{cases} \tag{4.39}$$

which can be transformed into a matrix form like

$$\mathbf{A} \cdot \hat{\mathbf{T}}_3 = \mathbf{D} \cdot \mathbf{S}_2 \tag{4.40}$$

where $\hat{\mathbf{T}}_3 = \begin{bmatrix} \hat{\mathbf{t}}_1 & \hat{\mathbf{t}}_2 & \hat{\mathbf{t}}_3 \end{bmatrix}^T$, $\mathbf{S}_2 = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 \end{bmatrix}^T$ and

$$\mathbf{A} = \begin{bmatrix} S(\mathbf{u}_{s_1 1})^2 & \mathbf{0} & \mathbf{0} \\ S(\mathbf{u}_{31})^2 & \mathbf{0} & -S(\mathbf{u}_{31})^2 \\ -S(\mathbf{u}_{12})^2 & S(\mathbf{u}_{12})^2 & \mathbf{0} \\ \mathbf{0} & S(\mathbf{u}_{s_2 2})^2 & \mathbf{0} \\ \mathbf{0} & -S(\mathbf{u}_{23})^2 & S(\mathbf{u}_{23})^2 \\ \mathbf{0} & \mathbf{0} & S(\mathbf{u}_{s_2 3})^2 \end{bmatrix}, \qquad \mathbf{D} = \begin{bmatrix} S(\mathbf{u}_{s_1 1})^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & S(\mathbf{u}_{s_2 2})^2 \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & S(\mathbf{u}_{s_2 3})^2 \end{bmatrix} \tag{4.41}$$

The uniqueness of the stationary solution is guaranteed if and only if matrix $\mathbf{A}$ has full rank. The rank of the symbolic matrix $\mathbf{A}$ can be easily obtained in a MATLAB environment and the results show that matrix $\mathbf{A}$ has indeed full rank, unless the pairs of measurements for each Target are collinear, and therefore the DES step in the construction method described in Subsection 4.3 preserves the uniqueness of the stationary solution.

Expanding this network to N Targets, the addition of another Target using the DES step will lead to

$$\hat{\mathbf{T}}^+_{N+1} = \hat{\mathbf{T}}_{N+1} + \alpha(\mathbf{F}_{N+1} \cdot \hat{\mathbf{T}}_{N+1} - \mathbf{G}_{N+1} \cdot \mathbf{S}_2) \tag{4.42}$$

with $\hat{\mathbf{T}}_{N+1} = \begin{bmatrix} \hat{\mathbf{t}}_1 & \hat{\mathbf{t}}_2 \dots \hat{\mathbf{t}}_{N+1} \end{bmatrix}^T$, $\mathbf{S}_2 = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 \end{bmatrix}^T$ and

$$\mathbf{F}_{N+1} = \begin{bmatrix} \mathbf{F}_N & \mathbf{B}_{N+1} \\ \mathbf{C}_{N+1} & S(\mathbf{u}_{iN+1})^2 + S(\mathbf{u}_{jN+1})^2 \end{bmatrix} \tag{4.43}$$

where $\mathbf{F}_N$ has negative eigenvalues and $\mathbf{B}_{N+1}$, $\mathbf{C}_{N+1}$ and $S(\mathbf{u}_{iN+1})^2 + S(\mathbf{u}_{jN+1})^2$ are blocks representing the new connections due to the addition of a new Target. Again, even though the necessary conditions $\det(\mathbf{F}_N) \neq 0$ and $\det(S(\mathbf{u}_{iN+1})^2 + S(\mathbf{u}_{jN+1})^2) \neq 0$ are met, further investigation is required to find the sufficient conditions for the matrix $\mathbf{F}_{N+1}$ to have all its eigenvalues negative.

Since the true Target positions also satisfy (4.42), the error system with state given by $\tilde{\mathbf{T}}_{N+1} = \hat{\mathbf{T}}_{N+1} - \mathbf{T}_{N+1}$ can be written as

$$\tilde{\mathbf{T}}^+_{N+1} = (\mathbf{I} + \alpha \mathbf{F}_{N+1}) \tilde{\mathbf{T}}_{N+1} \tag{4.44}$$

meaning that $\tilde{\mathbf{T}}_{N+1}$ will converge to zero if $(\mathbf{I} + \alpha \mathbf{F}_{N+1})^T (\mathbf{I} + \alpha \mathbf{F}_{N+1}) \prec \mathbf{I}$ or $\rho(\mathbf{I} + \alpha \mathbf{F}_{N+1}) < 1$.

Bearing in mind the conditions deduced in (4.35) and (4.36), assuming that $\mathbf{F}$ has only negative eigenvalues and by implying that $\rho(\mathbf{I} + \alpha \mathbf{F}_{N+1}) < 1$, for sufficiently small α, the convergence of the estimation process is guaranteed.

Under the assumption that the estimator converges, $\hat{\mathbf{t}}^+_i = \hat{\mathbf{t}}_i$ for $i = 1, 2, .., N+1$, and (4.42) becomes

$$\mathbf{F}_{N+1} \cdot \hat{\mathbf{T}}_{N+1} = \mathbf{G}_{N+1} \cdot \mathbf{S}_2 \tag{4.45}$$

If the solution is unique and corresponds to the correct Target positions, then a system of $2N + 2$ equations will be obtained. This system of equations can be transformed into a matrix form like

$$\mathbf{A} \cdot \hat{\mathbf{T}}_{N+1} = \mathbf{D} \cdot \mathbf{S}_2 \tag{4.46}$$

The uniqueness of the stationary solution is guaranteed if and only if matrix $\mathbf{A}$ has full rank. If the network is created using the DVA and DES steps described in Subsection 4.3, then the solution obtained with a convergent estimation process will be unique, corresponding to the true Targets' positions. In the following section, it will be shown that this is indeed the case for a specific network created with these two construction steps.

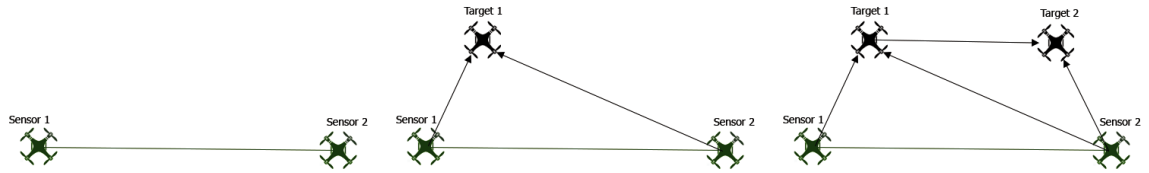
4.5 Results

4.5.1 Construction of a Rigid Formation

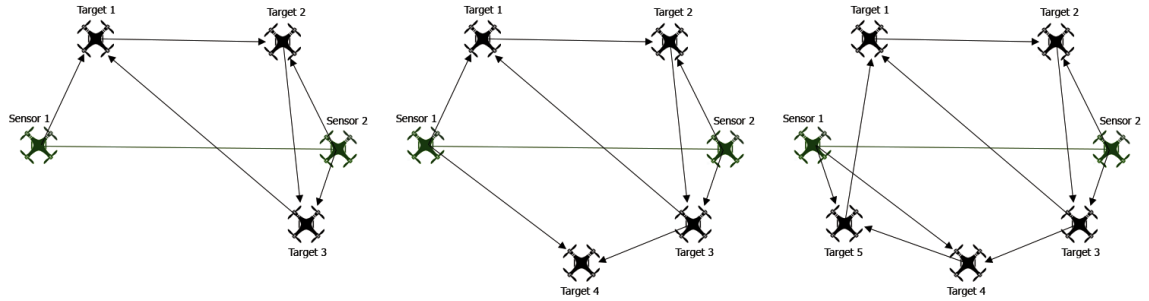
In this section, a network formation of two Sensors and five Targets is developed using the construction method described in Subsection 4.3. The network formation is built up in such a way that in the end there is no Target with two direct measurements from the 2 Sensors. Also, in order to construct a network involving every Target, each one should be used for at least one bearing measurement of another Target. The Targets need to be placed carefully in the network in order to avoid collinear bearing measurements of the same Target, i.e., there can exist collinear bearing measurements as long as they are directed to different Targets.

The network formation which will be studied from here on is the one depicted in Figure 4.2(f). Many other possible formations could be chosen and the main idea is that this can be applied in a greater scale by including even more Targets. As stated in Subsection 4.3, the construction of a rigid formation starts with a single link between two anchors (Sensors 1 and 2), thus initializing its grounded graph (Figure 4.2(a)). Then, Target 1 and Target 2 are included using the directed vertex addition step as depicted in Figures 4.2(b) and 4.2(c), respectively. Target 3 is added using the directed edge splitting step where the existing directed edge connecting Sensor 2 with Target 1 is removed and a new vertex representing Target 3 is added with three directed edges: one from Sensor 2 to Target 3, another from Target 2 to Target 3 and finally one link from Target 3 to Target 1 (Figure 4.2(d)). In Figure 4.2(e), Target 4 is added resorting to the directed vertex addition step and lastly Target 5 is introduced in the network formation by using the directed edge splitting step, where the existing directed edge connecting Sensor 1 with Target 1 is removed and a new vertex representing Target 5 is added with three directed edges: one from Sensor 1 to Target 5, another from Target 4 to Target 5 and finally one link from Target 5 to Target 1 (Figure 4.2(f)).

Note that, in the end, Target 1 has two directed links coming from two different Targets implying that Target 1 has no direct measurements from neither Sensor 1 nor Sensor 2, which have full knowledge of their localization. Also, after analysing the final network formation it is possible to conclude that the requirements set in the beginning of this section are all fulfilled, namely, there is no Target with two direct measurements from the 2 Sensors, every Target is used for at least one bearing measurement and there are no collinear bearing measurements.



(a) Grounded graph with two anchors (Sensors 1 and 2) (b) Directed vertex addition (Target 1) (c) Directed vertex addition (Target 2)



(d) Directed edge splitting (Target 3) (e) Directed vertex addition (Target 4) (f) Directed edge splitting (Target 5)

Figure 4.2: Steps in the Construction of a Network Formation with 2 Sensors and 5 Targets

4.5.2 Convergence and Uniqueness of the Stationary Solution

As mentioned in Subsection 4.4, the convergence and uniqueness of the stationary solution obtained when using a well-designed estimator like the "Stationary Target" estimator can be demonstrated by analysing the rank of matrix $\mathbf{A}$ and the condition $\rho(\mathbf{I} + \alpha\mathbf{F}) < 1$, as in (4.46) and (4.42).

For the network formation constructed in Subsection 4.5.1 and depicted in Figure 4.2(f), the dynamics of the estimation process can be expressed by the following system

$$\begin{cases} \mathbf{s}_1^+ = \mathbf{s}_1 \\ \mathbf{s}_2^+ = \mathbf{s}_2 \\ \hat{\mathbf{t}}_1^+ = \hat{\mathbf{t}}_1 + \alpha\left(S(\mathbf{u}_{51})^2(\hat{\mathbf{t}}_1 - \hat{\mathbf{t}}_5) + S(\mathbf{u}_{31})^2(\hat{\mathbf{t}}_1 - \hat{\mathbf{t}}_3)\right) \\ \hat{\mathbf{t}}_2^+ = \hat{\mathbf{t}}_2 + \alpha\left(S(\mathbf{u}_{s_2 2})^2(\hat{\mathbf{t}}_2 - \mathbf{s}_2) + S(\mathbf{u}_{12})^2(\hat{\mathbf{t}}_2 - \hat{\mathbf{t}}_1)\right) \\ \hat{\mathbf{t}}_3^+ = \hat{\mathbf{t}}_3 + \alpha\left(S(\mathbf{u}_{s_2 3})^2(\hat{\mathbf{t}}_3 - \mathbf{s}_2) + S(\mathbf{u}_{23})^2(\hat{\mathbf{t}}_3 - \hat{\mathbf{t}}_2)\right) \\ \hat{\mathbf{t}}_4^+ = \hat{\mathbf{t}}_4 + \alpha\left(S(\mathbf{u}_{s_1 4})^2(\hat{\mathbf{t}}_4 - \mathbf{s}_1) + S(\mathbf{u}_{34})^2(\hat{\mathbf{t}}_4 - \hat{\mathbf{t}}_3)\right) \\ \hat{\mathbf{t}}_5^+ = \hat{\mathbf{t}}_5 + \alpha\left(S(\mathbf{u}_{s_1 5})^2(\hat{\mathbf{t}}_5 - \mathbf{s}_1) + S(\mathbf{u}_{45})^2(\hat{\mathbf{t}}_5 - \hat{\mathbf{t}}_4)\right) \end{cases} \tag{4.47}$$

where, for the sake of simplicity, the notation $\hat{\mathbf{t}} \equiv \hat{\mathbf{t}}(t)$ and $\hat{\mathbf{t}}^+ \equiv \hat{\mathbf{t}}(t+1)$ was adopted. Also, $\mathbf{u}_{ij}$ and $\mathbf{u}_{s_i j}$ are the normalized relative vectors between the Target i or Sensor i, respectively, and Target j given by

$$\mathbf{u}_{ij} = \frac{\mathbf{t}_j - \mathbf{t}_i}{\|\mathbf{t}_j - \mathbf{t}_i\|} \tag{4.48}$$

$$\mathbf{u}_{s_i j} = \frac{\mathbf{t}_j - \mathbf{s}_i}{\|\mathbf{t}_j - \mathbf{s}_i\|} \tag{4.49}$$

As shown in Subsection 2.4.3, the weight value of α is subject to limitations. Even though it is possible

to design an estimator in which different weights are attributed to each measurement, a global constant value of α is chosen to avoid complications.

Using the notation introduced in Section 4.4 together with (4.47) yields

$$\hat{\mathbf{T}}_5^+ = \hat{\mathbf{T}}_5 + \alpha(\mathbf{F} \cdot \hat{\mathbf{T}}_5 - \mathbf{G} \cdot \mathbf{S}_2) \tag{4.50}$$

where $\hat{\mathbf{T}}_5 = \begin{bmatrix} \hat{\mathbf{t}}_1 & \hat{\mathbf{t}}_2 & \hat{\mathbf{t}}_3 & \hat{\mathbf{t}}_4 & \hat{\mathbf{t}}_5 \end{bmatrix}^T$, $\mathbf{S}_2 = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 \end{bmatrix}^T$ and

$$\mathbf{F} = \begin{bmatrix} S(\mathbf{u}_{51})^2 + S(\mathbf{u}_{31})^2 & \mathbf{0} & -S(\mathbf{u}_{31})^2 & \mathbf{0} & -S(\mathbf{u}_{51})^2 \\ -S(\mathbf{u}_{12})^2 & S(\mathbf{u}_{s_2 2})^2 + S(\mathbf{u}_{12})^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -S(\mathbf{u}_{23})^2 & S(\mathbf{u}_{s_2 3})^2 + S(\mathbf{u}_{23})^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -S(\mathbf{u}_{34})^2 & S(\mathbf{u}_{s_1 4})^2 + S(\mathbf{u}_{34})^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -S(\mathbf{u}_{45})^2 & S(\mathbf{u}_{s_1 5})^2 + S(\mathbf{u}_{45})^2 \end{bmatrix} \tag{4.51}$$

,

$$\mathbf{G} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & S(\mathbf{u}_{s_2 2})^2 \\ \mathbf{0} & S(\mathbf{u}_{s_2 3})^2 \\ S(\mathbf{u}_{s_1 4})^2 & \mathbf{0} \\ S(\mathbf{u}_{s_1 5})^2 & \mathbf{0} \end{bmatrix} \tag{4.52}$$

The error system with state given by $\tilde{\mathbf{T}}_5 = \hat{\mathbf{T}}_5 - \mathbf{T}_5$ can be written as

$$\tilde{\mathbf{T}}_5^+ = (\mathbf{I} + \alpha\mathbf{F})\tilde{\mathbf{T}}_5 \tag{4.53}$$

Since for this network localization problem every Target has 2 bearing measurements, a constant positive value of $\alpha < 1$ must be selected, as mentioned in Subsection 2.4.3. Also, as an additional constraint, α must be chosen such that $\rho(\mathbf{I} + \alpha\mathbf{F}) < 1$.

When the estimations converge

$$\hat{\mathbf{t}}_i^+ = \hat{\mathbf{t}}_i \qquad \text{for } i = 1, \ldots, 5 \tag{4.54}$$

and a system of 10 equations derived from (4.47) can be rewritten into matrix form

$$\mathbf{A} \cdot \hat{\mathbf{T}}_5 = \mathbf{D} \cdot \mathbf{S}_2 \tag{4.55}$$

where $\hat{\mathbf{T}}_5 = \begin{bmatrix} \hat{\mathbf{t}}_1 & \hat{\mathbf{t}}_2 & \hat{\mathbf{t}}_3 & \hat{\mathbf{t}}_4 & \hat{\mathbf{t}}_5 \end{bmatrix}^T$, $\mathbf{S}_2 = \begin{bmatrix} \mathbf{s}_1 & \mathbf{s}_2 \end{bmatrix}^T$ and the matrices $\mathbf{A}$ and $\mathbf{D}$ are

$$\mathbf{A} = \begin{bmatrix} S(\mathbf{u}_{51})^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & -S(\mathbf{u}_{51})^2 \\ S(\mathbf{u}_{31})^2 & \mathbf{0} & -S(\mathbf{u}_{31})^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & S(\mathbf{u}_{s_2 2})^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -S(\mathbf{u}_{12})^2 & S(\mathbf{u}_{12})^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & S(\mathbf{u}_{s_2 3})^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -S(\mathbf{u}_{23})^2 & S(\mathbf{u}_{23})^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & S(\mathbf{u}_{s_1 4})^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & -S(\mathbf{u}_{34})^2 & S(\mathbf{u}_{34})^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & S(\mathbf{u}_{s_1 5})^2 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -S(\mathbf{u}_{45})^2 & S(\mathbf{u}_{45})^2 \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & S(\mathbf{u}_{s_2 2})^2 \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & S(\mathbf{u}_{s_2 3})^2 \\ \mathbf{0} & \mathbf{0} \\ S(\mathbf{u}_{s_1 4})^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ S(\mathbf{u}_{s_1 5})^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{4.56}$$

From (4.56), (4.48) and (4.49) it is possible to conclude that matrix $\mathbf{A}$ is independent of the Target position estimations $\hat{\mathbf{t}}_i$ for $i = 1, \ldots, 5$ and only depends on the true positions of every element in the network formation. Therefore, the rank of matrix $\mathbf{A}$ and the spectral radius of $(\mathbf{I} + \alpha\mathbf{F})$ can be calculated straightforward after having defined the exact locations of the Sensors and Targets in the network formation. In practice, the Targets do not know their exact location but can infer the normalized relative vectors $\mathbf{u}_{ij}$ or $\mathbf{u}_{s_i j}$ by means of bearings measurements, i.e., using the azimuth angle ϕ and also the elevation angle ψ. For example, for a bearing measurement between Target i and Target j, the components of the normalized relative vector $\mathbf{u}_{ij} = \begin{bmatrix} u_{ij_x} & u_{ij_y} & u_{ij_z} \end{bmatrix}^T$ would be given by

$$\begin{aligned} u_{ij_x} &= \sin(\phi_{ij})\cos(\psi_{ij}) \\ u_{ij_y} &= \cos(\phi_{ij})\sin(\psi_{ij}) \\ u_{ij_z} &= \sin(\psi_{ij}) \end{aligned} \tag{4.57}$$

In this scenario, the 2 Sensors are placed in the boundaries imposed by the bridge deck where they can obtain GPS position information to localize themselves. The 5 Targets are placed forming a pentagonal shape, as shown in Figure 4.3. As mentioned in previous sections, the value of the bridge width is $d = 14$ m. The origin of the graph is considered to be in the middle of the bridge at an altitude slightly lower than the bridge deck to allow the Targets to perform their structural investigation tasks. Therefore, Sensor 1 and Sensor 2 are placed in opposite sides of the bridge with coordinates $\mathbf{s}_1 = \begin{bmatrix} -\frac{d}{2} & 0 & 0 \end{bmatrix}^T$ and $\mathbf{s}_2 = \begin{bmatrix} \frac{d}{2} & 0 & 0 \end{bmatrix}^T$, respectively. Then, the five Target locations are also defined taking into account the bridge width d and in such a way that 180^o (collinear) measurements are non-existent: $\mathbf{t}_1 = \begin{bmatrix} -\frac{d}{6} & \frac{d}{2} & 0 \end{bmatrix}^T$, $\mathbf{t}_2 = \begin{bmatrix} \frac{d}{6} & \frac{d}{2} & 0 \end{bmatrix}^T$, $\mathbf{t}_3 = \begin{bmatrix} \frac{d}{3} & -\frac{d}{4} & 0 \end{bmatrix}^T$, $\mathbf{t}_3 = \begin{bmatrix} \frac{d}{3} & -\frac{d}{4} & 0 \end{bmatrix}^T$ $\mathbf{t}_4 = \begin{bmatrix} 0 & -\frac{d}{2} & 0 \end{bmatrix}^T$ and $\mathbf{t}_5 = \begin{bmatrix} -\frac{d}{3} & \frac{d}{4} & 0 \end{bmatrix}^T$.

All the figures are represented in 2 dimensions because, in this case, the Targets and Sensors are all in the same horizontal plane. Nevertheless, this is a 3D problem and the estimation of each 3-component Target position vector is performed using the Skew operator $S(\cdot)$ of a 3-dimensional vector $\mathbf{u}_{ij}$ (or $\mathbf{u}_{s_i j}$).

After having defined the positions of each element in the network formation and obtained the bearings measurements, the rank of matrix $\mathbf{A}$ and the spectral radius of $(\mathbf{I} + \alpha\mathbf{F})$ can be easily calculated with a

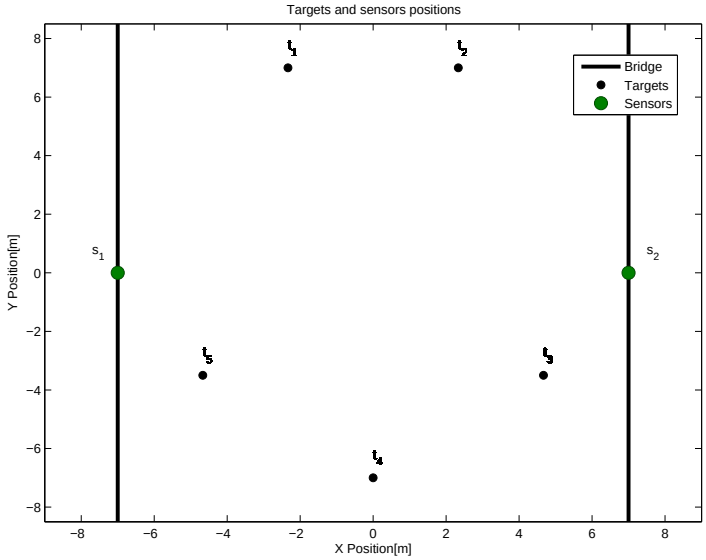


Figure 4.3: Network Formation with 2 Sensors and 5 Targets

MATLAB script. As expected, the matrix $\mathbf{A}$ of this network formation has full rank and therefore the uniqueness of the stationary solution is confirmed and corresponds to the exact locations of the Targets.

Also, with the value $\alpha = 0.6$ it follows that $\rho(\mathbf{I} + \alpha\mathbf{F}) < 1$, implying that the error $\tilde{\mathbf{T}}_5$ converges to zero. In Figure 4.4 the estimation process of the network formation is depicted. The initial estimates of the Targets' positions are randomly chosen in an area reasonably near the true positions. Nevertheless, the convergence of the estimator used is independent of the initial estimates and thus a blind guess would eventually lead to the intended solution, obviously taking more or less time to converge depending on the proximity of the initial guess to the true positions.

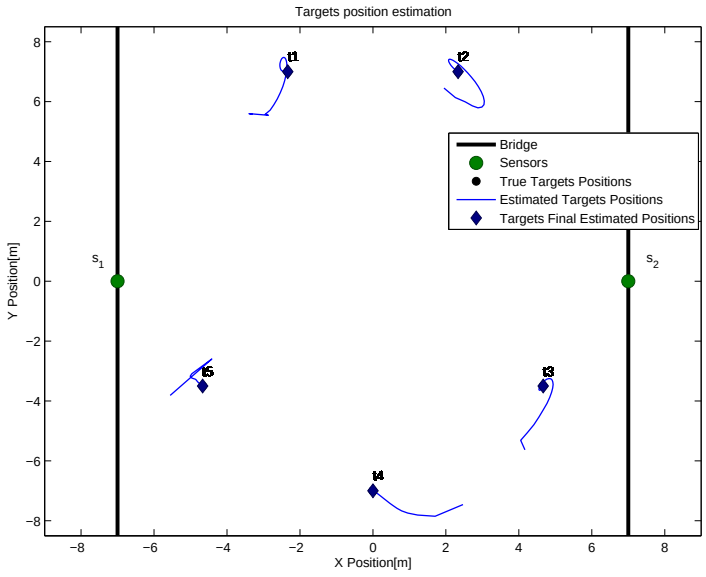


Figure 4.4: Network Formation Estimation Process with 2 Sensors and 5 Targets

In Figure 4.4, each blue line indicates the position estimate of the corresponding Target during the estimation process, having in its end a dark blue diamond which illustrates the final position estimate. Comparing Figures 4.3 and 4.4 it is clear that the estimates converge exactly to the true positions represented by black circles which are covered by the blue diamonds in Figure 4.4.

To analyse the convergence of the estimator over time, the comparison between the true and estimated position of all the five Targets is done in Figures 4.5, 4.6, 4.7, 4.8 and 4.9, in the 3D space. The fastness of the estimation algorithm is determined not only by the initial conditions, but also by the value of α. While lowering α would produce a slower but still convergent estimation, rising it too much would lead to instability and consequently the algorithm would not converge. With $\alpha = 0.6$, the estimation algorithm presents convergent results in a swift way. For instance, in Figure 4.5, in the y axis, the estimate of Target 1 y position is initially approximately 1.5 m away from the true value, taking about 3 seconds to reach an acceptable value for the estimate. It is crucial to bear in mind that each Target is estimating its position simultaneously and therefore any change in the estimate of a certain element of the network will have a chain effect in the other estimation processes. The complexity of this problem may be the cause to unpredictable behaviours in the estimation process as the one witnessed in the x axis graph of Figure 4.5, where the small bumps may indicate that the estimate of other Targets' positions was corrected and had impact in the estimation process of Target 1. Nevertheless, in the end the estimates always converge to the true values if the network formation is well constructed, as happens in this case. Moreover, it can be noted that the position of the Targets in the z axis is also estimated despite the fact that every element of the network formation is on the same horizontal plane.

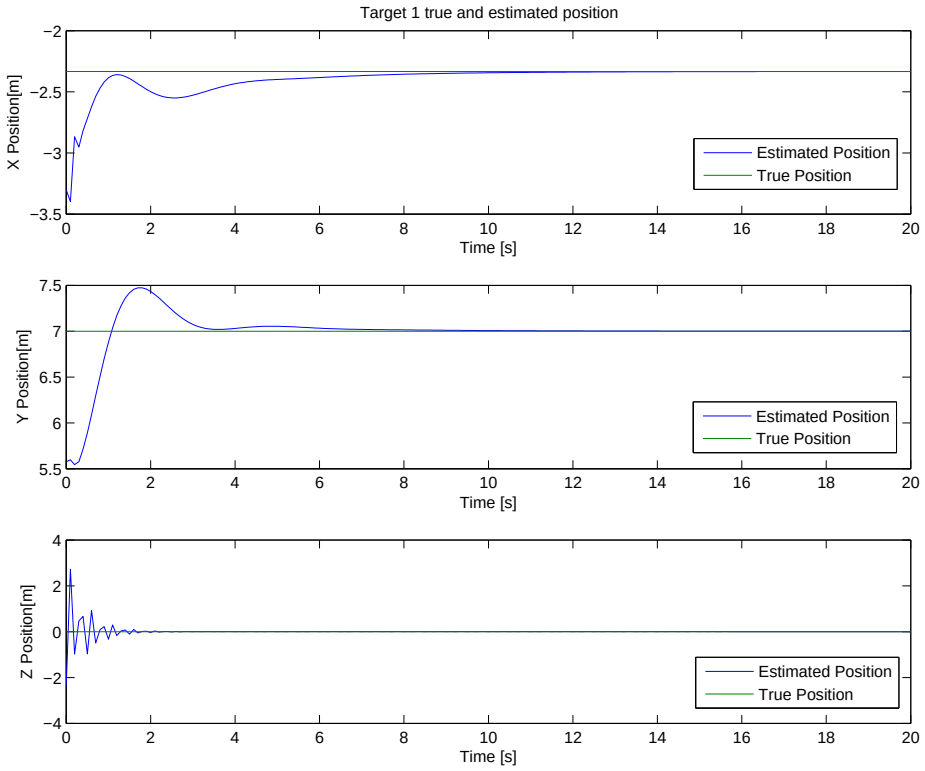


Figure 4.5: Target 1 True and Estimated x, y and z Position

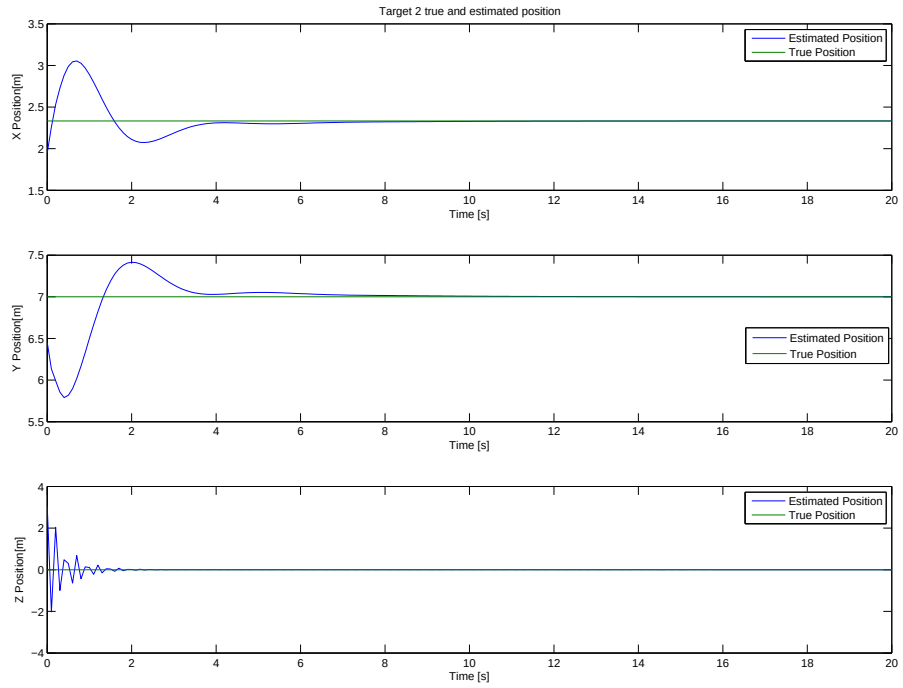


Figure 4.6: Target 2 True and Estimated x, y and z Position

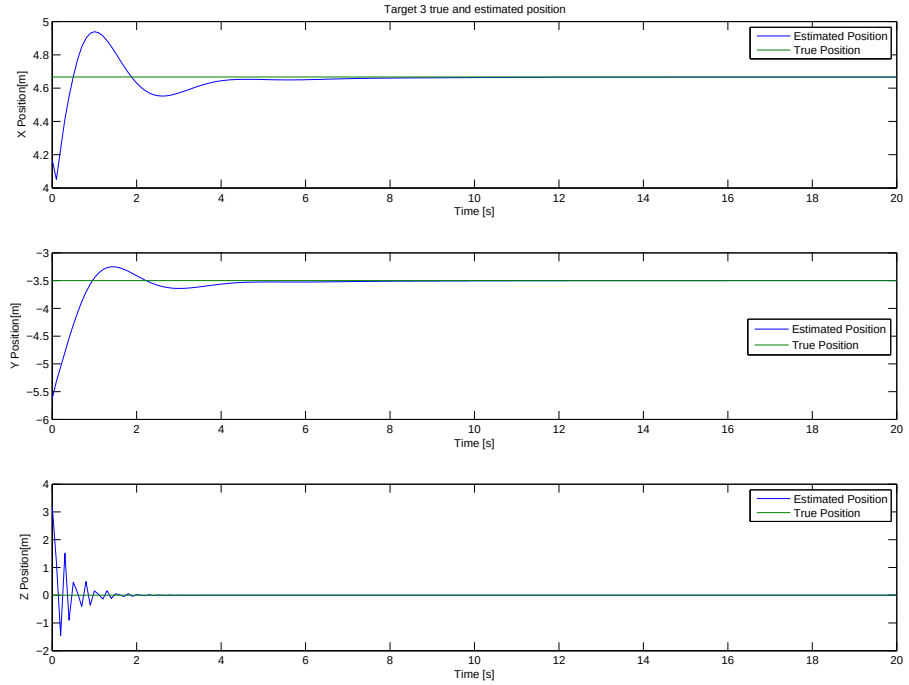


Figure 4.7: Target 3 True and Estimated x, y and z Position

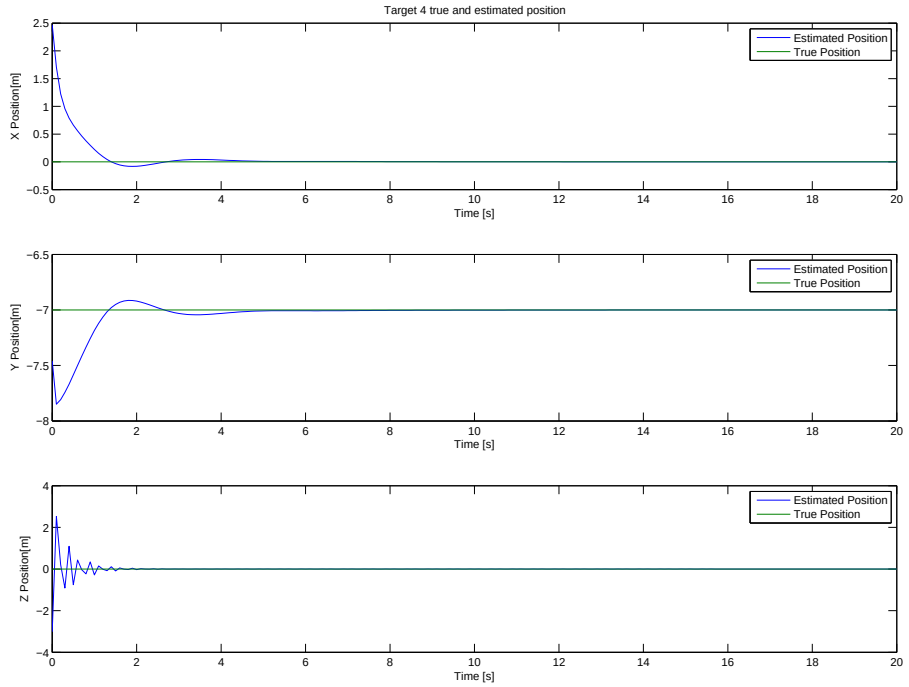


Figure 4.8: Target 4 True and Estimated x, y and z Position

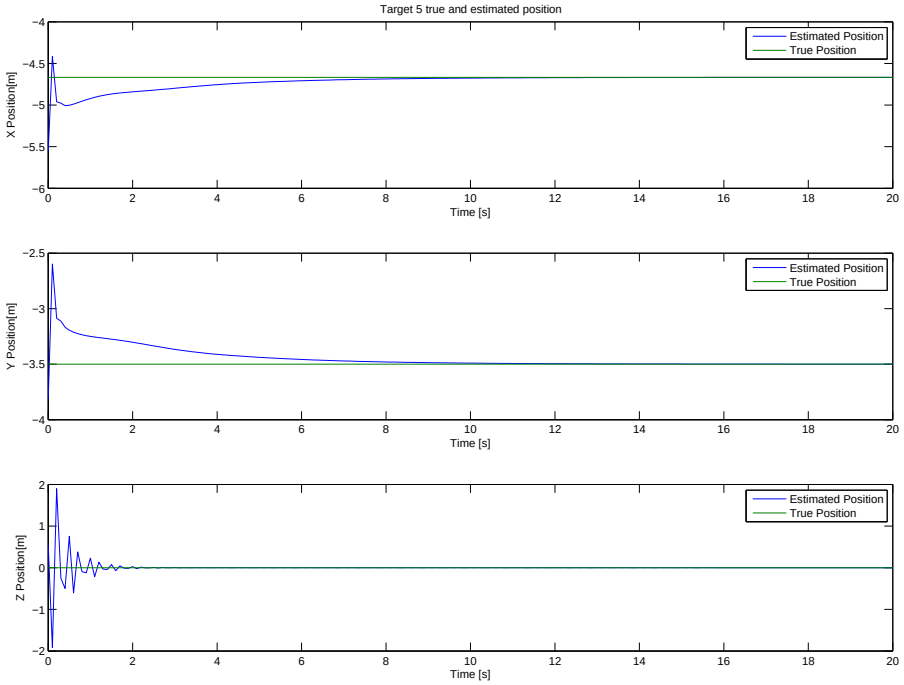


Figure 4.9: Target 5 True and Estimated x, y and z Position

Chapter 5

Conclusions and Future Work

5.1 Conclusions

This dissertation addresses the topic of simultaneous Target control, Target position estimation and Sensor-Target geometric configuration optimization. It also focus on the problem of network localization using bearings-only measurements. In Chapter 1, a more detailed description of the problem is done, followed by an overview of the most relevant state-of-the-art literature.

Chapter 2 provides a detailed analysis of the Target localization estimation problem. It starts by describing the Fisher Information Matrix and the Cramer-Rao Lower Bound which are used to quantify the performance of the estimation algorithms. Two novel estimators are designed to estimate the Target's position. The first, called "Stationary Target" estimator, is a simpler one which can only be used, as the name indicates, for estimating stationary Targets. The other estimator is the "Moving Target" estimator which is more complex but has the main advantage of estimating moving Targets. Chapter 2 also addresses the Target-Sensor geometric configuration with the goal of maximizing the information provided by the measurements so that the Target position estimation error is minimized. Several objective functions based on the FIM are tested, namely the D-optimality, which implicates calculating the determinant of the FIM, the E-optimality, also known as the smallest eigenvalue criterion, the A-optimality, involving the trace of the inverse of the FIM and the Sensitivity criterion, which calculates the trace of the FIM. These criteria are tested including the physical constraints imposed by the limits of the bridge and different geometric configurations are obtained, as shown in Figure 2.3. These geometric configurations are, in general, different from the ones obtained in literature mainly because it considers unconstrained optimization. For example, Bishop et al. [4] uses the D-optimality criterion in an unconstrained environment and obtains a 90^{o} configuration, different from the 120^{o} configuration achieved in this specific bridge scenario. After comparing the RMSE between the true and the estimated Target's position obtained with these criteria and with a non-optimal one named "Constant Offset", it can be concluded that, apart from the Sensitivity criterion, which ignore the angular information of the FIM, the optimality criteria are, in general, best suited for estimating the Target's position than the "Constant Offset" criterion. In the end, a situation where the Target is following a sinusoidal trajectory is studied and the different geometric configurations

obtained with the optimization of the various criteria described before are presented. It can be concluded that optimizing some objective functions such as the D-optimality, the E-optimality and the Sensitivity criteria leads to geometric configurations with a constant associated angle, whereas the A-optimality and "Constant Offset" criteria lead to more flexible geometric configurations, in which the angle subtended at the Target by the two Sensors fluctuates depending on the Target's x position.

In Chapter 3, a controller is added to command the Target's position according to a desired trajectory. After having designed the controller and implemented it in the simulation, the several criteria are tested and compared regarding not only the estimation error, but also the difference between the true and the desired position. The results obtained suggest that the E-optimality criterion leads to better estimation and control results in a scenario where the Target is stationary, whereas the A-optimality criterion is the best suited objective function for a scenario where the Target is following a sinusoidal trajectory. In both stationary and moving Target cases, the "Constant Offset" criterion has the highest errors and thus it is considered the worst of the four criteria studied.

Chapter 4 describes a possible construction method for directed rigid formations and demonstrates the convergence and uniqueness of the stationary solution constructed with that method. An example with five Targets and two Sensors is selected to show how the construction method previously developed can be put into practice and then the convergence and uniqueness of the stationary solution is confirmed by analysing the rank of a matrix which contains information about the existent links in the network and a condition involving the spectral radius of another matrix. If the first matrix has full rank and the condition is met, then the estimates will converge to the unique solution. The "Stationary Target" estimator is used to perform all the estimations of the Targets' positions and, in the end, as expected, the estimates of each Target converge to their exact location. The fastness of the estimation convergence is determined not only by the initial estimates of the positions, but also by the value of a constant α, used in the design of the "Stationary Target" estimator.

5.2 Future Work

There are several possible extensions of this work and some ideas are presented in this section. First of all, the next step to be considered should be the implementation of the algorithms developed in this work on real quadrotors, with all the practical issues which were not taken into account. Also, the quadrotors' specific dynamics are not considered and a suggestion for future work would be to incorporate the dynamics models in the simulation for it to become more realistic, thus improving the estimation performance in real world. Constraints on the quadrotor's maximum velocity due to hardware limitations are not contemplated and could be included in the future.

Another suggestion for improvement involves refining the optimization and estimation processes. Instead of using the current Target position estimate in the optimization process, better results could be achieved if an estimate of the future Target localization was used. This would have greater impact if the time needed to obtain the optimal Sensor-Target configuration was quantified. Other optimization considerations could also include additional operational constraints involving obstacles in the environment

such as trees or even the bridge's columns.

Lastly, as far as the network localization problem is concerned, an interesting extension could be the transposition of all the theory developed in Chapter 4 into reality, by testing it on real quadrotors. Further investigation can be made in order to find the sufficient conditions that prove the convergence of the estimation process when adding a new Target using the Directed Edge Splitting (DES) step. Chapter 4 deals with fixed networks and a stimulating challenge would be to develop a theory of localizable networks that, instead of stationary Targets, considers moving ones, under certain conditions.

Bibliography

[1] J. Aspnes, T. Eren, D. Goldenberg, a.S. Morse, W. Whiteley, Y. Yang, B. Anderson, and P. Belhumeur. A Theory of Network Localization. *IEEE Transactions on Mobile Computing*, 5(12):1–35, 2006. ISSN 1536-1233. doi: 10.1109/TMC.2006.174.

[2] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*, volume 9. 2001. ISBN 047141655X. doi: 10.1002/0471221279.

[3] K. L. Bell. MAP-PF position tracking with a network of sensor arrays. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, IV(4):849–852, 2005. ISSN 15206149. doi: 10.1109/ICASSP.2005.1416142.

[4] A. N. Bishop, B. Fidan, B. D. O. Anderson, K. Dogançay, and P. N. Pathirana. Optimality analysis of sensor-target localization geometries. *Automatica*, 46:479–492, 2010. ISSN 00051098. doi: 10.1016/j.automatica.2009.12.003.

[5] T. Eren. Using angle of arrival (bearing) information for localization in robot networks. *Turkish Journal of Electrical Engineering and Computer Sciences*, 15(2):169–186, 2007. ISSN 13000632. doi: 10.1109/CDC.2006.376721.

[6] T. Eren, P. Belhumeur, B. Anderson, and a.S. Morse. A framework for maintaining formations based on rigidity. *Proceedings of the 15th IFAC World Congress, Barcelona, Spain*, pages 2752–2757, 2002.

[7] S. Hammel, P. Liu, E. Hilliard, and K. Gong. Optimal observer motion for localization with bearing measurements. *Computers & Mathematics with Applications*, 18(1-3):171–180, 1989. ISSN 08981221. doi: 10.1016/0898-1221(89)90134-X.

[8] J. M. Hendrickx, B. D. O. Anderson, and V. D. Blondel. Rigidity and persistence of directed graphs. *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference, CDC-ECC '05*, 2005(c):2176–2181, 2005. doi: 10.1109/CDC.2005.1582484.

[9] M. Hernandez. Optimal Sensor Trajectories in Bearings-Only Tracking. *Proceedings of the Seventh International Conference on Information Fusion, FUSION 2004,*, pages 893–900, 2004.

[10] S. Kay. *Fundamentals of statistical signal processing: estimation theory*, volume 37. 1993. doi: 10.2307/1269750.

[11] H. Khalil. *Nonlinear Systems*. 2002. ISBN 0130673897, 9780130673893.

[12] U. A. Khan. High dimensional consensus in large-scale networks: Theory and applications. *ProQuest Dissertations and Theses*, 3375175(August):240, 2009.

[13] G. Laman. On graphs and rigidity of plane skeletal structures. *Journal of Engineering Mathematics*, 4(4):331–340, 1970. ISSN 00220833. doi: 10.1007/BF01534980.

[14] E. L. Lehmann and G. Casella. *Theory of Point Estimation , Second Edition Springer Texts in Statistics*, volume 41. 1998. ISBN 0387985026. doi: 10.2307/1270597.

[15] S. Martínez and F. Bullo. Optimal sensor placement and motion coordination for target tracking. *Automatica*, 42:661–668, 2006. ISSN 00051098. doi: 10.1016/j.automatica.2005.12.018.

[16] S. Ponda. *Trajectory Optimization for Target Localization Using Small Unmanned Aerial Vehicles*. PhD thesis, 2008.

[17] S. A. Quintero, D. A. Copp, and J. P. Hespanha. Robust uav coordination for target tracking using output-feedback model predictive control with moving horizon estimation.

[18] S. a. P. Quintero, F. Papi, D. J. Klein, L. Chisci, and P. Hespanha. Optimal UAV Coordination for Target Tracking using Dynamic Programming. *Decision and Control, (CDC), 2010 49th IEEE Conference on*, pages 4541–4546, 2010. ISSN 0743-1546. doi: 10.1109/CDC.2010.5717933.

[19] B. Ristic and M. S. Arulampalam. Tracking a manoeuvring target using angle-only measurements: Algorithms and performance. *Signal Processing*, 83(6):1223–1238, 2003. ISSN 01651684. doi: 10.1016/S0165-1684(03)00042-2.

[20] J. H. T. J. H. Taylor. The Cramer-Rao estimation error lower bound computation for deterministic nonlinear systems. *1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, 17(July), 1978. ISSN 0018-9286. doi: 10.1109/CDC.1978.268121.

[21] P. Tichavsky, C. Muravchik, and a. Nehorai. Posterior Cramer-Rao bounds for discrete-time nonlinear filtering. *IEEE Transactions on Signal Processing*, 46(5):1386–1396, 1998. ISSN 1053-587X. doi: 10.1109/78.668800.

[22] H. L. V. Trees. *Detection, Estimation, and Modulation Theory*, volume 6. 2001. ISBN 0471095176.

[23] O. Tremois and J. Le Cadre. Target motion analysis with multiple arrays: performance analysis. *IEEE Transactions on Aerospace and Electronic Systems*, 32(3), 1996. ISSN 0018-9251. doi: 10.1109/7.532262.

[24] D. Ucinski. *Optimal Measurement Methods for Distributed Parameter System Identification*. 2005. ISBN 9780849323133. doi: 10.1201/9780203026786.

[25] W. Whiteley. Some matroids from discrete applied geometry. *Contemporary Mathematics*, 197(July): 171–311, 1996.

[26] F. Z. F. Zhang, B. Grocholsky, and V. Kumar. Formations for localization of robot networks. *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, 4(2), 2004. ISSN 1050-4729. doi: 10.1109/ROBOT.2004.1308774.