TÉCNICO LISBOA

Range-Based Underwater Target Localization and Tracking

João Pedro Batista Lopes

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisors: Prof. António Manuel dos Santos Pascoal
Prof. Tor Johansen

Examination Committee

Chairperson: Prof. João Fernando Cardoso Silva Sequeira
Supervisor: Prof. António Manuel dos Santos Pascoal
Member of the Committee: Dr. Francisco José Curado Mendes Teixeira

February 2021

Declaration

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

Acknowledgments

First, I would like to thank my supervisors António Pascoal and Tor Arne Johansen for giving me the possibility of working in this thesis and always being optimistic about my progress. I would also like to thank them for providing me guidance and giving me the tools to learn about this topic.

I also wish to thank Praveen Jain for welcoming me to the offices at NTNU and providing motivating feedback to my work. The meetings and conversations we had were fundamental to guide me through this research. I am very grateful for his help and advice. In addition, I would like to thank Nguyen Hung from ISR for his guidance.

Lastly, I thank my family and friends for supporting me over the last five years and giving me motivation to continue even when difficulties appear.

Abstract

This work addresses the problem of multiple underwater target localization and tracking using acoustic range measurements in a marine scenario. Underwater range-based target localization consists in estimating the position of a single or multiple targets with range measurements from a set of Autonomous Surface Vehicles (ASV's), referred as trackers. In this scenario the trackers should plan their movements to maximize the range related information on the position of the targets. The goal of this work is to develop a strategy to compute optimal trajectories for the trackers that maximize the range information available.

Resorting to an Estimation theoretical setting, a suitable Fisher Information Matrix (FIM) is derived as a metric of the quantity of information about the position of the targets in the range measurements. First, assuming no restrictions on the trackers motion models and that the positions of the targets are known, optimal trajectories for the surface trackers which maximize the determinant of the (FIM) are discussed. Then, a Model Predictive Control (MPC) strategy is proposed to simultaneously estimate the position of the targets while finding optimal trajectories for the trackers restricted to their motion models. The performance of the proposed MPC strategy is assessed in simulations where multiple targets and trackers were considered.

Keywords

Range Based Target Tracking, Estimation Theory, Trajectory Optimization, Model Predictive Control

Resumo

Esta dissertação aborda o tema de localização e seguimento de múltiplos alvos subaquáticos usando medidas acústicas de distância num cenário marinho. A localização de alvos subaquáticos com recurso a medidas de distância consiste na estimação da posição de um ou mais alvos com medidas de distância a partir de um conjunto de Veículos Autónomos de Superfície (VAS), referidos como seguidores. Neste cenário os veículos seguidores deverão planear os seus movimentos de forma a maximizar a quantidade de informação presente nas medidas de distância acerca das posições dos alvos. O objectivo deste trabalho é desenvolver uma estratégia para calcular trajectórias óptimas para os veículos seguidores de forma a maximizar a informação disponível acerca das posições dos alvos.

Usando uma abordagem baseada na Teoria de Estimação, é utilizada a Matriz de Informação de Fisher (MIF) como métrica da quantidade de informação acerca da posição dos alvos nas medidas de distância. Primeiro, considerando não existirem restrições no modelo dinâmico dos veículos seguidores e que as posições dos alvos são conhecidas, são calculadas trajectórias óptimas para os seguidores que maximizam o determinante da MIF. De seguida, uma estratégia Controlo Preditivo é proposta para simultaneamente estimar a posição dos alvos enquanto trajectórias óptimas para os seguidores são calculadas respeitando as restrições impostas pelos respectivos modelos dinâmicos. A performance da estratégia de Controlo Preditivo é depois avaliada em simulações que consideram cenários com múltiplos alvos e múltiplos seguidores.

Palavras Chave

Seguimento de Alvos, Medidas de Distância, Teoria da Estimação, Optimização de Trajectórias, Controlo Preditivo

Contents

List of Figures

List of Tables

Acronyms

AUV	Autonomous Underwater Vehicle
ASV	Autonomous Surface Vehicle
CRLB	Cramer-Rao Lower Bound
EKF	Extended Kalman Filter
FIM	Fisher Information Matrix
IVP	Initial Value Problem
KBF	Kalman-Bucy Filter
KF	Kalman Filter
MPC	Model Predictive Control
NMPC	Nonlinear Model Predictive Control
NLP	Nonlinear Programming
PDF	Probability Density Function
OCP	Optimal Control Problem
ODE	Ordinary Differential Equation

1

Introduction

1.1 Motivation

The field of marine robotics has attracted increasing interest of the scientific research and commercial proposes over the last years. Within this field, the problem of underwater target localization has been of particular interest due to its multiple practical applications. Such applications include Autonomous Underwater Vehicle (AUV) aided navigation (see fig. 1.1), acoustic beacon localization, human diver localization, intruder vessel detection in maritime surveillance missions, and fish tracking to monitor migratory patterns, just to name a few. Due to the difficult conditions inherent to the underwater environment, such as the impossibility of using electromagnetic based communications, acoustic based measurements play a key role. In this setting, acoustic range-based systems for underwater target localization are of particular interest.

In its simplest form, the range-based target localization consists in estimating the position of a single or multiple submerged underwater targets with acoustic range measurements from a set of Autonomous Surface Vehicles (ASV's), referred as trackers. Since range measurements are used in the estimation process, the trajectories executed by the trackers have a direct impact on the estimation accuracy. This relation naturally leads to multiple problems such as observability analysis [2], or the problem to be addressed in this work, which is the study of "sufficiently exciting" motions for the surface trackers in order to maximize the range related information about the position of the targets.

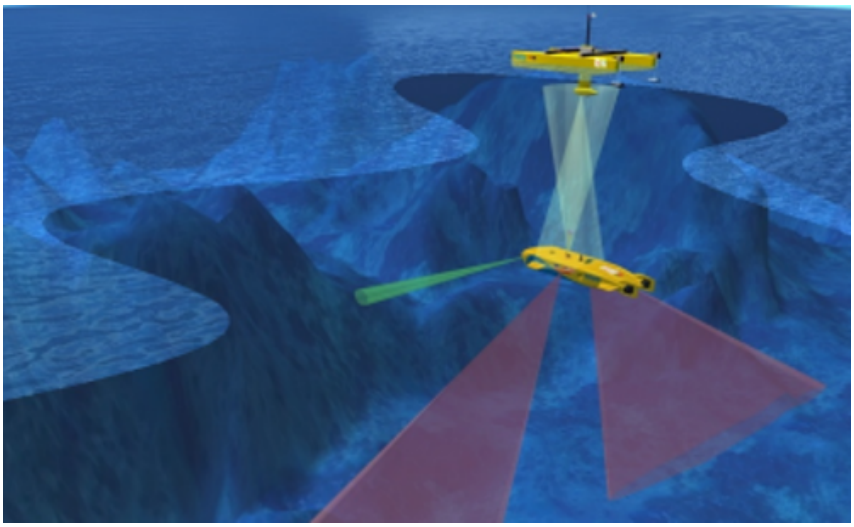


Figure 1.1: Underwater Autonomous Vehicle and its tracker

1.2 Proposed Work

The work proposed in this thesis is to study and design systems for range-based underwater target localization and tracking using single and multiple autonomous surface vehicles. The focus of the work

will be the generation of “sufficiently exciting motions” for the surface vehicles in order to track the underwater vehicles with the lowest uncertainty possible.

The tracking task will be performed by a set of surface vehicles using acoustic range measurements. In order to track the underwater vehicles, the measurements will be combined with the dynamic model of the targets and an Extended Kalman Filter will be used as a state estimator. This setup is shown in figure 1.2.

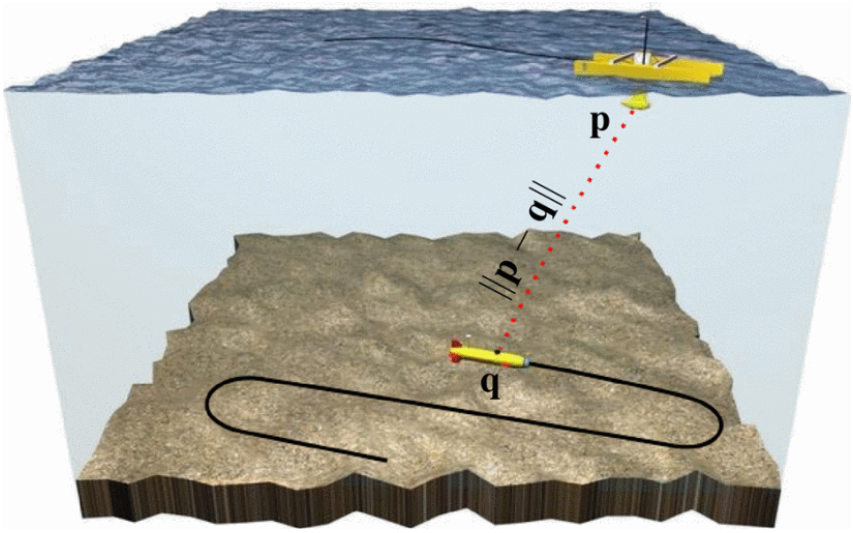


Figure 1.2: Target-Tracker setup with acoustic measurements of the ranges between the two vehicles

Estimation Theory tools such as the Fisher Information Matrix (FIM) will be used to define cost functions to encode the information about the positions of the targets. These cost functions will be used to determine the optimal positions for the trackers in order to localize the targets with the lowest uncertainty achievable. Since the objects to be tracked are moving underwater, the surface vehicles will have to move alongside them. The movements must be done in such way as to maximize the information about the targets and will be planned over a finite horizon using Model Predictive Control techniques. The trajectories planned are to be followed by the surface vehicles until new estimates of the positions of the targets becomes accessible. Every time a new estimation is performed, new optimal trajectories for the surface vehicles are computed.

1.3 Objectives

The objective of this thesis work is to design and study systems for range-based underwater target localization and tracking using single and multiple autonomous surface vehicles. The work will address the analysis of the best performance achievable and the generation of “sufficiently exciting motions” for the surface vehicles.

The work is divided in five parts:

- Conduct state of the art research regarding range-based target localization and tracking algorithms in a marine scenario.
- Definition of a suitable FIM as a measure of the quantity of information about the positions of the targets in the range measurements.
- Analysis of the best performance achievable which is characterized by tracker trajectories that maximize the determinant of the FIM.
- Development of a Model Predictive Control (MPC) strategy to simultaneously estimate the position of the targets while optimizing the trajectories of the trackers.
- Performance analysis of the proposed MPC strategy via numerical simulations.

1.4 Outline

After this introductory chapter, the following chapter will present a review of the related state-of-the-art and theoretical concepts with a summary of the most used filters in navigation systems (section 2.1), an introduction to the main tools from Estimation Theory (section 2.2), and a review of the existing work in the area of optimal sensor placement (section 2.4) and range-based target tracking (section 2.5)

Chapter 3 introduces the target tracking problem to be addressed in this thesis work. The target and tracker motion models are introduced as well as the measurement model. The problem is then formulated in section 3.4.

In chapter 4 ideal trajectories for the target tracking problem are computed for a different number of targets and trackers. First, the Fisher Information Matrix is derived as a measure of the quantity of information in the range measurements about the position of the targets. Then, using its determinant as a criterion to be maximized, ideal trajectories for a different number of surface trackers and underwater targets are discussed.

In chapter 5 the proposed MPC strategy is presented. Its three main components, optimal motion planing, motion control, and position estimation are discussed.

In chapter 6 the proposed MPC strategy is tested in simulations involving different scenarios. The results are analyzed while highlighting its capacities and limitations. The results obtained are compared with the ideal trajectories for the target tracking problem.

Chapter 7 concludes the thesis and summarizes the most notable observations. Also, possible future improvements are proposed and discussed.

2

Background Concepts and State-of-the-Art

In this section, the general theory used in navigation systems is presented. First, the filtering problem is formulated, and different filter structures are presented. The differences between stochastic and deterministic filter designs are exploited. Both discrete and continuous time filter structures are presented. Then, the main tools and concepts of Estimation Theory that will play an important role in this thesis work are reviewed. After that, a general form of an Optimal Control Problem (OCP) is presented. Since the main goal of this thesis is to implement a MPC strategy for target tracking, an OCP will arise to determine the optimal controls for the trackers to maximize the information about the positions of the targets. This review of concepts is complemented by presenting the existing work on optimal sensor placement, and range-based target tracking .

2.1 Filtering

2.1.1 Complementary Filters

Complementary Filters haven been studied to solve navigation problems, see [3] for a practical example. This kind of filters addresses explicitly the problem of merging information from sensors that operate over different, but complementary, frequency regions. The design of such filters on a time-invariant setting is purely deterministic in a sense that it is reduced to the problem of determining the frequency of a low-pass filter that matches the frequency characteristics of the sensors being used. Therefore, the low pass filter bandwidth becomes the tuning parameter in this framework.

A simple example of a Complementary Filter design for navigation systems is presented in [4] where the authors derive a filter for estimating the heading ψ of a vehicle based on measurements r_m and ψ_m of $r = \dot{\psi}$ and ψ respectively. This measurements are given by a rate gyro and a fluxgate compass. Such sensors operate in different frequency regions, while the compass performs well in low frequency domain the rate gyro provides biased information in the same frequency being more reliable at higher frequency regions. The example is described in figure 2.1 where r_d and ψ_d represent the rate gyro and compass measurement noises, respectively.

Let $\psi(s)$ and $r(s)$ being the Laplace transforms of the signals ψ and r respectively. Then for every $k > 0$, $\psi(s)$ can be decomposed as

$$\psi(s) = \frac{s+k}{s+k}\psi(s) = \frac{k}{s+k}\psi(s) + \frac{s}{s+k}\psi(s) = T_1(s)\psi(s) + T_2(s)\psi(s) \tag{2.1}$$

where $T_1(s) = k/(s+k)$ and $T_2(s) = s/(s+k)$. T_1 and T_2 satisfy the relation

$$T_1(s) + T_2(s) = I. \tag{2.2}$$

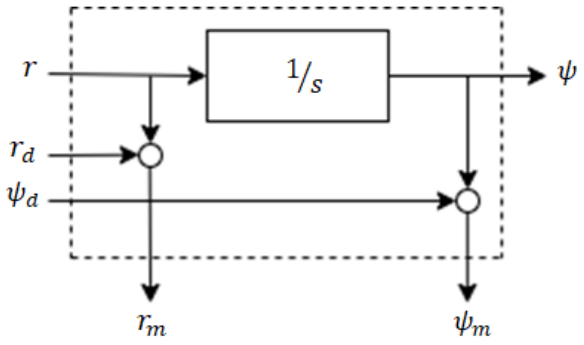


Figure 2.1: Process model

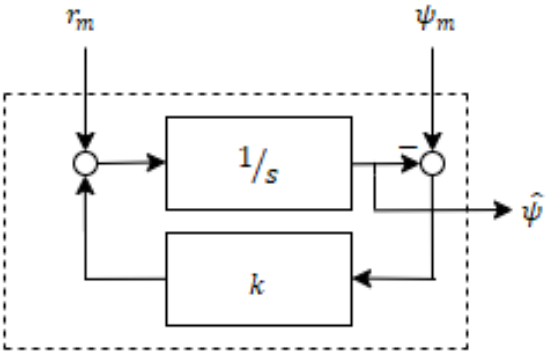


Figure 2.2: Complementary Filter Design

Since $r(s) = s\psi(s)$ from the above equations it follows that

$$\psi(s) = F_\psi(s)\psi(s) + F_r(s)r(s) \tag{2.3}$$

where $F_\psi(s) = T_1(s) = k/(s+k)$ and $F_r(s) = T_2(s) = s/(s+k)$. This relationship suggest the following filter structure

$$\hat{\psi} = \mathcal{F}_\psi \psi_m + \mathcal{F}_r r_m \tag{2.4}$$

where $\mathcal{F}_\psi$ and $\mathcal{F}_r$ are LTI operators with transfer functions $F_\psi(s)$ and $F_r(s)$ respectively. As represented in figure 2.2, the filter admits the state realization

$$\begin{aligned} \dot{\hat{\psi}} &= -k\hat{\psi} + k\psi_m + r_m \\ &= r_m + k(\psi_m - \hat{\psi}). \end{aligned} \tag{2.5}$$

Now let $\mathcal{T}_1$ and $\mathcal{T}_2$ be LTI operators with transfer functions $T_1(s)$ and $T_2(s)$ respectively. Simple

computations show that

$$\hat{\psi} = (\mathcal{T}_1 + \mathcal{T}_2)\psi + \mathcal{F}_\psi \psi_d + \mathcal{F}_r r_d. \tag{2.6}$$

From the equation above, and since $(\mathcal{T}_1 + \mathcal{T}_2)\psi = \psi$, it is evident that the estimate of the heading $\hat{\psi}$ is equal to the true value of ψ plus disturbance terms from the sensor measurements.

This filter structure derivation makes some relevant properties visible. The filter trusts the information provided by the compass at lower frequency, since $T_1(s)$ is low pass, and merges it with the information given by the rate gyro at higher frequencies since $T_2(s) = I - T_1(s)$. The break frequency is uniquely determined by k, making it the tuning parameter in the filter.

This approach is especially useful for practical situations where the characterization of the disturbances in a stochastic setting is not possible [4], with the filter design taking into account explicitly the sensor bandwidths.

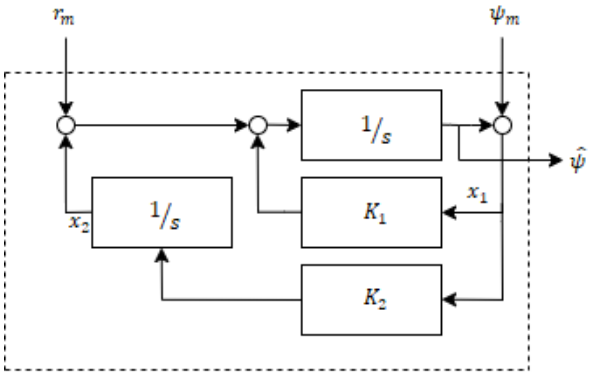


Figure 2.3: Complementary Filter with bias estimation

Also in [4], the authors present a more realistic situation where the filter is designed to accomplish steady state rejection of the rate gyro bias. To do so, the filter is augmented with an integrator, as represented in figure 2.3, and has the following state realization

$$\begin{aligned} \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \end{bmatrix} &= \begin{bmatrix} -k_1 & 1 \\ -k_2 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \psi + \begin{bmatrix} 1 \\ 0 \end{bmatrix} r_m \\ \hat{\psi} &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{aligned} \tag{2.7}$$

where x_1 and x_2 denote the states associated with the estimate of the heading $\hat{\psi}$ and the rate gyro bias respectively, and k_1 and k_2 the filter gains. The relation can be re-written as

$$\begin{cases} \dot{\mathbf{x}} = A\mathbf{x} + Bu + H(y - \hat{y}) \\ \hat{y} = C\mathbf{x} \end{cases} \tag{2.8}$$

where $\mathbf{x} = [x_1 x_2]^T$, $u = r_m$, $y = \psi_m$ and

$$\begin{aligned} A &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ C &= \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad H = \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}. \end{aligned} \tag{2.9}$$

In this case,

$$\hat{\psi} = (\mathcal{T}_1 + \mathcal{T}_2)\psi + \eta, \tag{2.10}$$

where

$$T_1(s) = \frac{k_1 s + k_2}{s^2 + k_1 s + k_2}, \quad T_2(s) = \frac{s^2}{s^2 + k_1 s + k_2} \tag{2.11}$$

and $\hat{\psi} = \mathcal{F}_\psi \psi_d + \mathcal{F}_r r_d$ represents the noise term. The noise intensity depends on $\mathcal{F}_\psi = T_1(s)$ and $\mathcal{F}_r = s/(s^2 + k_1 s + k_2)$. As it was stressed before, $T_1(s)$ and $T_2(s)$ are complementary since $T_1(s) + T_2(s) = I$, where $T_1(s)$ is a low pass and $T_2(s)$ a high pass. With this structure, the filter has the capability to merge the information provided by the compass at low frequency with the information given by the rate gyro at higher frequencies. Moreover, the bias introduced in the rate gyro measurements is rejected, since $F_r(0) = 0$.

It is important to remark that in spite of their intrinsic simplicity, complementary filters are especially suited to tackle a number of navigation problems where navigational data is available in complementary frequency ranges. In practice, the structure proposed can be extended to deal with more complex cases that benefit from the core filter structure proposed. For example, complementary filter structures may arise in the design of navigation systems using a stochastic framework.

2.1.2 Kalman-Bucy Filter: Continuous-Time Case

A different state estimation method that is widely studied in literature is the Kalman-Bucy Filter (KBF), see [5] for its derivation. While Complementary Filters are naturally defined in a deterministic framework, the Kalman-Bucy Filters are formulated instead in a stochastic setting, thus requiring the characterization of the process and observation noises. Important to stress that, in spite of their intuitive interpretation, frequency domain designs cannot be extended to multi-input multi-output time varying and nonlinear systems. However, the lessons learned are instrumental in guiding state-space based designs.

The Kalman-Bucy filter is an optimal state observer in a well defined mathematical sense. The objective of the KBF is to optimize the observer gains such that the average of the mean-square prediction error of a variable estimate is minimized. Considered the process model

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + w(t) \\ y(t) &= Cx(t) + v(t) \end{aligned} \tag{2.12}$$

where $x(t)$ is the state of the system, $u(t)$ is a known deterministic input, $y(t)$ is the measurement, $w(t)$ and $v(t)$ are white Gaussian noises with $E[w(t)w(t)^T] = Q_0$ and $E[v(t)v(t)^T] = R_0$. The Kalman-Bucy filter propagates in a recursive way the state estimate $\hat{x}(t)$ that is unbiased, that is,

$$E[x(t) - \hat{x}(t)] = 0 \tag{2.13}$$

and minimizes

$$\underset{x}{E} \left\langle |\hat{x} - x|^2 \right\rangle \tag{2.14}$$

The estimate is obtained by solving the differential equation:

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L_0(y(t) - C\hat{x}(t)), \tag{2.15}$$

and the optimal gain L_0 is given by:

$$L_0 = \Sigma C^T R_o^{-1} \tag{2.16}$$

where the matrix Σ is the symmetric and positive semidefinite solution of the algebraic Riccati equation

$$A\Sigma + \Sigma A^T + Q_o - \Sigma C^T R_o^{-1} C\Sigma = 0. \tag{2.17}$$

If the system model is time-invariant and the noise statistics constant, then under the assumption that the pair(A, C) is observable the positive definite solution of the algebraic Riccati equation exists and is unique, and the closed loop system is asymptotically stable [5].

Lets now reconsider the problem presented in section 2.1.1, where it was required to estimate the heading ψ of a vehicle based on measurements r_m and ψ_m of $r = \dot{\psi}$ and ψ respectively. It was also a requirement to accomplish steady state rejection of the rate gyro bias. The Kalman-Bucy filter can be used to solve the problem in an optimal way. Considering the system dynamics described by 2.12 and recalling 2.9 the Kalman-Bucy filter is designed to minimize the cost function 2.14. The state transition noise is considered to be white Gaussian zero mean noise with co-variance described by a diagonal matrix

$$E\left\{ \begin{bmatrix} w_1(t) \\ w_2(t) \end{bmatrix} \begin{bmatrix} w_1(t) & w_2(t) \end{bmatrix} \right\} = Q_o = \begin{bmatrix} q_{11} & 0 \\ 0 & q_{22} \end{bmatrix} \tag{2.18}$$

and the observation noise

$$E[v(t)^2] = R_o. \tag{2.19}$$

After a few computations, the solution to the Riccati equation 2.17 can be easily derived for this specific problem, after which the optimal observer gains can be obtained as

$$L_o = \frac{\Sigma C^T}{R_o} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \frac{1}{R_o} = \begin{bmatrix} \sqrt{2\sqrt{\frac{q_{22}}{R_o}} + \frac{q_{11}}{R_o}} \\ \sqrt{\frac{q_{22}}{R_o}} \end{bmatrix} \tag{2.20}$$

This result is essential to establish a comparison between the Complementary and the Kalman-Bucy Filter. Here the filter design is uniquely based on the description of the variances of the process and measurement noises. It is important to notice that the observer gains are "proportional" to the state prediction variance and "inversely proportional" to the measurement variance, thus specifying how much the measurements can be trusted. When designing the filter, the variances take the role of "tuning knobs" and therefore the filter design is entirely based on a stochastic setting. On the other hand, in a Complementary Filter the tuning parameter is, as explained in section 2.1.1, the low-pass filter bandwidth. The Complementary Filter is designed in a deterministic framework and its design process is entirely determined by the sensor bandwidths.

2.1.3 Kalman Filter: Discrete-Time Case

The Kalman Filter (KF) belongs to a class of filters known as Gaussian Filters. The Filter dynamics results in a consecutive sequence of prediction and filtering cycles that are derived under the framework of Gaussian probability density functions. Under the Gaussian assumption for the system initial state and all measurement and process noises, the Kalman Filter is the optimal minimum mean-square error state estimator [6]. This filter is the discrete time version of the Kalman-Bucy Filter presented in section 2.1.2.

The Filter uses the system dynamical and observation models

$$\begin{aligned} x_{k+1} &= A_k x_k + B_k u_k + \varepsilon_k \\ y_k &= C_k x_k + \delta_k \end{aligned} \tag{2.21}$$

where $x(k) \in \mathcal{R}^n$, $u(k) \in \mathcal{R}^m$, $y(k) \in \mathcal{R}^r$ are the system state, the control input and the observation respectively. The randomness of state transition, or process noise, is modelled by the zero mean, Gaussian vector ε_k with co-variance R_t. While the vector δ_k describes similar measurement noise with co-variance Q_t

The initial state x_0, is a Gaussian random vector with mean and co variance matrix respectively given

by

$$\begin{aligned} E[x_0] &= \bar{x}_0 \\ E[(x_0 - \bar{x}_0)(x_0 - \bar{x}_0)^T] &= \Sigma_0. \end{aligned} \tag{2.22}$$

Taking the Gaussian viewpoint the Kalman Filter evaluates and propagates the conditional probability density function

$$p(x_k \mid Y_1^k, U_0^{k-1}) \tag{2.23}$$

for every k, where Y_1^k and U_0^{k-1} are the set of all the past measurements and control inputs respectively. That is, the filter propagates the conditional probability density function of the system state, conditioned on the knowledge of the data coming from the measurements and from the control inputs [6]. The goal of the filter is to produce an estimate $\hat{x}_k$ from the probability function. In the Kalman filter, the state estimate $\hat{x}_k$ corresponds to the mean of the probability function [7].

Let $p(x_k \mid Y_1^k, U_0^{k-1}) = \mathcal{N}(\hat{x}_{k|k}, P_{k|k})$ represent the Probability Density Function (PDF) as a Gaussian. The mean of the PDF, $\hat{x}_{k|k}$, is the state estimate and the co-variance matrix $P_{k|k}$ quantifies the uncertainty of the estimate. Due to the linearity of the state transition and observation models, rather than propagating the entire PDF the Kalman Filter only propagates the mean and co-variance of the PDF [6].

The filter starts with an initial estimate of the system state, $\hat{x}_0$, and a correspondent co-variance P_0, and follows a sequence of prediction and update steps at each time instant to improve the state estimate. In the prediction step, the filter uses the most recent state estimate, the control input and the deterministic version of the state transition function 2.21 to compute a new state belief $\hat{x}_{k+1|k}$. The covariance is also updated using the system dynamics, $P_{k+1|k}$. Following the structure in [6] the predict step is defined as

$$\hat{x}_{k+1|k} = A_k \hat{x}_{k|k} + B_k u_k \tag{2.24}$$

$$P_{k+1|k} = A_k P_{k|k} {A_k}^T + R_k. \tag{2.25}$$

The second step transforms the belief $\hat{x}_{k+1|k}$ in the final filter belief $\hat{x}_{k+1|k+1}$ by incorporating the measurement y_{k+1}. In the update step the Kalman gain is also computed, which is "proportional" to the state prediction variance and "inversely proportional" to the measurement variance and so specifying how much the measurement is incorporated in the new state estimate. Again from [6] the update step is defined as

$$K_{k+1} = P_{k+1|k} C_{k+1}^T \left[C_{k+1} P_{k+1|k} C_{k+1}^T + Q_k \right]^{-1} \tag{2.26}$$

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}\left[y_{k+1} - C_{k+1}\hat{x}_{k+1|k}\right] \tag{2.27}$$

$$P_{k+1|k+1} = \left[I - K_{k+1}C_{k+1}\right] P_{k+1|k}. \tag{2.28}$$

Figure 2.4 shows the filter dynamics.

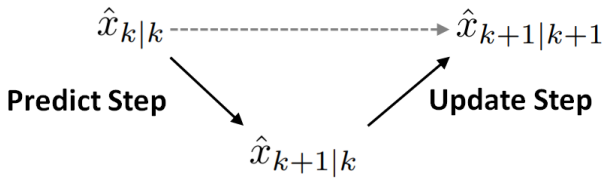


Figure 2.4: Kalman Filter Dynamics

In real applications, it is not usual the case where the system dynamics and the measurement model are linear and thus the Kalman Filter is not suitable for most of the practical examples. To overcome the non-linearities that commonly arise in real systems the Extended Kalman Filter (EKF) is presented.

2.1.4 Extended Kalman Filter: Discrete-Time

When the process or the measurement model of a system is non-linear, the Kalman Filter, as explained in section 2.1.3, no longer propagates Gaussian functions that minimize the mean squared error estimate. The EKF is a non optimal approach to solve the problem using linearized models of the process and observation models around the last state estimates to compute the next state belief [6].

In the extended version of the Kalman Filter, the process model and measurement model, are described by non-linear functions as follows:

$$\begin{aligned} x_{k+1} &= g(x_k, u_k) + \varepsilon_k \\ y_k &= h(x_k) + \delta_k. \end{aligned} \tag{2.29}$$

The Filter produces an approximate of the optimal state belief. This approximation is still a Gaussian function and thus it can be uniquely represented by a mean and co-variance. The filter structure is the same as the Kalman Filter but the predict stage is now defined as

$$\hat{x}_{k+1|k} = g(x_{k|k}, u_{k|k}) \tag{2.30}$$

$$P_{k+1|k} = G_k P_{k|k} G_k^T + R_k. \tag{2.31}$$

where G_k is the Jacobian of the state transition model equation computed at the last state estimate. The update step is similarly defined as

$$K_{k+1} = P_{k+1|k} H_{k+1}^T \left[H_{k+1} P_{k+1|k} H_{k+1}^T + Q_k \right]^{-1} \tag{2.32}$$

$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1} \left[y_{k+1} - h(\hat{x}_{k+1|k}) \right] \tag{2.33}$$

$$P_{k+1|k+1} = \left[I - K_{k+1} H_{k+1} \right] P_{k+1|k} \tag{2.34}$$

where H_k is the Jacobian of the observation model function computed in $\hat{x}_{k+1|k}$.

The EKF is one of the most popular tools for state estimation in robotics [6]. Its applications in navigation systems are wide and several systems using this state estimator have been designed throughout the years, see [8] for a practical example.

2.2 Estimation Theory

When estimating a parameter or a set of parameters, such as state variables, it is important to measure the ”quality” of the estimators being used. This is done by computing a bound that defines the best performance that can be achieved and this bound is then used as a benchmark. In many applications, [9–11], these bounds are also used as cost functions in optimization problems where maximum information about a set of parameters to be estimated has to be extracted.

In order to evaluate the quality of an estimator, one can determine the bias and the variance of the estimate. An estimator is said to be unbiased if the average value of the estimate equals the value of the parameter to be estimated. In the following sections lower bounds for the variances of any unbiased estimator are defined [7].

2.2.1 Single Parameter Estimation

Given a vector of observations, Z, if $\hat{x}(Z)$ is an unbiased estimator of a scalar non-random parameter x, then it is possible to define a lower bound for its variance. This lower bound is called the Cramer-Rao Lower Bound (CRLB), defined as

$$Var[\hat{x}(Z) - x] \geq \left(E\left\{ \left[\frac{\partial ln \Lambda(x)}{\partial x} \right]^2 \right\} \right)^{-1} \tag{2.35}$$

where $\Lambda(x) = p(Z|x)$ is the likelihood function and Z denotes the measurements. Any existing unbiased estimator has its variance lower bounded by the CRLB. If the estimator verifies the bound with an

equality then it is said to be an efficient estimator.

Now a similar bound for the random parameter estimation case is introduced. If x is a random variable and $\hat{x}(Z)$ is an estimator for the variable x then mean-square error of any estimate $\hat{x}(Z)$ satisfies

$$E\{[\hat{x}(Z)-x]^2\} \geq \Big(E\Big\{\Big[\frac{\partial lnP(Z,x)}{\partial x}\Big]^2\Big\}\Big)^{-1}. \tag{2.36}$$

It is important to notice the difference between the two bounds. While the computation of the first one involves the conditioned probability $\Lambda(x) = p(Z|x)$, the second one involves the joint probability $P(Z,x)$.

2.2.2 Multiple Parameter Estimation

In many practical situations the parameter to be estimated is multidimensional. An example is the case of a navigation system where one wishes to estimate the velocity and position of an autonomous vehicle. In this and similar situations the bounds previously defined must be revised.

If x is a multidimensional non-random parameter to be estimated with any unbiased estimator $\hat{x}(Z)$ then

$$Var[\hat{x}(Z)-x] \geq J^{-1} \tag{2.37}$$

where J is the FIM defined by

$$\begin{aligned} J &\triangleq -E\Big(\{\bigtriangledown_x\left[ln\Lambda(x)\right]\}\{\bigtriangledown_x\left[ln\Lambda(x)\right]\}^T\Big) \\ &= -E\Big[\bigtriangledown_x\left(\{\bigtriangledown_x\left[ln\Lambda(x)\right]\}^T\right)\Big] \end{aligned} \tag{2.38}$$

and $\Lambda(x) = p(Z|x)$ is the likelihood function. The Fisher Information Matrix is commonly referred as measure of the quantity of information about the parameter to be estimated, x, in the vector observations, Z, [12].

As in the scalar case it is also possible to define a lower bound for the variance of any unbiased estimator $\hat{x}(Z)$ of a multidimensional random variable x. This bound is derived from the bound for non-random parameters estimation but here the Fisher Information Matrix is defined as

$$J \triangleq J_D + J_P \tag{2.39}$$

where the matrix J_D is the matrix defined in 2.38 and represents the information obtained from the data. The matrix J_P represents the priori information and is defined as

$$\begin{aligned} J_P &\triangleq -E\Big(\{\bigtriangledown_x\left[lnP(x)\right]\}\{\bigtriangledown_x\left[lnP(x)\right]\}^T\Big) \\ &= -E\Big[\bigtriangledown_x\left(\{\bigtriangledown_x\left[lnP(x)\right]\}^T\right)\Big] \end{aligned} \tag{2.40}$$

where $P(x)$ is the priori probability density function.

2.2.3 Summary

It must be emphasized that all the previous bounds are defined as functions of the measurement model and the parameter to be estimated. This fact may seem to undermine any practical utility of these results, since if one knows the parameter, there is no need to estimate it. Nonetheless, these results are of great interest, since they provide a relation between optimal measurement setups and the parameters of interest. In cases where the estimation is performed with an iterative process every time an estimation is computed it can be used to determine a new optimal measurement setup. Using adequate estimation algorithms by minimizing the CRLB one can expect to minimize the actual estimation error.

2.3 Optimal Control Problem

Besides the problem of determining the best performance achievable for the target position estimation problem, which is given by the CRLB defined in the previous section. This work also addresses the problem of computing optimal motions for the trackers. To compute these motions a MPC strategy is proposed. In this context an OCP arise to determine the optimal controls for the trackers to maximize the information about the positions of the targets. In this section a general OCP is presented and the main methods to solve it are discussed.

A continuous OCP is an infinite dimensional optimization problem [1]. A motion planning problem for a single vehicle, like the one that will be addressed in this work, can be converted in the form of an optimal control problem of the form (see [13])

$$\begin{aligned} \min_{x(t);u(t)} \quad & \varphi(x(\cdot),u(\cdot)) \\ \text{subject to} \quad & x(t_0) = x_0, && \text{(fixed initial value)} \\ & \dot{x}(t) = f(x(t),u(t)), \quad t \in [t_0,t_f]\,, && \text{(ODE model)} \\ & 0 \leq c(x(t),u(t)), \quad t \in [t_0,t_f]\,, && \text{(path constraints)} \\ & r(x(t_f)) = 0, && \text{(fixed final value)} \end{aligned} \tag{2.41}$$

in which a dynamic process, $x(t)$, is determined for a continuous period of time $[t_0,t_f]$, described by an Ordinary Differential Equation (ODE) given by $\dot{x}(t) = f(x(t),u(t))$, affected by a control $u(t)$, such that a cost $\varphi(\cdot)$ is minimized and satisfying the path constraints $c(x(t),u(t))$, and the initial and terminal constraints, $x(t_0) = x_0$ and $r(x(t_f)) = 0$ respectively.

To solve an OCP like the one in equation (2.41) multiple methods can be found in the literature. Generally, there are three different approaches to solve an OCP, [13]: Dynamic Programming, Indirect Methods and Direct Methods. Direct Methods are also divided in two categories [1]: Collocation Methods and Shooting Methods. To solve the OCP that arises in the context of the optimal motion planning in this

work, Shooting Methods will be used since they are the most widespread and successfully used in the literature.

2.3.1 Direct Methods

Direct methods are based on a discretization of the infinite dimensional OCP, like the one in equation (2.41), into a finite dimensional Nonlinear Programming (NLP) problem. Then, the NLP is solved by a numerical optimization solver (see [14]).

2.3.2 Direct Single Shooting

The direct shooting method starts by descritizing the control input. In the time horizon of the OCP, $[t_0, t_f]$ a grid of N points is defined and the control input is taken as a piece-wise constant function

$$u(t) = q_i \quad \forall\, t \in [t_i, t_{i+1}]\,, \tag{2.42}$$

where $q_i \in q$ has the dimensions of the control input, $[t_i, t_{i+1}]$ denotes the size of the discretization step and $q = (q_0, ..., q_{N-1})$. With this notation the descritized control input can be denoted as $u(t; q)$.

The next step in the direct shooting method is to use the state trajectory as a dependent value of the controls $u(t; q)$, and using an ODE solver, solve the following Initial Value Problem (IVP)

$$\begin{aligned} &x(t_0) = x_0, \\ &\dot{x}(t) = f(x(t), u(t; q)), \quad t \in [t_0, T] \end{aligned}\ , \tag{2.43}$$

to obtain the evolution of $x(t)$ in the time interval $[t_0, t_f]$, denoted by $x(t; q)$. If the path constraints are also discretized, that is, for example by requiring $c(x(t), u(t)) \geq 0$ only at the grid points t_i, then the OPC in equation (2.41) is transformed into the following finite dimensional NLP problem:

$$\begin{aligned} \min_{q} \quad & \varphi\left(x(t_i; q), u(t_i; q)\right) \\ \text{subject to} \quad & x(t_0) = x_0, \\ & \dot{x}(t) = f(x(t_i; q), u(t_i; q)), \quad i = 0, ..., N, \\ & c(x(t_i; q), u(t_i; q)) \geq 0, \qquad i = 0, ..., N, \\ & r(x(t_f; q)) = 0. \end{aligned} \tag{2.44}$$

This NLP is solved by a finite dimensional numerical optimization solver (see [14]).

The direct shooting method has some disadvantages. For example, it is impossible to use previous knowledge to initialize the state trajectory since the only initialization parameters in the NLP are x_0 and the control inputs $u(t; q)$. Also, the existence of solution is highly dependent on the correct initialization of these parameters. However, due to its simplicity, and its straightforward implementation, the single shooting approach is very often used in engineering applications.

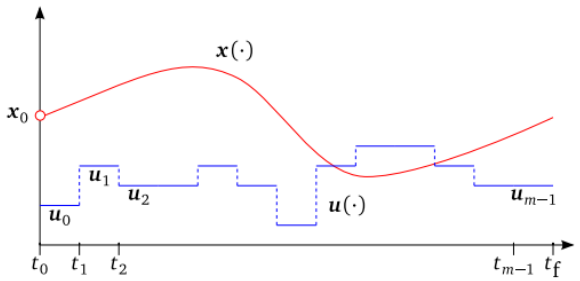


Figure 2.5: Illustration of the single shooting method applied to an OPC [1]

.

2.3.3 Direct Multiple Shooting

The direct multiple shooting is often referred as an hybrid method as it has characteristics of both direct and collocation methods. On one side it uses state trajectory discretization as the collocation methods, but on the other hand it still relies on solving IVP's as the direct single shooting method does.

With multiple shooting method the control input $u(t)$ is still discretized as in equation (2.42). However, instead of solving the ODE for the entire interval $[t_0, t_f]$ as in single shooting, in the multiple shooting method the ODE is solved for each interval $[t_i, t_{i+1}]$ independently. This is the main difference between the two methods, and is this characteristic that gives this method the name multiple shooting. Instead of solving the ODE for the entire time interval of the OCP ("single shot"), this method rather solves the ODE in the N grid intervals independently ("multiple shots").

The IVP in equation (2.43) can be re-written for the multiple shooting method as

$$\begin{aligned} x(t_i) &= s_i, \\ \dot{x}(t_i) &= f(x_i(t), q_i), \quad t \in [t_i, t_{i+1}] \end{aligned} \tag{2.45}$$

where s_i is an initial value of the state trajectory in the time interval $[t_i, t_{i+1}]$. By solving these IVP's N state trajectories pieces $x(t; s_i, q_i)$ are obtained. Simultaneously the cost function on these intervals are also computed $l_i(t_i, s_i, q_i)$, for all $i = 0, ..., N-1$

In order to obtain physical feasible solutions, it is necessary to impose continuity constrains on the solution $x(t)$, that is $s_{i+1} = x_i(t_{i+1}; t_i, s_i, q_i)$. Here $x_i(t_{i+1}; t_i, s_i, q_i)$ denotes the final value $x(t_{i+1})$ of the IVP when starting with an initial condition $x(t_i) = s_i$ and applying the control q_i.

The discretized optimal control problem resulting from application of the direct multiple shooting discretization is presented in the following equation

$$\begin{aligned} \min_{q} \quad & \textstyle\sum_{i=0}^{N-1} l_i(t_i, s_i, q_i) \\ \text{subject to} \quad & x(t_0) = s_0, \\ & s_{i+1} = x_i(t_{i+1}; t_i, s_i, q_i), \quad i = 0, ..., N-1, \\ & h(s_i, q_i) \geq 0, \qquad\qquad\quad i = 0, ..., N, \\ & r(s_N) = 0. \end{aligned} \tag{2.46}$$

Again, this NLP is solved by a finite dimensional numerical optimization solver [14].

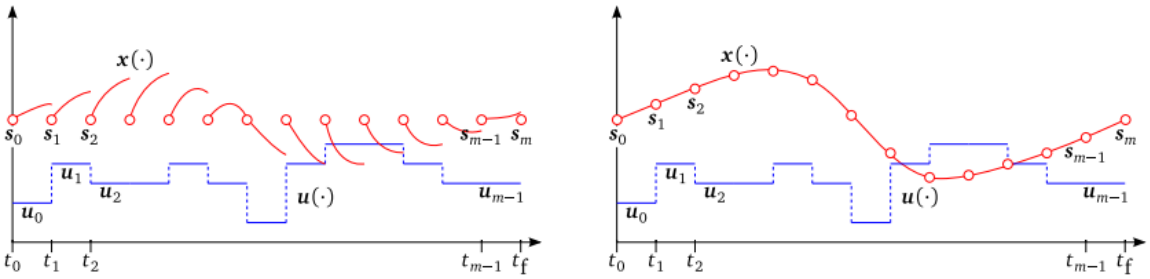


Figure 2.6: Illustration of the multiple shooting method applied to an OPC. On the left figure there are no continuity constraints thus the solution does not have a physical significance. On the right with the continuity constrains fulfilled a correct solution was found. [1]

.

Comparing with the single shooting method, one advantage of the multiple shooting method is that the knowledge of the state trajectory can be used in the initialization variables. Stability of the solution process and faster convergence are also improvements comparing with the single shooting [1].

2.4 Optimal Sensor Placement

Over the years, the concepts of Estimation Theory have been used to solve problems of optimal sensor placement. A first approach on optimal sensor placement by in Porat&Nehorai in [10] tackles the problem of estimating a Vapor-Emitting source with a single moving sensor. The idea is to use a single moving sensor to perform a task that, in principle, would need an array of stationary sensors. After a measurement is taken, the optimal criterion used in this paper is to minimize as much as possible the expected location error. This is done by computing the gradient of the CRLB on the estimated position with respect to the sensor's coordinates and then moving the sensor in the opposite direction of the gradient. For the moving-sensor algorithm it is assumed that new measurements are available every ΔT second. Then, every time a new measurement comes, an estimation of the source position is performed and the CRLB is computed with the current estimate, $\hat{x}$. Considering (x, y, z) the location of the sensor at the $n-th$ measurement, let S_{n+1} be the set of reachable points by the sensor at the $(n+1)T$ instant. The sensor is then moved to the new location in S_{n+1} so as to minimize the CRLB. The new location can be chosen by searching over a grid of points in S_{n+1}. However, the authors propose an alternative to this procedure that is more efficient computationally. This alternative is to compute the gradient of the CRLB after $n+1$ measurement with respect to the prospective new sensor position. Then move the sensor to the opposite direction of the gradient.

Also, in this paper, the authors give a practical example where the motion of a sensor when estimating the position of a static source is shown. It is shown that the path travelled by the sensor consists in

circling around the source with decreasing distance. This behaviour is very intuitive since the circling movement provides spatial diversity and by decreasing the distance to the target the sensors become less sensitive to noise. A limitation of this work is that it does not address the problem of moving targets or the problem of multiple moving sensors and their movement coordination.

A different approach was adopted in the work by Martinez&Bullo [11], where the authors investigate the design of distributed motion coordination algorithms in order to maximize the information gathered by a network of sensors in static and dynamic target-tracking scenarios. To this end, the authors define a cost function that encodes a "sensitivity performance measure" (the determinant of the Fisher Information Matrix) and an algorithm is designed to maximize it. The localization of static targets is solved as a non-random parameter estimation problem while in the case of dynamic target-tracking scenario the targets are random parameters evolving under a stochastic difference equation.

In this work it is considered that the measurements taken are "Range-Measurements". Under the assumption of Gaussian noise measurements with diagonal correlation, the authors derive the expressions for the determinant of the Fisher Information Matrix for models in non-random static scenarios, for 2 and 3-dimensional state spaces. Considering that the sensors measure distances to the target, it is shown that the optimal sensor configuration is the one where the sensors are uniformly placed in a circular way around the target. This result is aligned with [10], and again the idea is to provide spatial diversity when placing the sensors.

Later, a target tracking scenario is considered where the sensors can move in a convex domain around the area in which the target is known to lie. Discrete-time decentralized control laws, relying only on local information, are defined to achieve the uniform configuration around the target estimate.

Following the work done by Martinez&Bullo, a specific approach for the case of tracking an underwater robot with multiple surface vehicles using acoustic range measurements is investigated by Moreno&Pascoal in [9]. Here, in contrast with the work done in [11], the authors characterize explicitly, for both 2D and 3D cases, the optimal sensor positioning with respect of their distances to the target to be located, as shown in figure 2.7.

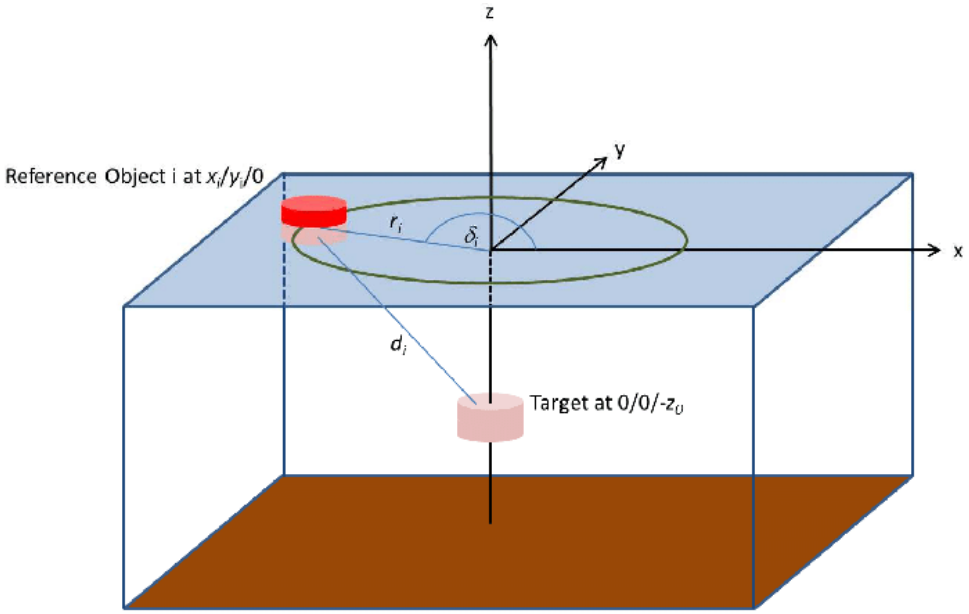


Figure 2.7: Optimal Sensor Placement

The authors show that for the specific case of an underwater target and surface sensors the optimal configuration (the configuration that carries more information about the target position) is the one where the sensors are equally distributed in a circumference centered above the target position. The results are validated with a set of Monte Carlo Simulations, and provide a contribution to the area of Optimal Sensor Placement for the localization of underwater targets in realistic marine scenarios.

2.5 Range-Based Target Tracking Previous Work

The range-based target tracking and estimation problems have been extensively studied over the last years. This problem has many similarities to the classical problem of range-based vehicle positioning, and a lot of knowledge from that problem has been transposed to the target tracking problem. For instance, in [15] both the problem of range-based marine vehicle positioning and the problem of target estimation are addressed. Vehicle positioning is the dual problem of the target estimation problem. The goal of a positioning system is to estimate the position of a vehicle from a sequence of range measurements to a set of fixed or moving beacons whose position are known. On the contrary, in the target estimation problem a vehicle whose position is known has to estimate the position of an unknown target using range measurements. However, in both problems a vehicle should find optimal trajectories, either for self-positioning or target localization. In [15], vehicle trajectories for multiple vehicle positioning are computed by maximizing the determinant of a defined parametric FIM and for experimental purposes the dual problem of target estimation is also considered. The authors developed an integrated motion

planning, control, and estimation algorithm for range-based single target localization and tested it with practical experiments in a marine scenario.

The problem of cooperative range-based target localization is explicitly addressed in the problem formulation in [16] and [17]. The single tracker, single target problem is extended in these papers for the case where a tracker works in cooperation with another vehicle that can also measure ranges to the target and shares the information with the tracker. It is shown in simulations that there are advantages in using two trackers instead of just one.

Solutions for the problem of target localization and tracking resorting to MPC strategies have been previously studied [18–20]. In [18] the authors propose a feedback law that drives a follower vehicle to a target vehicle using an online estimate of the target states. Simulations show that solving the target tracking problem independently from the target estimate easily leads to unsatisfactory results, where the follower vehicle is driven to weakly observable trajectories and as a result the states of the target cannot be estimated. Therefore the proposed strategy in [18] embeds in the optimisation problem of the MPC scheme an index of observability of the target states based on the observability matrix. For a single follower taking bearing measurements to a single target, the proposed strategy is proven to be able to drive the follower to highly observable trajectories and as a result to successfully estimate the target states. Based on this work, in [19] the authors propose an Nonlinear Model Predictive Control (NMPC) approach for an AUV to track and estimate a moving target using range-only measurements. Again, an index of observability of the target states based on the observability matrix was utilized in the optimisation problem of the NMPC to steer the vehicle to highly observable trajectories. The proposed strategy was tested in simulation for a 3D case with a single AUV and a single target moving in the surface.

The problem of multiple target localization and pursuit using range measurements from multiple trackers is explicitly addressed in [20]. Here, the proposed MPC strategy is rooted on Estimation Theory setting and requires the computation of a Bayesian Fisher Information Matrix as a measure to quantify the range information for the estimation of the target's states. The objective of the strategy is to generate optimal motion for the trackers maximizing the predicted range information about the target's states. The Bayesian FIM reveals itself to be more appropriate than the parametric FIM found previously in the literature [16] because it allows capturing the case where the target moves, as described by a possible nonlinear system, with uncertainty in the initial states and measurement noise. Its computation is also simpler and more transparent than the observability matrix in [18, 19]. Also in [20], using FIM relative geometries between targets and trackers are derived for which the range information computed by the determinant of the FIM is maximum.

In [21] a Moving Path Following [22] approach to the target estimation and tracking using range-only measurements was investigated. While in MPC based strategies trajectories are planned over a finite horizon, in this setting a geometric path that guarantees that the target states remain highly observable

is chosen beforehand. An EKF estimator was used to estimate the target position and velocity and the proposed strategy was tested for a single target single tracker scenario in 2D.

3

Target Tracking Problem

The developed target tracking strategy can handle target tracking problems that involve the following three elements: (i) Trackers for which the motion trajectory has to be computed in order to maximize the information about the position of the targets; (ii) Targets that can be static or moving whose positions needs to be estimated; (iii) A measurement model that relates the position of the targets with an acoustic range measurement. Each of these elements contribute to the optimization problem of determining the trackers optimal trajectories. The following subsections each analyse a specific element of this target tracking problem.

3.1 Tracker Model

The tracker motion model describes how the position of each tracker evolves in time. In this work, each tracker will be an Autonomous Surface Vehicle (ASV). Hereafter consider an arbitrary number of $p \in \mathbb{N}$ trackers and, for a given p, a set $\mathcal{S} = \{1, .., p\}$.

In what follows consider a global North-East-Down (NED) [23] inertial coordinate frame $\{\mathcal{I}\} = \{x_{\mathcal{I}}, y_{\mathcal{I}}, z_{\mathcal{I}}\}$ and, for a given $i \in \mathcal{S}$, a body-fixed coordinate frame $\{\mathcal{B}\}^{[i]} = \{x_{\mathcal{B}}^{[i]}, y_{\mathcal{B}}^{[i]}, z_{\mathcal{B}}^{[i]}\}$, with origin at the i^{th} vehicle's center of mass.

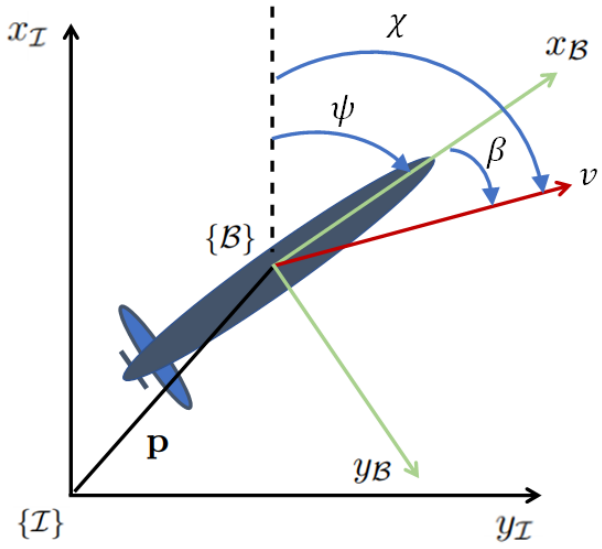


Figure 3.1: Tracker dynamics representation

Here, each tracker will be considered a rigid body that operates at the surface. This is the same to say that the trackers operate at a known constant depth in $\{\mathcal{I}\}$, that is $z^{[i]}(t) = \bar{z}^{[i]}$, for all $t > 0$ and $i \in \mathcal{S}$, where $\bar{z}^{[i]} = 0$. In this model it is assumed that the tracker can move with a certain linear velocity and can define its orientation in a 2-dimensional space. It is also assumed that sideslip angle is so small that can be neglected. With this assumptions the kinematics of the i^{th} vehicle, $i \in \mathcal{S}$, is described by the

following model in $\mathbb{R}^2$,

$$\begin{matrix} \dot{\mathbf{p}}^{[i]} = v^{[i]}[cos(\chi^{[i]}), sin(\chi^{[i]})]^T \\ \dot{\chi}^{[i]} = r^{[i]} \end{matrix}, \tag{3.1}$$

where $\mathbf{p}^{[i]} = [x^{[i]}, y^{[i]}]^T \in \mathbb{R}^2$ is the horizontal position of the tracker in $\{\mathcal{I}\}$, $v^{[i]}$ is the vehicle speed, $\chi^{[i]}$ is the course angle and $r^{[i]}$ is the course angle rate. Note that from the tracker dynamics represented in Figure 3.1, when the sideslip angle, represented by β is zero, the course angle equals the heading (yaw) angle, that is $\chi^{[i]} = \psi^{[i]}$. Also, in that case the course angle rate equals the heading angle rate.

In the presence of external disturbances, such as ocean currents, the model (3.1) is still valid. In that scenario, the vehicle velocity $v^{[i]}$ represents instead the magnitude of the total velocity vector of the vehicle in the inertial frame $\{\mathcal{I}\}$ and course angle $\chi^{[i]}$ is defined with respect to the relative velocity of the vehicle with respect to the water, that is, the total inertial velocity minus the current velocity [23].

The goal is to find optimal speeds and course angle rates which drive the vehicle to trajectories providing the maximum information about the target position. In order to obtain more realistic results and smoother velocities a model that has as inputs accelerations instead of velocities is introduced. Therefore consider $\mathbf{v}^{[i]} = [a_v^{[i]}, a_r^{[i]}]^T \in \mathbb{R}^2$ as the input vector and $\mathbf{z}^{[i]} = [x^{[i]}, y^{[i]}, \chi^{[i]}, v^{[i]}, r^{[i]}]^T \in \mathbb{R}^5$ as the state vector. The motion model in (3.1) can be rewritten as

$$\begin{cases} \dot{\mathbf{z}}^{[i]} = \mathbf{g}(\mathbf{z}^{[i]}, \mathbf{v}^{[i]}) \\ \mathbf{p}^{[i]} = C\mathbf{z}^{[i]} \end{cases} \tag{3.2}$$

where $\mathbf{g} : \mathbb{R}^5 \times \mathbb{R}^2 \mapsto \mathbb{R}^5$ is given by,

$$\mathbf{g}(\mathbf{z}^{[i]}, \mathbf{v}^{[i]}) = \begin{bmatrix} v^{[i]} \cdot cos(\chi^{[i]}) \\ v^{[i]} \cdot sin(\chi^{[i]}) \\ r^{[i]} \\ a_v^{[i]} \\ a_r^{[i]} \end{bmatrix}. \tag{3.3}$$

and $C = \begin{bmatrix} I_2 & 0_{2\times 3} \end{bmatrix} \in \mathbb{R}^{2\times 3}$. Here I_2 is a two by two identity matrix and $0_{2\times 3}$ a zero matrix with size 2×3.

For computational implementation purposes the model in (3.2) has to be discretized as

$$\begin{cases} \dot{\mathbf{z}}_{k+1}^{[i]} = \mathbf{g}_d(\mathbf{z}_k^{[i]}, \mathbf{v}_k^{[i]}) \\ \mathbf{p}_k^{[i]} = C\mathbf{z}_k^{[i]} \end{cases}, \tag{3.4}$$

where $g_d(\cdot)$ is a non-linear discrete version of the function $g(\cdot)$ in (3.2) which depends on the discretization process.

Besides the motion model, and due to the physical limitations of each vehicle, the tracker is also

subjected to state and input constraints. That is, the vehicle admits control inputs in the set

$$\mathcal{V}^{[i]} := \{(a_v^{[i]}, a_r^{[i]}) : \quad a_{v_{min}}^{[i]} \leqslant a_v^{[i]} \leqslant a_{v_{max}}^{[i]}, \quad a_{r_{min}}^{[i]} \leqslant a_r^{[i]} \leqslant a_{r_{max}}^{[i]}\} \tag{3.5}$$

and that the states are constrained to the set

$$\begin{aligned} \mathcal{Z}^{[i]} := \{(x^{[i]}, y^{[i]}, \chi^{[i]}, v^{[i]}, r^{[i]}) : \\ x_{min}^{[i]} \leqslant x^{[i]} \leqslant x_{max}^{[i]}, \\ y_{min}^{[i]} \leqslant y^{[i]} \leqslant y_{max}^{[i]}, \\ \chi_{min}^{[i]} \leqslant \chi^{[i]} \leqslant \chi_{max}^{[i]}, \\ 0 \leqslant v^{[i]} \leqslant v_{max}^{[i]}, \\ r_{min}^{[i]} \leqslant r^{[i]} \leqslant r_{max}^{[i]}\}. \end{aligned} \tag{3.6}$$

3.2 Target Model

The target motion model describes how the position of each target evolves with time. In what follows the targets will be refereed to as Autonomous Underwater Vehicles (AUV's), but they can also be another motion system, e.g. a diver or a fish [24]. As for the trackers, an arbitrary number of $q \in \mathbb{N}$ targets is considered and for a given q a set $\mathcal{S}_{\mathcal{T}} = \{1, .., q\}$ is defined.

The primary objective of the target tracking problem is to estimate the trajectories of the target states over time. Although each target might have a shape and orientation, in order to keep the target tracking problem as general as possible, the target will be represented in the motion model as a mass point without shape or orientation. The mass point represents the vehicle's centre of mass that can move in a 3-dimensional space.

Each target will be assumed to operate at a known constant but possible different depth in $\{\mathcal{I}\}$, that is $z^{[j]}(t) = \bar{z}_T^{[j]}$, for all $t > 0$ and $j \in \mathcal{S}_T$. Let $\mathbf{q}_k^{[j]} = [x_{T;k}^{[j]}, y_{T;k}^{[j]}]^T \in \mathbb{R}^2$ be the position of the j^{th} target at the k time instant in the horizontal plane $x_{\mathcal{I}} - y_{\mathcal{I}}$. Consider that the trackers are carried with the task of estimation the position of the targets. The dynamics of the j^{th} target, with $j \in \mathcal{S}_T$, is defined by the following discrete motion model

$$\mathbf{x}_{k+1}^{[j]} = \mathbf{f}(\mathbf{x}_k^{[j]}, \mathbf{u}_k^{[j]}) \tag{3.7}$$

where $\mathbf{x}_k^{[j]} \in \mathbb{R}^n$, with $n \geq 2$, is the target state and $\mathbf{u}_k^{[j]} \in \mathbb{R}^2$ is the target input vector both at the discrete time instant k. The function $\mathbf{f}(\cdot)$ describes the target dynamical model.

From this point on, two different scenarios will be considered.

3.2.1 Scenario A

In scenario A it is assumed that the velocity vector of each target is known to all the trackers in the mission. Situations with this particularity may occur for example in path following missions [25] [26]. In this type of mission, an underwater vehicle performs a pre-defined mission, so it is feasible that a tracker will possess information about the pre-defined trajectories for the target velocity vector. With this assumption, in the horizontal plane $x_{\mathcal{I}} - y_{\mathcal{I}}$, the dynamics of the j^{th} the target, with $j \in \mathcal{S}_T$, is defined by the following discrete motion model

$$\mathbf{x}_{k+1}^{[j]} = A\mathbf{x}_k^{[j]} + B\mathbf{u}_k^{[j]} \tag{3.8}$$

where $\mathbf{x}_k^{[j]} = \mathbf{q}_k^{[j]} \in \mathbb{R}^2$, $\mathbf{u}_k^{[j]} \in \mathbb{R}^2$ is the velocity vector at the discrete time instant k, $A = I_2$, $B = T_s I_2$, I_2 is an identity matrix of size 2 and T_s the sampling interval.

Additionally, assume that at each time instant k the target states, $\mathbf{x}_k$, are described by a Gaussian distribution and that its prior information is given by

$$\mathbf{x}_0^{[j]} \sim \mathcal{N}(\boldsymbol{\mu}_{A,0}^{[j]}, P_{A,0}^{[j]}) \tag{3.9}$$

where $\boldsymbol{\mu}_{A,0}^{[j]} \in \mathbb{R}^2$ is the expected value of the prior information distribution and $P_{A,0}^{[j]} \in \mathbb{R}^{2\times 2}$ is the covariance matrix, where the subscript A denotes the association with scenario A.

3.2.2 Scenario B

In scenario B, the velocity vector of each target is assumed to be unknown to all the trackers. With this assumption, the trackers are carried with the additional mission of estimating the velocities of all the targets. An additional and final assumption is made in this scenario. Assume that the target's velocity vector changes slowly, so that it can be considered constant over a short period of observation, that is $\dot{\mathbf{q}}^{[j]} = [0, 0]$ for all $j \in \mathcal{S}_T$.

In the horizontal plane $x_{\mathcal{I}} - y_{\mathcal{I}}$, the dynamics of the j^{th} target, with $j \in \mathcal{S}_T$, is defined by the following discrete motion model

$$\mathbf{x}_{k+1}^{[j]} = A_B\mathbf{x}_k^{[j]} \tag{3.10}$$

with $\mathbf{x}_k^{[j]} = [\mathbf{q}_k^{[j]}; \dot{\mathbf{q}}_k^{[j]}] \in \mathbb{R}^4$ and

$$A_B = \begin{bmatrix} I_2 & T_s I_2 \\ 0_{2\times 2} & I_2 \end{bmatrix} \in \mathbb{R}^{4\times 4}, \tag{3.11}$$

where I_2 is an identity matrix of size 2, T_S is the sampling interval and $0_{2\times 2}$ is matrix full of zeros of size two by two.

As in scenario A, assume that at each time instant k the target states, $\mathbf{x}_k$, are described by a Gaussian distribution and that its prior information is given by

$$\mathbf{x}_0^{[j]} \sim \mathcal{N}(\boldsymbol{\mu}_{B,0}^{[j]}, P_{B,0}^{[j]}) \tag{3.12}$$

where $\boldsymbol{\mu}_{B,0}^{[j]} \in \mathbb{R}^4$ is the expected value of the prior information distribution, $P_{B,0}^{[j]} \in \mathbb{R}^{4\times4}$ is the covariance matrix, and the subscript B denotes the association with scenario B.

3.3 Measurement Model

The measurement model relates the target states $\boldsymbol{x}_k^{[j]}$ with measurements of the ranges to the trackers. It is assumed that the trackers will use acoustic range measurements to estimate the targets position. At each time instant k, let $d_k^{[ij]}$ be the real distance between the i^{th} tracker and the j^{th} target, defined by

$$d_k^{[ij]} = \sqrt{\left\|\mathbf{p}_k^{[i]} - \mathbf{q}_k^{[j]}\right\|^2 + (\bar{z}^{[i]} - \bar{z}_T^{[j]})^2}, \tag{3.13}$$

where $\|\cdot\|$ denotes the Euclidean norm, $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$

Let $y_k^{[ij]}$ denote the measurement of the actual range between the i^{th} tracker and the j^{th} target corrupted by additive noise $\eta_k^{[ij]}$, at the discrete time instant k. In what follows, $\eta_k^{[ij]}$ is taken to be a zero mean white Gaussian process $\mathcal{N}(0, \Sigma)$ with variance σ.

With this notation, the measurement model is given by

$$y_k^{[ij]} = d_k^{[ij]} + \eta_k^{[ij]}. \tag{3.14}$$

Here it is assumed that the measurement noise model is distance independent. This assumption is in line with the common assumptions found in the literature for both theoretical work [27] and system implementation [15]. In practice, the range range measurements can only be taken up to a certain distance d_{max}, that depends on the measuring devices used and the respective mission [9]. To cope with this limitation it is assumed that all measurements are taken within this distance, that is $d_k^{[ij]} < d_{max}$, for every $k \in \mathbb{N}$, $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$.

3.4 Problem Formulation

The work proposed in this thesis is to study and design systems for range-based underwater target localization and tracking using single and multiple autonomous surface vehicles. The focus of the work will be the generation of motions for the surface vehicles in order to gather the maximum range information

about the target states.

The task of generating motions for the surface vehicles consists in determining the controls in (3.5), for trackers with the motion model described in (3.2), which maximize some cost that transcribes the range information about the target states in (3.8) and (3.10) for scenario A and B respectively. The information about the target states will be quantified resorting to a theoretical standpoint, using the definition and properties of the Fisher Information Matrix.

4

Fisher Information Matrix

In the previous chapter the target tracking problem to be addressed in this work was formulated. As mentioned before, Estimation Theory concepts, namely the Fisher Information Matrix, will be used to quantify the amount of information that the trackers can get about the targets position from the range measurements.

There are two main contributions in this chapter. First, we describe how to derive the Fisher Information Matrix in the context of the target tracking problem. Second, the concept of Fisher Information Matrix derived for this problem is used to determine optimal geometries for the trackers that maximize, theoretically, the information available about the targets positions given the range measurements.

4.1 Nonlinear Filtering Problem

Filtering is the estimation of the (current) state of a dynamic system [12]. Consider the nonlinear filtering problem

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}_k(\mathbf{x}_k) + \omega_k \\ \mathbf{z}_k &= \mathbf{h}_k(\mathbf{x}_k) + \eta_k \end{aligned} \tag{4.1}$$

where $\mathbf{x}_k \in \mathbb{R}^n$ is the n-dimensional system state at time k, $\mathbf{z}_k \in \mathbb{R}^m$ is a vector of measured data and $h_k(\cdot)$ and f_k and h_k are non-linear functions. The noises ω_k and η_k are Gaussian with zero mean and covariance matrices Q_k and R_k respectively.

Let $\hat{\mathbf{x}}_k$ be an estimate of $\mathbf{x}_k$ based on a set of m observations and the prior knowledge of initial probability density function $p(x_0)$. According to [7], the covariance matrix of $\hat{\mathbf{x}}_k$ denoted by P_k, is lower bounded by

$$\mathbb{E}\left\{[\hat{\mathbf{x}}_k - \mathbf{x}_k][\hat{\mathbf{x}}_k - \mathbf{x}_k]^T\right\} \geq L_k^{-1} \tag{4.2}$$

where L_k is the Fisher Information Matix and its inverse the Cramér-Rao Lower Bound and $\mathbb{E}$ corresponds to the expected value operator. According to [28], the Fisher Information Matix, L_k, can be computed recursively by the following formula

$$L_{k+1} = D_k^{22} - D_k^{21}(J_k + D_k^{11})^{-1} D_k^{12} \tag{4.3}$$

where

$$D_k^{11} = \mathbb{E}\{[\nabla_{x_k} \mathbf{f}_k^T(\mathbf{x}_k)] Q^{-1} [\nabla_{x_k} \mathbf{f}_k^T(\mathbf{x}_k)]^T)\} \tag{4.4}$$

$$D_k^{21} = -\mathbb{E}\{\nabla_{x_k} \mathbf{f}_k^T(\mathbf{x}_k)\} Q^{-1} = [D_k^{12}]^T \tag{4.5}$$

$$D_k^{22} = Q^{-1} + \Omega_{k+1} \tag{4.6}$$

with

$$\Omega_{k+1} = \mathbb{E}\{[\nabla_{x_{k+1}} \mathbf{h}_k^T(\mathbf{x}_{k+1})] R^{-1} [\nabla_{x_{k+1}} \mathbf{h}_k^T(\mathbf{x}_{k+1})]^T\}, \tag{4.7}$$

where for a given $\mathbf{x} = [x_1, ..., x_n]^T \in \mathbb{R}^n$, $\nabla_{\mathbf{x}} \triangleq [\frac{\partial}{\partial x_1}, ..., \frac{\partial}{\partial x_n}]$. The initial Fisher Information Matrix L_0 can be calculated from the *a priori* probability function $p(\mathbf{x}_0)$

$$L_0 = \mathbb{E}\left\{[\nabla_{\mathbf{x}_0} \log(p(\mathbf{x}_0))][\nabla_{\mathbf{x}_0} \log(p(\mathbf{x}_0))]^T\right\}. \tag{4.8}$$

4.2 FIM in the Context of Target Localization

Using the notation introduced in the previous section, in the context of range based underwater target localization, $\mathbf{x}_k$ is target state at time k, $\mathbf{z}_k$ is the measured range at time k and the function $\mathbf{h}_k(\cdot)$ correspond to the distances between the target and the trackers. Since the two different target motion models introduced in section 3.2 are linear, a special case of the filtering problem in (4.1) where the state transition equation is linear is considered in this section. When the state transition equation is linear the filtering problem in (4.1) is given by,

$$\begin{aligned} \mathbf{x}_{k+1} &= A\mathbf{x}_k + B\mathbf{u}_k + \omega_k \\ \mathbf{z}_k &= \mathbf{h}_k(\mathbf{x}_k, \boldsymbol{\phi}_k) + \eta_k \end{aligned} \tag{4.9}$$

where ϕ_k are the known deterministic positions of the trackers. Considering the filtering problem in (4.9), the equations (4.4) and (4.5) can be rewritten as

$$D_{11} = AQ^{-1}A^T \tag{4.10}$$

$$D_{21} = -AQ^{-1} = D_{12}. \tag{4.11}$$

With these equations (4.3), can be simplified as

$$\begin{aligned} L_{k+1} &= Q^{-1} + \Omega_{k+1} - Q^{-1}A(L_k + A^TQ^{-1}A)^{-1}A^TQ^{-1} \\ &= (Q + AL_k^{-1}A^T)^{-1} + \Omega_{k+1}. \end{aligned} \tag{4.12}$$

Considering no process noise, i.e Q is a null matrix, the recursion formula in (4.12) to compute the Fisher Information Matrix at every discrete time instant k is simply given by

$$L_{k+1} = [A^{-1}]^T L_k A^{-1} + \Omega_{k+1} \tag{4.13}$$

Since the measurement function that translates the range measurements between the trackers and the targets is non-linear it is, in general, impossible to compute the expectation in (4.7) analytically. Instead, Ω_{k+1} can be approximated using the Monte-Carlo method [20]. Using $M \in \mathbb{N}$ samples of the initial target states $\mathbf{x}_0$ from the *a priori* probability function $\mathbf{x}_k^{(r)} \sim p(\mathbf{x}_0)$, with $r \in \{1, ...M\}$, the

target dynamics in (3.8) and a sequence of known and deterministic tracker positions $\boldsymbol{\phi}_k$, the Monte-Carlo approximation consists in after determining M deterministic realisations of the target states $\mathbf{x}_{k+1}^{(r)}$ approximate expectation in (4.7) by

$$\Omega_{k+1} \approx \frac{1}{M} \sum_{r=1}^{M} H_{k+1}(\mathbf{x}_{k+1}^{(r)}, \boldsymbol{\phi}_k), \tag{4.14}$$

where the function $H_{k+1}(\mathbf{x}_{k+1}^{(r)}, \boldsymbol{\phi}_k)$ is given by

$$H_{k+1}(\mathbf{x}_{k+1}^{(r)}, \boldsymbol{\phi}_k) = [\nabla_{x_{k+1}} \mathbf{h}_k^T(\mathbf{x}_{k+1})] R^{-1} [\nabla_{x_{k+1}} \mathbf{h}_k^T(\mathbf{x}_{k+1})]^T. \tag{4.15}$$

4.3 FIM for Target Motion Model

4.3.1 Scenario A

Recalling the notation introduced in Chapter 3 and the target motion model for the scenario A introduced in Section 3.2.1, let $L_{A,k}^{[i,j]}$ be the Fisher Information Matrix about the target states $\mathbf{x}_k^{[j]} \in \mathbb{R}^2$ computed from the range measurements taken by the tracker i, with $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$. The target motion model in (3.8) is a specific case of the filtering problem introduced in (4.9) where $A = I_2$ is an identity matrix of size 2, $B = T_s I_2$ with T_s the sampling interval, $R = \sigma$, $Q = 0$ and $\boldsymbol{\phi}_k = \mathbf{p}_k$. The function $\mathbf{h}_k(\cdot)$ corresponds to the distance $d_k^{[ij]}$ between the i^{th} tracker and the j^{th} target introduced in (3.13). Using this notation the Fisher Information Matrix $L_{A,k}^{[ij]}$ can be computed recursively using equation (4.13)

$$L_{A,k+1}^{[i,j]} = [A^{-1}]^T L_{A,k}^{[i,j]} A^{-1} + \Omega_{k+1}^{[i,j]}. \tag{4.16}$$

where $\Omega_{k+1}^{[ij]}$ is given by equation (4.7). Considering the observation model in (3.14) it follows that

$$\nabla_{x_{k+1}^{[j]}} \mathbf{h}_k^T(\mathbf{x}_{k+1}^{[j]}) = \begin{bmatrix} \frac{(x_{k+1}^{[i]} - x_{k+1}^{[j]})}{d_{k+1}} \\ \frac{(y_{k+1}^{[i]} - y_{k+1}^{[j]})}{d_{k+1}} \end{bmatrix} = \begin{bmatrix} a_{k+1}^{[i,j]} \\ b_{k+1}^{[i,j]} \end{bmatrix}, \tag{4.17}$$

and from this result and using equation (4.7) it follows that Ω_{k+1} is given by

$$\Omega_{k+1}^{[i,j]} = \mathbb{E}\left\{ \frac{1}{\sigma^2} \begin{bmatrix} {a_{k+1}^{[i,j]}}^2 & a_{k+1}^{[i,j]} b_{k+1}^{[i,j]} \\ a_{k+1}^{[i,j]} b_{k+1}^{[i,j]} & {b_{k+1}^{[i,j]}}^2 \end{bmatrix} \right\}. \tag{4.18}$$

Since $\mathbf{x}_k^{[i]} = \mathbf{q}_k^{[i]} \in \mathbb{R}^2$ the Fisher Information Matrix about the target states is a two-by-two matrix. The recursion in (4.16) begins with $L_{A,0}^{[i,j]}$ computed by (4.8). Considering the prior information about the

target states defined in (3.9), $L_{A,0}^{[i,j]}$ is given by

$$L_{A,0}^{[i,j]} = [P_{A,0}^{[j]}]^{-1}, \tag{4.19}$$

where $P_{A,0}^{[j]} \in \mathbb{R}^{2\times 2}$ is the covariance matrix of the initial Gaussian distribution about the target states.

Now, let $L_{A,k}^{[j]}$ be the Fisher Information Matrix about the target states $\mathbf{x}_k^{[j]} \in \mathbb{R}^2$ computed from all range measurements taken by all the p trackers. In this scenario the function $\mathbf{h}_k(\cdot)$ corresponds to the vector containing all the distances between the trackers and the j^{th} target, $\mathbf{h}_k(\cdot) = [d_k^{[1,j]}, ..., d_k^{[p,j]}]^T \in \mathbb{R}^p$ and R is a diagonal matrix whose diagonal elements are σ, that is $R = \text{Diag}(\sigma, ..., \sigma) \in \mathbb{R}^{p\times p}$. Using a vector of measurements in the computation of (4.3) instead of a single range measurement the FIM $L_{A,k}^{[j]}$ is given by

$$L_{A,k+1}^{[j]} = [A^{-1}]^T L_{A,k}^{[j]} A^{-1} + \sum_{i=1}^{p} \Omega_{k+1}^{[i,j]}. \tag{4.20}$$

The recursion in (4.20) is again initialised with the prior information about the target states (4.19), that is

$$L_{A,0}^{[j]} = L_{A,0}^{[i,j]} = [P_{A,0}^{[j]}]^{-1}. \tag{4.21}$$

4.3.2 Scenario B

Now, considering the target motion model for the scenario B introduced in Section (3.2.2), let $L_{B,k}^{[i,j]}$ be the Fisher Information Matrix about the target states $\mathbf{x}_k^{[j]} \in \mathbb{R}^4$ computed from the range measurements taken by the tracker i, with $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$. Again, the target motion model in (3.10) is a specific case of the filtering problem introduced in (4.9) where $A = A_B$, with A_B given by equation (3.11), $R = \sigma$, $Q = 0$, $\boldsymbol{\phi}_k = \mathbf{p}_k^{[i]}$, and $\mathbf{h}_k(\cdot)$ = $d_k^{[i,j]}$. Using this notation the Fisher Information Matrix $L_{B,k}^{[i,j]}$ can be computed recursively using equation (4.13)

$$L_{B,k+1}^{[i,j]} = [A_B^{-1}]^T L_{B,k}^{[i,j]} A_B^{-1} + \Theta_{k+1}^{[i,j]}, \tag{4.22}$$

with

$$\Theta_{k+1}^{[i,j]} = \begin{bmatrix} \Omega_{k+1}^{[i,j]} & 0_{2\times 2} \\ 0_{2\times 2} & 0_{2\times 2} \end{bmatrix}, \tag{4.23}$$

where $0_{2\times 2}$ is a two by two zero matrix and $\Omega_{k+1}^{[i,j]}$ is given by equation (4.7).

Since $\mathbf{x}_k^{[j]} = [\mathbf{q}_k^{[j]}, \dot{\mathbf{q}}_k^{[j]}] \in \mathbb{R}^4$ the Fisher Information Matrix about the target states is a four-by-four matrix. The recursion in (4.16) begins with $L_{B,0}^{[i,j]}$ computed by (4.8). Considering the prior information about the target states defined in (3.9), $L_{B,0}^{[i,j]}$ is given by

$$L_{B,0}^{[i,j]} = [P_{B,0}^{[j]}]^{-1}, \tag{4.24}$$

where $P_{B,0}^{[j]} \in \mathbb{R}^{4\times4}$ is the covariance matrix of the initial Gaussian distribution about the target states.

Similarly to scenario A, let $L_{B,k}^{[j]}$ be the Fisher Information Matrix about the target states $\mathbf{x}_k^{[j]} \in \mathbb{R}^4$ computed from all range measurements taken by all the p trackers. In this scenario the function $\mathbf{h}_k(\cdot)$ corresponds to the vector containing all the distances between the trackers and the j^{th} target, $\mathbf{h}_k(\cdot) = [d_k^{[1,j]}, ..., d_k^{[p,j]}]^T \in \mathbb{R}^p$ and R is a diagonal matrix whose diagonal elements are σ, that is $R = \text{Diag}(\sigma, ..., \sigma) \in \mathbb{R}^{p\times p}$. Using a vector of measurements in the computation of (4.3) instead of a single range measurement the FIM $L_{B,k}^{[j]}$ is given by

$$L_{B,k+1}^{[j]} = [A_B^{-1}]^T L_{B,k}^{[j]} A_B^{-1} + \sum_{i=1}^{p} \Theta_{k+1}^{[i,j]}. \tag{4.25}$$

Again, the recursion in (4.20) is initialised with the prior information about the target states (4.19), that is

$$L_{B,0}^{[j]} = L_{B,0}^{[i,j]} = [P_{B,0}^{[j]}]^{-1}. \tag{4.26}$$

4.4 Optimal Geometries for Target Tracking

In the previous section, the FIM was derived under the context of the target tracking problem formulated in chapter (3). According to that formulation, as explained in section (3.2), the targets position are assumed to be described in every time instant by a Gaussian distribution. Thus, the concept of FIM derived in the previous section represents a measure of the quantity of information about a random parameter to be estimated (the target's states), in a vector observations (the range measurements). Instead, in this section, in order to derive a deterministic formulation of the Fisher Information Matrix [7], it is assumed that the target's states are described by parameters rather than Gaussian variables. This deterministic formulation will be used for a preliminary study of the optimal geometries between targets and trackers in situations where there are no restrictions and in which there is a prior knowledge of the target's states. That is, assuming that the trackers know the targets positions what are the optimal geometries between those in order to maximize the range information about the target states. The ideia behind this preliminary study is to have an insight on how the trackers trajectories will look like in an ideal scenario. These trajectories can then be used in the future to compare with the trajectories generated by the MPC algorithm. However, the MPC strategy will deal with the problems related to the restrictions imposed on the movement of the trackers and the fact that there is uncertainty associated with the knowledge of the position of the targets.

4.4.1 Deterministic Formulation

Considering the target motion model in the scenario A, the equation (4.16) allows to recursively compute the FIM about the states of the j target, $\mathbf{x}_k^{[i,j]} \in \mathbb{R}^2$, at any time instant k, from the range measurements collected from the i^{th} tracker, $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$. It can be seen from the same equation that if the initial state of the target is considered to be deterministic then the information available about the target states is only dependent on the positions of the target and tracker and their relative distance [20]. This dependency allows the computation of the FIM at the time instant k in a compact formula given by

$$L_{A,k}^{[i,j]} = \frac{1}{\sigma^2} \sum_{n=1}^{k} \begin{bmatrix} {a_n^{[i,j]}}^2 & a_n^{[i,j]} b_n^{[i,j]} \\ a_n^{[i,j]} b_n^{[i,j]} & {b_n^{[i,j]}}^2 \end{bmatrix}, \tag{4.27}$$

with $a_n^{[i,j]}$ and $b_n^{[i,j]}$ given in (4.17). The previous equation defines the FIM, $L_{A,k}^{[i,j]}$, when the states of the target are considered to be deterministic.

Consider now the equation (4.20) that allows to recursively compute the FIM about the states of the j target, $\mathbf{x}_k^{[j]} \in \mathbb{R}^2$, at any time instant k, with the range measurements collected from all the $p \in \mathbb{N}$ trackers. Following the same procedure as in (4.20), if the initial state of the target is considered to be deterministic, it is possible to derive an expression for $L_{A,k}^{[j]}$ that is only dependent on the positions of the target and trackers and their relative distances

$$L_{A,k}^{[j]} = \sum_{i=1}^{p} L_{A,k}^{[i,j]}, \tag{4.28}$$

with $L_{A,k}^{[i,j]}$ given by (4.27).

A similar relation can be derived for the case where the target motion model is the model in scenario B. In this case the equation (4.22) allows to recursively compute the FIM about the states of the j target, $\mathbf{x}_k^{[i,j]} \in \mathbb{R}^4$, at any time instant k, from the range measurements collected from the i^{th} tracker, $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$. Again, if the initial state of the target is considered to be deterministic then the information available about the target states is only dependent on the positions of the target and tracker and their relative distance. As in scenario A it is possible to compute the FIM at the time instant k in a compact formula given by

$$L_{B,k}^{[i,j]} = \begin{bmatrix} A_k^{[i,j]} & B_k^{[i,j]} \\ B_k^{[i,j]} & C_k^{[i,j]} \end{bmatrix}, \tag{4.29}$$

with

$$A_k^{[i,j]} = L_{A,k}^{[i,j]} \tag{4.30}$$

$$B_k^{[i,j]} = -\sum_{n=1}^{k} \frac{1}{\sigma^2} \begin{bmatrix} \tau_n {a_n^2}^{[i,j]} & \tau_n a_n^{[i,j]} b_n^{[i,j]} \\ \tau_n a_n^{[i,j]} b_n^{[i,j]} & \tau_n b_n^{[i,j]} \end{bmatrix}, \tag{4.31}$$

$$C_k^{[i,j]} = \sum_{n=1}^{k} \frac{1}{\sigma^2} \begin{bmatrix} \tau_n^2 {a_n^{[i,j]}}^2 & \tau_n^2 a_n^{[i,j]} b_n^{[i,j]} \\ \tau_n^2 a_n^{[i,j]} b_n^{[i,j]} & \tau_n^2 {b_n^{[i,j]}}^2 \end{bmatrix}, \tag{4.32}$$

where $L_{A,k}^{[i,j]}$ given by (4.27), $\tau_n = (k-n)T_s$, $n = \{1, ..., k\}$ and T_s is the sampling interval. The equation (4.29) defines the FIM, $L_{B,k}^{[i,j]}$, when the states of the target are considered to be deterministic. The proof of this result can be found in [20].

As in the previous scenario, consider the equation (4.25) that allows to recursively compute the FIM about the states of the j target, $\mathbf{x}_k^{[j]} \in \mathbb{R}^4$, at any time instant k, with the range measurements collected from all the $p \in \mathbb{N}$ trackers. Following the same procedure as in (4.25), if the initial state of the target is considered to be deterministic, $L_{B,k}^{[j]}$ is given by

$$L_{B,k}^{[j]} = \sum_{i=1}^{p} L_{B,k}^{[i,j]}, \tag{4.33}$$

with $L_{B,k}^{[i,j]}$ given by (4.27). Notice that the expected value operator, $\mathbb{E}$, in (4.18) was discarded in (4.27), (4.28), (4.29) and (4.33) since the target states are deterministic.

4.4.2 Optimal Geometries

In this subsection ideal geometries between the trackers and the targets that maximize the range information acquired will be discussed. Using the deterministic formulation of the FIM introduced in section (4.4.1), the criterion used to maximize the range-related information acquired will be the determinant of the Fisher Information Matrix. Given a sequence of positions for each target, an ideal geometry is a consecutive sequence of positions for the trackers, such that the Fisher Information Matrix determinant is maximal. Different combinations in the number of targets and trackers and their influence on the optimal trajectories will be studied.

With this objective in mind let $\alpha_k^{[i,j]}$ be the angle between the projection on the $x_{\mathcal{I}} - y_{\mathcal{I}}$ plane of the relative position vector from tracker i to target j and the $x_{\mathcal{I}}$ − axis at the discrete time instant k. The angle $\alpha_k^{[i,j]}$ is illustrated in Figure 4.1 and is given by

$$\cos(\alpha_k^{[i,j]}) = \left(\frac{x_k^{[i]} - x_{T;k}^{[j]}}{\left\| \mathbf{p}_k^{[i]} - \mathbf{q}_k^{[j]} \right\|} \right). \tag{4.34}$$

In the specific case where the trackers and the targets operate at the same depth, according to [29] the following equation is an equivalent definition of $L_{A,k}^{[i,j]}$ in (4.28)

$$L_k^{[i,j]} = \frac{1}{\sigma^2} \sum_{n=1}^{k} \begin{bmatrix} \cos(\alpha_n^{[i,j]})^2 & \cos(\alpha_n^{[i,j]}) \sin(\alpha_n^{[i,j]}) \\ \cos(\alpha_n^{[i,j]}) \sin(\alpha_n^{[i,j]}) & \sin(\alpha_n^{[i,j]})^2 \end{bmatrix}. \tag{4.35}$$

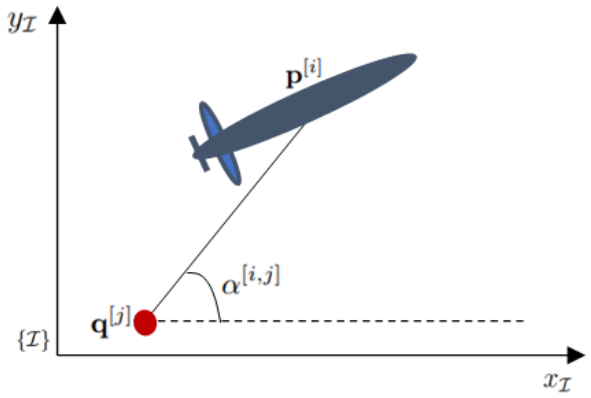


Figure 4.1: Angle $\alpha^{[i,j]}$ formed by the relative position vector from tracker i to target j and the $x_{\mathcal{I}}-$ axis

Let also $c^{[i,j]}$ be a constant given by

$$c^{[i,j]} = \left(\frac{z^{[i]} - z_T^{[j]}}{d_{max}} \right). \tag{4.36}$$

4.4.2.A Single Target - Single Tracker

Consider the case where a single tracker is trying to localize and track a single target. Since, in this scenario, $p = i = 1$ and $q = i = 1$ the superscripts i and j will be discarded for simplicity of notation.

Scenario A In this scenario the Fisher Information Matrix determinant, that encodes the information about the target states, is maximal when $L_{A,k} = L_{A,k}^{[\text{opt}]}$ [20] where

$$L_{A,k}^{[\text{opt}]} = \frac{1-c}{\sigma^2} \begin{bmatrix} \frac{k}{2} & 0 \\ 0 & \frac{k}{2} \end{bmatrix}, \tag{4.37}$$

with c given by (4.36).

Scenario B For scenario B the Fisher Information Matrix determinant, that encodes the information about the target states, is maximal when $L_{B,k} = L_{B,k}^{[\text{opt}]}$ [20] with

$$L_{B,k}^{[\text{opt}]} = \frac{1-c}{2\sigma^2} \begin{bmatrix} kI_2 & \Delta_1 I_2 \\ \Delta_1 I_2 & \Delta_2 I_2 \end{bmatrix}, \tag{4.38}$$

where I_2 is two-by-two identity matrix, c is given by (4.36), $\Delta_1 \triangleq -\sum_{n=1}^{k} \tau_n$ and $\Delta_2 \triangleq -\sum_{n=1}^{k} \tau_n^2$. As in equation (4.29) $\tau_n = (k-n)T_s$, $n = \{1, ..., k\}$ and T_s is the sampling interval

That is, in order to obtain the maximum range information about the target states, a single tracker must realize k measurements from k positions such that $L_{A,k} = L_{A,k}^{[\text{opt}]}$, considering the target model in

scenario A, and $L_{B,k} = L_{B,k}^{[\text{opt}]}$, considering the target model described in scenario B.

It is now possible to study target tracking geometries that lead to the optimal Fisher Information Matrices defined in equations (4.37) and (4.38) for scenario A and B respectively. Starting with scenario A, consider that the target and tracker operate at the same depth, that is $z_T = z$ and consequently $c = 0$. From equations (4.35) and (4.37), in order to obtain the maximum range information the following conditions must be satisfied:

$$\begin{aligned} \sum_{n=1}^{k} \cos^2(\alpha_n) &= \frac{k}{2} \\ \sum_{n=1}^{k} \sin^2(\alpha_n) &= \frac{k}{2} \\ \sum_{n=1}^{k} \cos(\alpha_n)\sin(\alpha_n) &= 0, \end{aligned} \tag{4.39}$$

where α_n is given by (4.34). It is proven in [29] using Fourier analysis [30] that one possible solution for the conditions in (4.39) is to have

$$\alpha_n = \frac{2\pi(n-1)}{k}, \tag{4.40}$$

with $n \in \{1, ..., k\}$. Notice that there are an infinity number of solutions for this problem, since other solutions can be obtained by simply allowing $\alpha_n = \frac{2\pi n}{k} + \alpha_c$ where α_c is a constant, but arbitrary, angle in $[0, 2\pi]$.

From equation (4.40) it is possible to establish the following relation between two successive measurements that guarantee the maximum range information:

$$\alpha_{n+1} - \alpha_n = \frac{2\pi}{N}, \tag{4.41}$$

with $n \in \{1, .., k\}$ and $N \in \mathbb{N}$, $N > 2$. If (4.41) is satisfied between two consecutive range measurements then at the discrete time instants $k = Nl$ with $l \in \mathbb{N}$, $L_{A,k} = L_{A,k}^{[\text{opt}]}$ and as a consequence the Fisher Information Matrix determinant is maximal.

An ideal trajectory for a tracker to guarantee the maximum range information about the target is to encircle the target position such that two successive measurements satisfy the condition in (4.41). An illustration of such trajectory is shown in Figure 4.2. If in addition, the trackers and the target are operating at different depths the ranges measured by the tracker should be the maximum possible, $d_n = d_{max}$ for every $n \in \{1, ..., k\}$ [20].

The same conclusion on the ideal trajectory can be drawn for scenario B and the proof can be found in [20]. That means that the same trajectory, illustrated in Figure 4.2, is ideal for both scenario A and B. Therefore, a configuration that maximizes the amount of information in the range measurements about

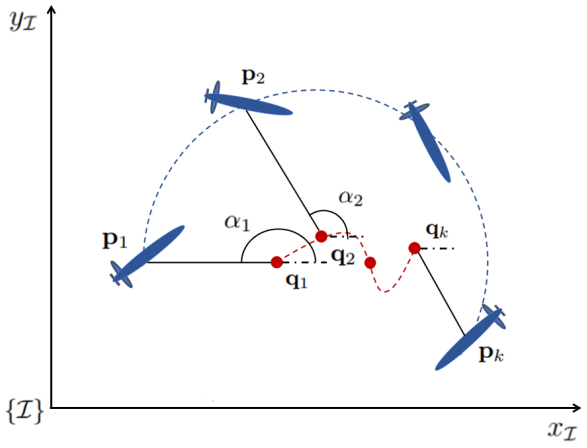


Figure 4.2: Illustration of an ideal geometry between a single target (red) a single tracker (blue). The tracker encircle the target over time.

the target states in scenario A, that is the target position, will maximize the amount of information about the target states in scenario B, that is the target position and velocity. Henceforward the same conclusion can be drawn for all different cases with a different number of targets or trackers, that is to say, an ideal geometry for scenario A is an ideal geometry for scenario B.

4.4.2.B Single Target - Multiple Trackers

Consider now the case where p trackers are trying to localize and track a single target j.

Secnario A If the target states are considered deterministic then the determinant of $L_{A,k}^{[i,j]}$ is maximal when $L_{A,k}^{[j]} = L_{B,k}^{[j][\text{opt}]}$ [20], where

$$L_{A,k}^{[j][\text{opt}]} = \left(p - \sum_{i=1}^{p} c^{[i,j]}\right) \frac{1}{\sigma^2} \begin{bmatrix} \frac{k}{2} & 0 \\ 0 & \frac{k}{2} \end{bmatrix}. \tag{4.42}$$

Secnario B If the target states are considered deterministic then the determinant of $L_{B,k}^{[i,j]}$ is maximal when $L_{k}^{[j]} = L_{B,k}^{[j][\text{opt}]}$ [20], where

$$L_{B,k}^{[j][\text{opt}]} = \left(p - \sum_{i=1}^{p} c^{[i,j]}\right) \frac{1}{\sigma^2} \begin{bmatrix} \frac{k}{2} & 0 \\ 0 & \frac{k}{2} \end{bmatrix}. \tag{4.43}$$

From the Fisher Information Matrix that encodes the maximum information about the target states from the range measurements for both scenario A and B, it is possible to study the target trackers geometries that lead to these matrices in equations (4.42) and (4.43). Again, an ideal geometry for the target model in scenario A is also an ideal geometry for the target model in scenario B, and thus

no distinction will be made when discussing the optimal geometries between scenario A and B. For this reason, in the remaining of this section, when referring to the Fisher Information Matrix, $L_k^{[j]}$, the subscript A or B is dropped. The proof for the results in this subsection can be found in [20].

Let $\beta^{[i,r;j]}$ be the angle formed by the two relative position vectors from the trackers i and r, to the target j in the $x_\mathcal{I} - y_\mathcal{I}$ plane, as shown in Figure 4.3 .

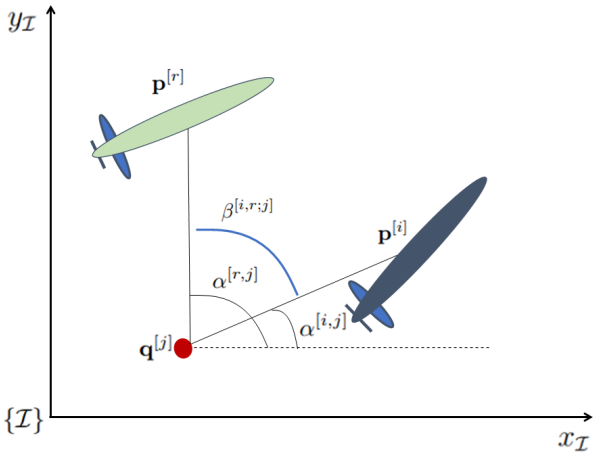


Figure 4.3: Illustration of the angle $\beta^{[i,r;j]}$ (in blue) formed by the two relative position vectors from the trackers i and r, to the target j in the $x_\mathcal{I} - y_\mathcal{I}$ plane.

Consider the case where two trackers are localizing a single target j, that is $p = 2$. In order to the determinant of $L_k^{[j]}$ to be maximal and consequently obtain the maximal range information an ideal trajectory for the trackers is to maintain the relative position vectors from them to the target orthogonal. That is,

$$\beta_n^{[1,2;j]} = \frac{\pi}{2} + l\pi \tag{4.44}$$

for all $n \in \{1, ..., k\}$ and $l \in \mathbb{N}$. If in addition, the trackers and the target are operating at different depths the ranges measured by each tracker should be the maximum possible, $d_n^{[i,j]} = d_{max}$ with $n \in \{1, ..., k\}$ and $i \in \{1, 2\}$, [20]. Such geometry is illustrated in Figure 4.4.

Consider now the case where $p \geqslant 3$ trackers are localizing a single target j. In order to retrieve the maximal range-related information about the target states an ideal trajectory for the trackers is to maintain a formation around the target position in such way that the trackers are equally distributed in a circumference centred in the projection of the target position on the $x_\mathcal{I} - y_\mathcal{I}$, [9] [20]. This formation is equivalent of having

$$\beta_n^{[i,r;j]} = \frac{2\pi}{p} \tag{4.45}$$

between two consecutive trackers in the formation, where $i, r \in 1, ..., p$ and $n \in \{1, ..., k\}$. If in addition, the trackers and the target are operating at different depths the ranges measured by each tracker should be the maximum possible, $d_n^{[i,j]} = d_{max}$ with $n \in \{1, ..., k\}$ and $i \in \{1, 2\}$, [20]. Figure 4.5 illustrates,

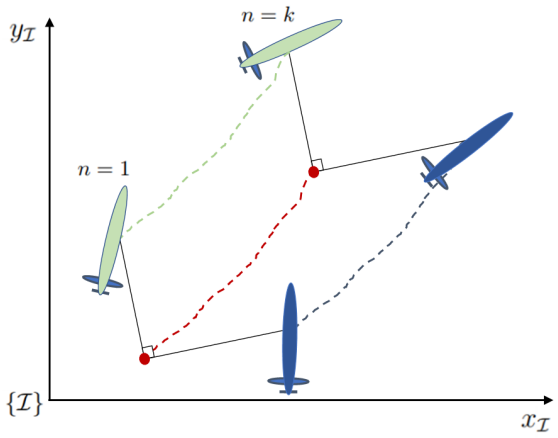


Figure 4.4: An ideal geometry between a single target (red) and two trackers (blue and green)

for the case of three trackers and one target, an ideal geometry that leads to the maximum rage-related information about the target states. In this case the trackers and the target are at the same depth and they move in a such way that the target is kept in the centre of a circumference where the three trackers are equally distributed.

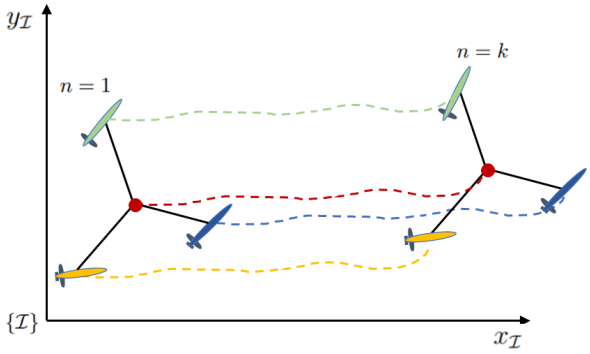


Figure 4.5: An ideal geometry between a single target (red) and three trackers (blue, green and yellow)

4.4.2.C Multiple Target - Multiple Tracker

If p trackers are localizing q targets, $q, p \in \mathbb{N}$, an ideal geometry between trackers and targets maximizes the determinant of every Fisher Information Matrix about the states of each target simultaneously [20]. That is $L_k^{[j]} = L_{k;[\text{opt}]}^{[j]}$, for $j \in \{1, ..., p\}$, where $L_{k;[\text{opt}]}^{[j]}$ is given by (4.42) and (4.43) for the target model in scenario A and B respectively. Again, a geometry that maximizes the amount of information about the targets' states in scenario A also maximizes the amount of information about the targets' states in scenario B.

It is now possible to study target tracking geometries that lead to the optimal Fisher Information

Matrices defined in equations (4.42) and (4.43) for every target in operation. Starting with scenario where only two trackers are trying to localize q targets, for the determinant of $L_k^{[j]}$ to be maximal for every $j \in \{1, ..., q\}$, and consequently obtain the maximal range information, an ideal trajectory for the trackers is to maintain the relative position vectors from them to the each target orthogonal [20]. That is, the condition in (4.44) must be verified for every target. If in addition, the trackers and the target are operating at different depths the ranges measured by each tracker should be the maximum possible, $d_n^{[i,j]} = d_{max}$ with $n \in \{1, ..., k\}$ and $i \in \{1, 2\}$, [20]. Such geometry is illustrated in Figure 4.6.

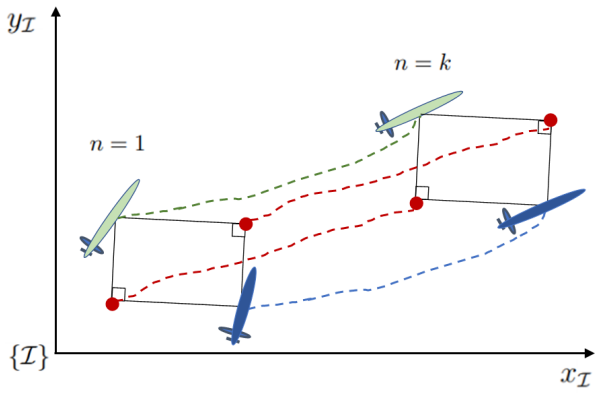


Figure 4.6: An ideal geometry between two targets (red) and two trackers (blue and green)

The same conclusion can be drawn for the case where $p \geqslant 3$ trackers are in operation. That is, for the determinant of $L_k^{[j]}$ to be maximal for every $j \in \{1, ..., q\}$, and consequently obtain the maximal range information, an ideal trajectory for the trackers is to maintain the condition in (4.45) verified, instead of the condition in (4.44).

5

Model Predictive Control

As shown in last section, the problem of determining optimal positions for trackers to maximize the information about the target states can be solved theoretically without resorting to an MPC strategy. In fact the problem was already addressed with some extend in previous publications [11] [9]. Even providing important information about how the ideal trajectories look like, this approach has as it major drawback the fact that it requires knowledge of the positions of the targets and their future motion. Thus, an effective target localization system must be able to simultaneously use the theoretical knowledge of the optimal sensor placement to plan the trajectories of each tracker, estimate the position of each target and re-plan the trajectories as the knew estimation is processed. An MPC strategy naturally arrises in this context. By planning the optimal trajectories with MPC it is possible to deal with practical situations such as: The prior information about the target states is incorrect; The motion of the trackers is restricted to a motion model dependent on each vehicle dynamics; There are restrictions in space, for example the tracker may need to keep the range to the target lower than a certain distance or avoid obstacles; The real position of the targets has to be estimated rather than being known to the tracker.

5.1 MPC Strategy

As mentioned before in this section, we will assume that the trackers do not know the positions of the targets, thus an estimator will be included in the proposed strategy. This MPC strategy will include three stages: i) Optimal motion planning for the trackers based on the information about the targets states ii) Motion control of the trackers based on the planned trajectories iii) Estimation of the positions of the targets using range measurements. The determinant of Fisher Information Matrix defined in Section 4 will be used as a cost function to be maximized, thus assuring that the motions renders the maximum information about the positions of the targets. The objective of this strategy is to maximize the information available over a short period of time by choosing the appropriate control inputs, and as a consequence the maneuvers for the trackers. Then only the first input is applied and the previous process is repeated.

The MPC blueprint is represented in figure 5.1. It is assumed that the planing stage is executed by a centralised system that posses information about the position of each tracker, $\mathbf{p}_k^{[i]}$ with $i \in \mathcal{S}$, and an estimation of the position of each target $\hat{\mathbf{x}}_k^{[j]}$ with $j \in \mathcal{S}_t$. Similarly, the estimation stage is assumed to be executed by a centralized system that in addition to having information about the position of all trackers and an estimation of the position of all targets, it also possess all the collected measurements from each tracker to each target at the discrete time instant k.

The following sections focus on the implementation of the Extended Kalman Filter (EKF) in the estimation stage and the construction of the Optimal Control Problem (OCP) to be solved in the planning stage.

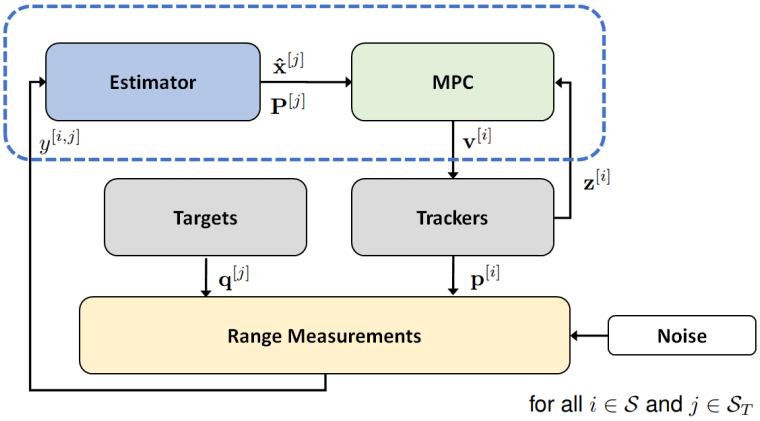


Figure 5.1: Model predictive control strategy to localize an track multiple targets

5.2 Extended Kalman Filter Implementation

Since the measurement equation (3.14) is non-linear, target position estimation using range measurements, is a non-linear state estimation problem. The Kalman Filter is therefore inappropriate for this specific problem as explained in Section 2.1. To overcome the non-linaraty of the measurement equation an Extended Kalman Filter (EKF) is used. The EKF lifts the requirement of a linear measuremnt equation on the filtering problem by approximating the non-linear equation by a first order Taylor Series expansion [31].

The EKF recursive equations were presented in Section 2.1.4. In the context of range-based target localization, the goal of the filter is to produce an estimate of each target states $\hat{\mathbf{x}}_{k+1}^{[j]}$ at the discrete time $k+1$ and its covariance $P_{k+1}^{[j]}$, given the measurement $\mathbf{y}_{k+1}^{[j]}$, and an estimate of the target states $\hat{\mathbf{x}}_{k}^{[j]}$ at the discrete time k, with its covariance matrix $P_{k}^{[j]}$, where $j \in \mathcal{S}_T$.

The predict step in the filter follows the linear target motion model in equation (3.8) and its given by

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k}^{[j]} &= A\hat{\mathbf{x}}_{k|k}^{[j]} + B\mathbf{u}_{k}^{[j]} \\ P_{k+1|k}^{[j]} &= A\, P_{k|k}^{[j]} + Q_k \end{aligned} \tag{5.1}$$

where Q_k is the covariance matrix of the state transition, or process noise, in (4.1).

Considering the measurement model described in Section 3.3, the measurement prediction is given by

$$\hat{\mathbf{y}}_{k+1|k}^{[j]} = \mathbf{h}_k(\hat{\mathbf{x}}_{k+1|k}^{[j]}) = \begin{bmatrix} \hat{d}_{k+1|k}^{[i,j]} \\ \vdots \\ \hat{d}_{k+1|k}^{[p,j]} \end{bmatrix} \tag{5.2}$$

where $\hat{d}_{k+1|k}^{[i,j]}$ denotes the predicted distance between the i^{th} tracker and the j^{th} target given by (3.13),

for all $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$.

The update step is then defined by

$$\begin{aligned} K_{k+1}^{[j]} &= P_{k+1|k}^{[j]} H_{k+1}^T \left[H_{k+1} P_{k+1|k}^{[j]} H_{k+1}^T + R_k \right]^{-1} \\ \hat{\mathbf{x}}_{k+1|k+1}^{[j]} &= \hat{\mathbf{x}}_{k+1|k}^{[j]} + K_{k+1}^{[j]} \left[\mathbf{y}_{k+1}^{[j]} - \mathbf{h}_k(\hat{\mathbf{x}}_{k+1|k}^{[j]}) \right] \\ P_{k+1|k+1}^{[j]} &= \left[I_n - K_{k+1}^{[j]} H_{k+1} \right] P_{k+1|k}^{[j]}, \end{aligned} \tag{5.3}$$

where I_n is an identity matrix of size n, n is the dimension of the target state vector, R_k is the covariance matrix of the measurement noise in equation (4.1) and H_{k+1} is the linearized measurement matrix, or Jacobian, evaluated in the predicted state $\hat{\mathbf{x}}_{k+1|k}^{[j]}$. defined by

$$H_{k+1} = \left[\nabla_{\mathbf{x}_{k+1}^{[j]}} \mathbf{h}_k^T(\mathbf{x}_{k+1}^{[j]}) \right]^T \Big|_{\mathbf{x}_{k+1} = \hat{\mathbf{x}}_{k+1|k}}. \tag{5.4}$$

Since the dimension of the target state vector is different between the target motion model in scenario A and in scenario B, the Jacobian matrix, although computed with equation (5.4) in both cases, has different dimensions. The distinction is made in the two following paragraphs.

Scenario A: Recall that in scenario A is assumed that the velocity vector of each target is known to all the p trackers in the mission, and thus only the position needs to be estimated. In that scenario, as explained in 3.2.1, the state vector of the target j is given by $\mathbf{x}_k^{[j]} = \mathbf{q}_k^{[j]} = [x_{T;k}^{[j]}, y_{T;k}^{[j]}]^T \in \mathbb{R}^2$, where $\mathbf{q}_k^{[j]}$ is the target position vector at discrete time k in the horizontal plane $x_\mathcal{I} - y_\mathcal{I}$. Recalling that $\mathbf{p}_k = [x_k^{[i]}, y_k^{[i]}] \in \mathbb{R}^2$ denotes the horizontal position of the tracker i in $\mathcal{I}$, the Jacobian matrix in equation (5.4) is given by

$$H_{k+1} = \begin{bmatrix} h_{1,1} & h_{1,2} \\ \vdots & \vdots \\ h_{p,1} & h_{p,1} \end{bmatrix}. \tag{5.5}$$

and its elements are computed by

$$\begin{aligned} h_{i,1} &= \frac{\left(x_{k+1}^{[i]} - \hat{x}_{T;k+1|k}^{[j]} \right)}{\hat{d}_{k+1|k}^{[i,j]}} \\ h_{i,2} &= \frac{\left(y_{k+1}^{[i]} - \hat{y}_{T;k+1|k}^{[j]} \right)}{\hat{d}_{k+1|k}^{[i,j]}}, \end{aligned} \tag{5.6}$$

for all $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$.

Scenario B: In scenario B is assumed that the trackers need to estimate both the position and the velocity vectors of each target. In that scenario, as explained in 3.2.2, the state vector of the target j is

given by $\mathbf{x}_k^{[j]} = [\mathbf{q}_k^{[j]}; \dot{\mathbf{q}}_k{}^{[j]}] \in \mathbb{R}^4$. Since the target state vector has four dimensions, the Jacobian matrix in (5.4) is given by

$$H_{k+1} = \begin{bmatrix} h_{1,1} & h_{1,2} & h_{1,3} & h_{1,4} \\ \vdots & \vdots & \vdots & \vdots \\ h_{p,1} & h_{p,1} & h_{p,3} & h_{p,4} \end{bmatrix}. \tag{5.7}$$

where its elements $h_{i,1}$ and $h_{i,2}$ are given by (5.6) and $h_{i,3} = h_{i,4} = 0$ for all $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$.

The filter is initialised with an initial estimation of the target state vector and an associated co-variance that represents the uncertainty on the initial guess. Then the filter follows a consecutive sequence of prediction and filtering cycles in order to improve the estimation as explained in section 2.1.4. With this implementation there is one Extended Kalman Filter for each target, however it is assumed that all the computations are performed in a centralised system. The filter was implemented in *Matlab*.

5.3 Localization and Pursuit

As stated before, the main goal of the designed MPC strategy is to find optimal controls for the trackers such that the trackers follow trajectories that yield maximum range-information about the target states. However, other objectives, such as i) keeping the distance between the trackers and trackers below a certain distance ii) minimize the energy consumption iii) control the smoothness of the control inputs, can be simultaneously be achieved. In this section the distinct objectives in the MPC strategy are discussed and cost function that transcribe these objectives are presented.

Using equation (4.20), at time k it is possible to predict the FIM constructed with $p*N$ measurements in the time instants $k+1, ..., k+N$, where N is the prediction horizon and p is the number of trackers. Here is relevant to remark how the predicted FIM depends on the trackers inputs $\mathbf{v}^{[i]}\,;\, i \in \mathcal{S}$ and states $\mathbf{z}^{[i]}\,;\, i \in \mathcal{S}$, through the motion model in section 3.1. Let $\hat{\mathbf{x}}_k^{[j]}\,;\, j \in \mathcal{S}_T$ be the state estimate of the j^{th} target at the discrete time k and $\hat{P}_k^{[j]}$ be the correspondent covariance matrix. Then the predicted FIM be defined by

$$L_p^{[j]}(\mathbf{z}_k^{[i]}\,,\, \hat{\mathbf{x}}_k^{[j]}\,,\, \hat{P}_k^{[j]}\,,\, \bar{\mathbf{v}}^{[i]}\,,\, \bar{\mathbf{u}}^{[j]}) \triangleq \begin{cases} L_{A,N}^{[j]} & \text{in scenario A} \\ L_{B,N}^{[j]} & \text{in scenario B} \end{cases}, \tag{5.8}$$

where $\bar{\mathbf{v}}^{[i]} = \left[\left[\mathbf{v}_k^{[i]}\right]^T, ..., \left[\mathbf{v}_{k+N-1}^{[i]}\right]^T\right]^T\,;\, i \in \mathcal{S}$, and $\bar{\mathbf{u}}^{[j]} = \left[\left[\mathbf{u}_k^{[j]}\right]^T, ..., \left[\mathbf{u}_{k+N-1}^{[j]}\right]^T\right]^T\,;\, j \in \mathcal{S}_T$, are the predicted tracker and target input vectors over the prediction horizon respectively. $L_{A,N}^{[j]}$ and $L_{B,N}^{[j]}$ are computed by equations (4.20) and (4.25) respectively . As explained in Section 4, at each time instant k, the FIM is initialized with the prior information about the target states, here defined by $\hat{P}_k^{[i]}$.

The objective of the optimizer is to, at each time instant k, find the next N optimal tracker inputs $\bar{\mathbf{v}}^{[i]}$

that maximize the determinant of the Fisher Information Matrix in (5.8) which is equivalent to minimize

$$J_{\text{FIM}} = \sum_{j=1}^{q} -\lambda_j \cdot \log\left(\det\left(L_p^{[j]}(\mathbf{z}_k^{[i]}\,,\,\hat{\mathbf{x}}_k^{[j]}\,,\,\hat{P}_k^{[j]}\,,\,\bar{\mathbf{v}}^{[i]},\,\bar{\mathbf{u}}^{[j]})\right)\right), \tag{5.9}$$

where q is the number of targets, and λ_j is the Pareto weight that capture the relative importance attached to each of the targets in terms of desired positioning accuracy [32]. The logarithm of the determinant of the FIM is used instead of simply the determinant because it makes the computation of the optimal solution quicker and easier [33].

As explained in Section 3.2, in practice the range range measurements can only be taken up to a certain distance d_{max}. In order to assure that the trackers are always in the vicinity of the targets, such that their distance is always lower than d_{max}, a pursuit cost is introduced in the optimal control problem,

$$J_{\text{Pursuit}} = \sum_{i=1}^{p}\sum_{j=1}^{q}\sum_{n=k+1}^{k+N} -\log(d_{max}^{[i,j]} - \bar{d}_n^{[i,j]} - \sigma), \tag{5.10}$$

where $\bar{d}_n^{[i,j]}$ denotes the predicted distance between the i^{th} tracker and the j^{th} target, which is computed using equation (3.13) over the prediction horizon, σ is the variance of the measurement noise, and $d_{max}^{[i,j]}$ is the upper bound for the distance between the i^{th} tracker and the j^{th} target, for all $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$.

Let $\mathbf{e}_k^{[i]} = \left[v_k^{[i]}, r_k^{[i]}\right]$, where $v_k^{[i]}$ and $r_k^{[i]}$ are respectively the linear and angular velocity of the i^{th} tracker at the discrete time instant k. In order to curb the energy consumption over all the trackers and using $\mathbf{e}_k^{[i]}$ as a proxy for the energy consumption, a energy-based cost to be minimized in the optimal control problem is introduced:

$$J_{\text{Energy}} = \sum_{i=1}^{p}\sum_{n=k}^{k+N-1} \mathbf{e}_n^{[i];T} E_i \mathbf{e}_n^{[i]}, \tag{5.11}$$

where $E_i \in \mathbb{R}^2$ is a diagonal matrix, whose diagonal elements are positive an indicate the costs related with the linear and angular velocities.

In order to control the smoothness of the linear and angular velocities of each tracker, consider the following input related cost to be minimized in optimal control problem

$$J_{\text{Input}} = \sum_{i=1}^{p}\sum_{n=k}^{k+N-1} \mathbf{v}_n^{[i];T} W_i \mathbf{v}_n^{[i]}, \tag{5.12}$$

where $\mathbf{v}_k^{[i]} = \left[v_k^{[i]}, r_k^{[i]}\right]$ corresponds to the control inputs of each tracker at the discrete time instant k and $W_i \in \mathbb{R}^2$ is a diagonal matrix, whose diagonal elements are positive an indicate the costs related with the linear and angular accelerations.

5.4 Optimal Control Problem

The trackers, the targets and their relative distances determine the constrains and variables of the optimization problem to be solved. The objectives of the optimization problem were described in the previous section. As explained, apart from the main goal of determining trajectories for the trackers that are rich in range-information about the targets states, there are other goals such as, energy saving, control smoothness and minimal distance between trackers and targets. All these costs variables depend on the optimization variables which are the trackers' input controls. In order to jointly optimize all the different costs while giving more importance to some of them, Pareto optimization techniques will be used [32] [34]. The idea is to optimize the combination of the logarithms of the determinants of the FIM's for each target and all the other costs presented in the previous section. Different weights can be associated with each cost, transcribing the importance of such cost in the overall optimization problem. With this formulation different targets can have different weights in the optimzation problem, and thus the trajectories can be planned to render more information about the states of the targets with higher weight.

The optimal control problem to be addressed in the proposed MPC strategy is presented in equation (5.13).

$$
\begin{aligned}
\min_{\bar{\mathbf{v}}^{[i]}; i\in\mathcal{S}} \quad & J_{\text{FIM}} + \rho_1 J_{\text{Pursuit}} + \rho_2 J_{\text{Energy}} + \rho_3 J_{\text{Input}} \\
\text{subject to} \quad & \\
& \bar{\mathbf{z}}_{n+1}^{[i]} = \mathbf{g}(\bar{\mathbf{z}}_n^{[i]}, \bar{\mathbf{v}}_n^{[i]}), && i \in \mathcal{S}, \\
& \bar{\mathbf{p}}_n^{[i]} = C\bar{\mathbf{z}}_n, && i \in \mathcal{S}, \\
& \bar{\mathbf{z}}_k^{[i]} = \mathbf{z}_k^{[i]}, && i \in \mathcal{S}, \\
& \bar{\mathbf{z}}_n^{[i]} \in \mathcal{Z}^{[i]}\ ,\ \bar{\mathbf{v}}_n^{[i]}, \in \mathcal{V}^{[i]}, && i \in \mathcal{S}, \\
& \bar{\mathbf{x}}_{n+1}^{[j]} = f(\bar{\mathbf{x}}_n^{[j]}, \bar{\mathbf{u}}_n^{[j]}), && j \in \mathcal{S}_T, \\
& \bar{\mathbf{x}}_k^{[j]} = \hat{\mathbf{x}}_k^{[j]}, && j \in \mathcal{S}_T, \\
& \bar{d}_n^{[i,j]} = \left\| \bar{\mathbf{p}}_n^{[i]} - \bar{\mathbf{q}}_n^{[j]} \right\|, && j \in \mathcal{S}_T \text{ and } i \in \mathcal{S}, \\
& \text{for every } n \in \{k, ..., k+N-1\}, &&
\end{aligned}
\tag{5.13}
$$

where $\rho_1, \rho_2, \rho_3 \geq 0$ are the weighting cost associated with the pursuit, energy and input costs respectively. Notice that the constraints equations are associated with the target, tracker and measurement models presented in chapter 3. The variables with bar in the constrain equations of the optimal control problem in (5.13) represent the predicted variables. The bar distinguish them from the actual variables, which are without bar.

The solution of the optimal control problem depends on the initial conditions, which are different every time instant due to the target and tracker movements. As a consequence the optimal control problem has to be solved at every time k in order to determine the new optimal solution. Even providing optimal control inputs for a short period of time N for each tracker, the only control input that effectively applied to each tracker is the first one. Then, in the next time instant, a new optimal control problem will be

solved and a new solution for the control inputs for each target is determined.

5.5 Receding Horizon Implementation

The flow chart of the proposed algorithm for multiple target localization and tracking using multiple trackers is illustated in Figure 5.2.

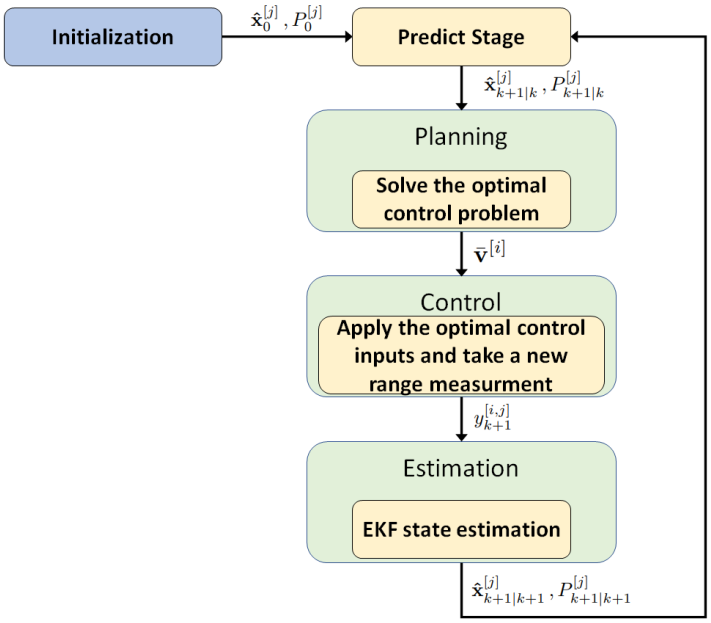


Figure 5.2: Model predictive control strategy to localize an track multiple targets

The pseudo-code of the proposed algorithm is described in Algorithm 5.1. For simulation proposes the algorithm was entirely implemented in *Matlab*. The software Casadi [35] was used to solve the OPC that arises in the proposed MPC strategy using the multiple shooting method described in section 2.3.3.

Algorithm 5.1: Receding horizon algorithm for target localization and tracking using multiple trackers; For every $i \in \mathcal{S}$ and $j \in \mathcal{S}_T$

Initialization: $(k = 0)$
For model A: $\hat{\mathbf{x}}_0^{[j]} = \hat{\mathbf{x}}_{A,0}^{[j]},\ P_0^{[j]} = P_{A,0}^{[j]}$
For model B: $\hat{\mathbf{x}}_0^{[j]} = \hat{\mathbf{x}}_{B,0}^{[j]},\ P_0^{[j]} = P_{B,0}^{[j]}$
for every sampled time $k = 1, 2....$ **do**
    EKF predict stage to estimate $\hat{\mathbf{x}}_{k|k+1}^{[j]}, P_{k|k+1}^{[j]}$;
    Solve the OPC in equation (5.13) to obtain the optimal control inputs $\bar{\mathbf{v}}$;
    Move the trackers using the optimal controls;
    Obtain the new ranges from all the trackers to all the targets;
    EKF update stage to obtain the estimate $\hat{\mathbf{x}}_{k+1|k+1}^{[j]}, P_{k|k+1}^{[j]}$;
end

6

Results

In order to assess the performance of the proposed MPC, strategy simulations were conducted and the corresponding results are presented in this chapter. Different scenarios are analysed where the movement of a different number of trackers is planned in order to maximize the information about the positions of several targets. The target dynamic model and observation model, presented in sections 3.2 and 3.3 respectively, were used to perform the simulation experiments. The tracker motion model used was introduced in section 3.1. This model was discretized in order to be implemented computationally.

In all the different simulation scenarios, the sampling period of the MPC scheme is $T = 2s$ and the prediction horizon is set as $N = 6$. The measurement sampling period is considered to be $2s$. It is also assumed that the ranges can only be measured within a maximum distance of $100m$ and thus the trackers are required to keep a distance less than $100m$ to every target. To simulate the randomness in the measures, the standard deviation of the measurement model is $\sigma = 0.5m$ in all simulations. Furthermore, it was assumed that the trackers' linear and angular velocities were constrained to

$$v^{[i]} \in [0, 4]\ m/s\ ;\ r^{[i]} \in [-0.2, 0.2]\ rad/s \tag{6.1}$$

for all $i \in \mathcal{S}$. Also the control inputs, linear and angular accelerations were constrained to

$$a_v^{[i]} \in [-0.1, 0.1]\ m/s^2\ ;\ a_r^{[i]} \in [-0.01, 0.01]\ rad/s^2 \tag{6.2}$$

for all $i \in \mathcal{S}$. The simulations were conducted using *Matlab* and, as explained in section 5.5, the software Casadi [35] was used to solve the optimal control problem in the MPC.

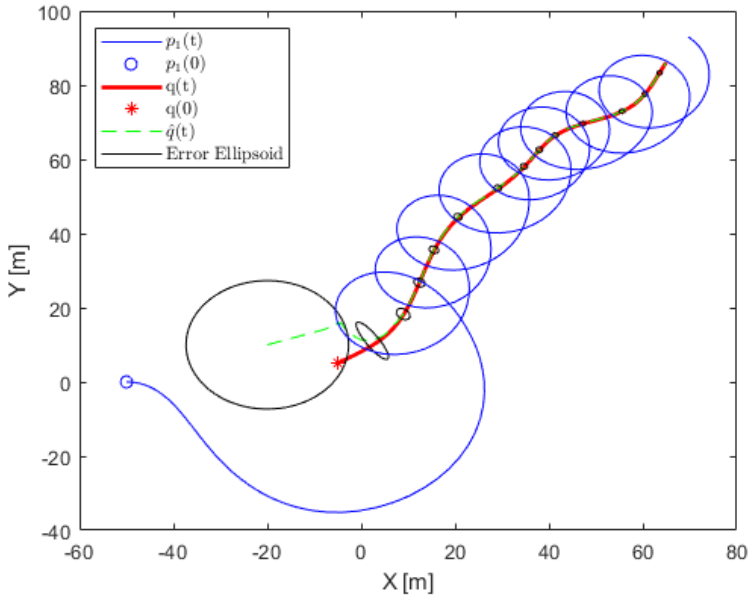
6.1 Single Target - Single Tracker Results

In this section, a single tracker, $p = 1$, trying to localize a single moving target, $q = 1$, using range information was considered. In order to establish a comparison between the performance of the proposed strategy in scenario A and scenario B, the same simulation is repeated for the target model for scenario A, described in section 3.2.1, and the target model for scenario B, described in section 3.2.2. The simulation parameters that are common to both scenarios are given in table 6.1.

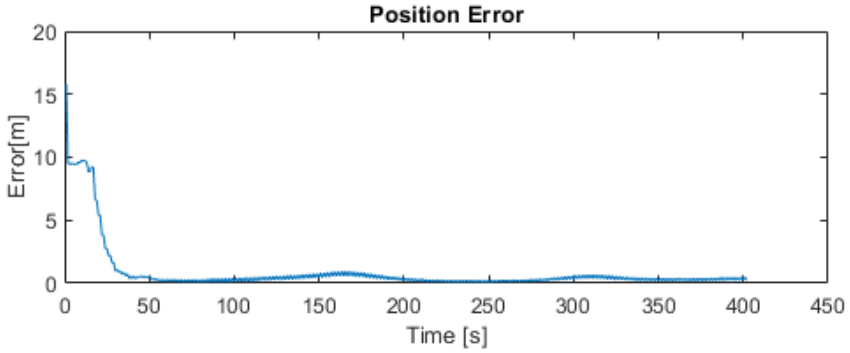
Table 6.1: Single target single tracker simulation parameters

		Parameters
Tracker	Initial Position	$\mathbf{p}_0 = [-50, 0]\ m$
Target	Velocity Vector	$\mathbf{u} = \begin{bmatrix} 0.2 + 0.1\cos(0.1x^q) \\ 0.2 + 0.1\sin(0.1x^q) \end{bmatrix} m/s$
	Initial Position	$\mathbf{q}_0 = [-5, 5]\ m$
	Depth	$z_T = -5\ m$
MPC	Weighting Parameters	$\rho_1 = 0.01\ ,\ \rho_2 = 0\ ,\ \rho_3 = 0$

Model A: The results of this simulation are depicted in fig. 6.1. Here, the target motion model from scenario A in section 3.2.1 was used to conduct the simulation. That is, in this simulation the target's velocity vector is known to the tracker, thus only the position has to be estimated. The EKF was feed with a wrong initial guess of the target position, $\hat{\mathbf{x}}_0 = [-20, 10]^T$. The uncertainty of the estimation of the position is represented in fig. 6.1a by the black ellipses that are plotted every 20s from $\hat{P}_{A,k}$



(a) Top view of the target (in red) and tracker (in blue) trajectories. The dashed green line represents the estimated position of the target and the black ellipsoids the uncertainty of the estimation

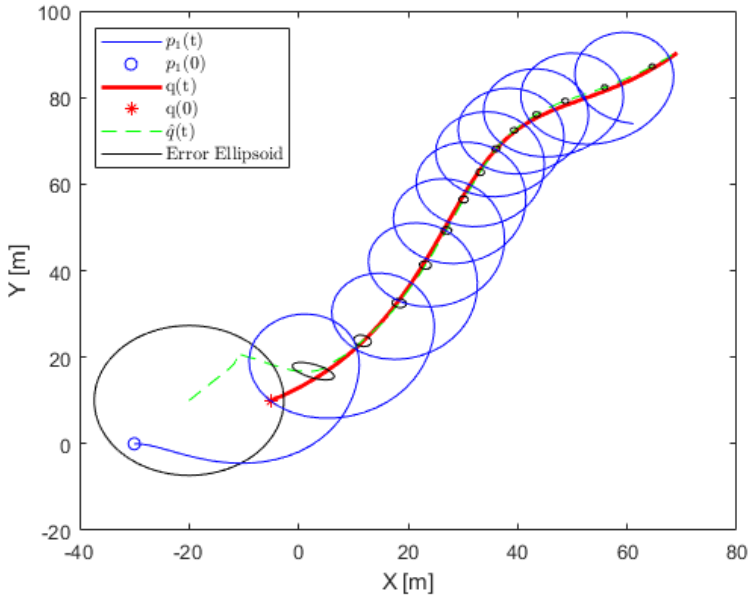


(b) Position estimated error

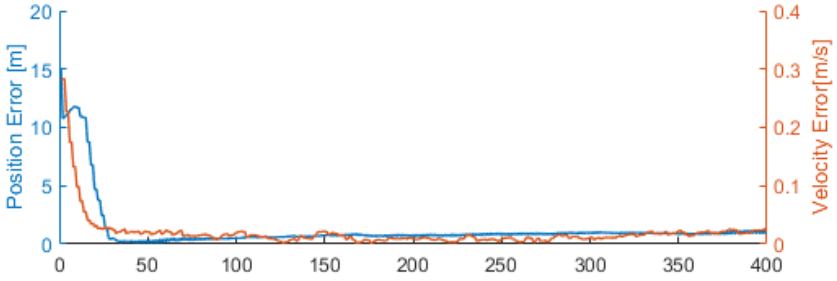
Figure 6.1: Single tracker single target simulation results (model A)

Model B: The results of this simulation are depicted in fig. 6.2. Here, the same simulation setup was repeated but this time the target motion model from scenario B in section 3.2.2 was used. That is, in this simulation the target's velocity vector is unknown to the tracker, thus it has to be estimated along with the position. The EKF was feed with a wrong initial guess of the target states, position and velocity,

$\hat{\mathbf{x}}_0 = [-20, 10, 0, 0]^T$. The uncertainty of the position estimation is represented in fig. 6.2a by the black ellipses that are plotted every 20s from $\hat{P}_{B,k}$.



(a) Top view of the target (in red) and tracker (in blue) trajectories. The dashed green line represents the estimated position of the target and the black ellipsoids the uncertainty of the estimation



(b) Position and velocity estimated errors

Figure 6.2: Single tracker single target simulation results (model B)

In this first simulation setup, a single tracker trying to localize a moving target with range measurements was considered. It can be seen from figures fig. 6.1b and fig. 6.2b that, for both scenarios A and B, the proposed strategy works well, since the position estimation error and the velocity estimation error converge to zero after a short period of time, less than a minute in both cases.

It can be seen in fig. 6.1a and fig. 6.2a that the tracker's trajectory is similar regardless of the target motion model used. Such trajectory corresponds to a circular movement around the target estimated position region, represented in the figures by the black ellipses. These results are in line with the ones presented in section 4.4.2.A, where it was shown that the tracker trajectory which generates more range

information about the target's position corresponds precisely to a circular trajectory around the position of the latter. Thus, it is proven that this MPC strategy is capable of generating trajectories for the tracker which approximate those that theoretically are optimal.

Also, in figures fig. 6.1a and fig. 6.2a it can be seen that by setting the weight of the pursuit cost, ρ_1, different from zero, the MPC strategy manages to keep the tracker at a short distance to the target, always less than $100m$.

6.2 Single Target - Two Trackers Results

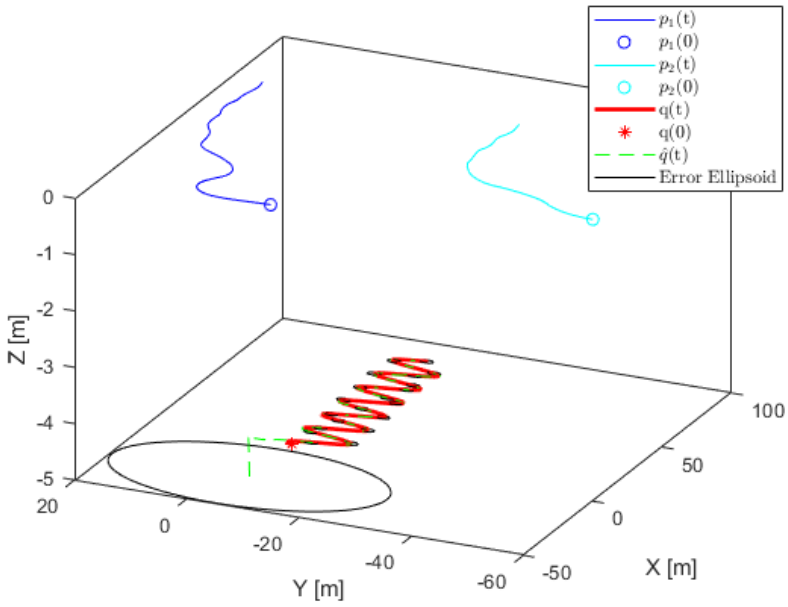
In this section, two trackers, $p = 2$, trying to localize a single moving target, $q = 1$, using range information were considered. In order to understand the limitations and assumptions made in each of the target models, A and B, defined in section 3.2, here a different target movement is considered in each simulation.

In both simulations the MPC weights associated with each cost in the minimization function are $\rho_1 = 0.01$, $\rho_2 = 1$ and $\rho_3 = 0$. Also, the costs related with the linear and angular velocities are given by $E_i = Diag(0.01, 0.01)$; for $i = \{1, 2\}$ in both simulations.

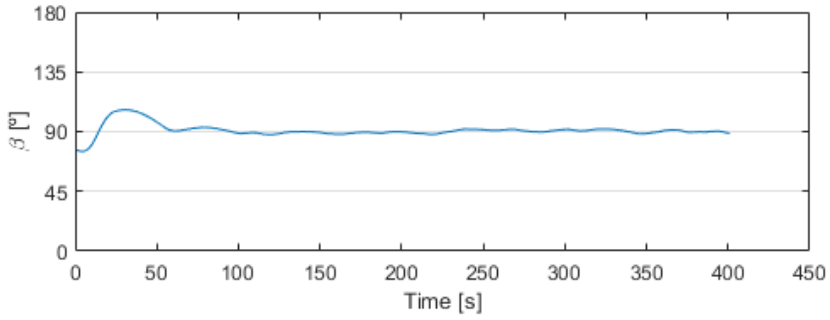
Model A: Here, the target motion model from scenario A in section 3.2.1 was used to conduct the simulation. That is, in this simulation the target's velocity vector is known to the trackers, thus only the position has to be estimated. The EKF was fed with a wrong initial guess of the target position, $\hat{\mathbf{x}}_0 = [-25, -5]^T$. The uncertainty of the estimation of the position is represented in fig. 6.3a by the black ellipses that are plotted every 20s from $\hat{P}_{A,k}$. The simulation parameters used for this simulation are given in table table 6.2.

Table 6.2: Single target two trackers simulation parameters (model A)

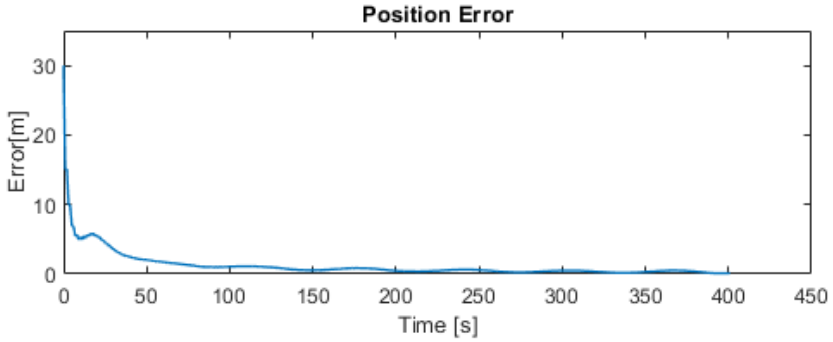
	Parameters	
Trackers	Initial Positions	$\mathbf{p}_0^{[1]} = [-30, -10]^T\ m\ ;\ \mathbf{p}_0^{[2]} = [0, -60]^T\ m$
Target	Velocity Vector	$\mathbf{u} = \begin{bmatrix} 0.2 \\ 0.5\sin(0.5x^q) \end{bmatrix} m/s$
	Initial Position	$\mathbf{q}_0 = [5, -5]^T\ m$
	Depth	$z_T = -5\ m$



(a) 3D-Trajectories of two trackers (blue and cyan) tracking a single target (red). The dashed green line represents the estimated position of the target and the black ellipsoids the uncertainty of the estimation



(b) Angle β between the two trackers and the target



(c) Target position estimation error

Figure 6.3: Two trackers single target simulation results (model A)

Model B: Here, the target motion model from scenario B in section 3.2.2 was used to conduct the simulation. That is, in this simulation the target's velocity vector is unknown to the trackers, thus both the

position and velocity must be estimated. The EKF was fed with a wrong initial guess of the target position and velocity, $\hat{\mathbf{x}}_0 = [-10, -5, 0, 0]^T$. The uncertainty of the estimation of the position is represented in fig. 6.4 by the black ellipses that are plotted every 20s from $\hat{P}_{B,k}$.

The simulation parameters used for this simulation are given in table table 6.3. Since one of the assumptions made when defining the target motion model B was that the target's velocity vector changes slowly, in this simulation the target cannot exhibit the same movement as in the previous simulation. Instead, the target moves slowly in order to estimate its states correctly.

Table 6.3: Single target two trackers simulation parameters (model B)

	Parameters	
Trackers	Initial Positions	$\mathbf{p}_0^{[1]} = [-30, -10]^T\ m\ ;\ \mathbf{p}_0^{[2]} = [0, -60]^T\ m$
Target	Velocity Vector	$\mathbf{u} = \begin{bmatrix} 0.2 + 0.1\cos(0.1x^q) \\ 0.2 + 0.1\sin(0.1x^q) \end{bmatrix} m/s$
	Initial Position	$\mathbf{q}_0 = [-5, 5]^T\ m$
	Depth	$z_T = -5\ m$

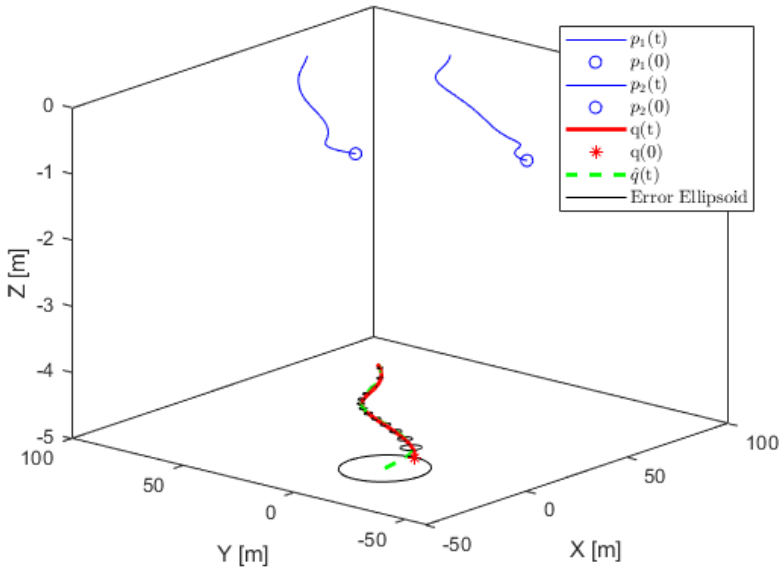
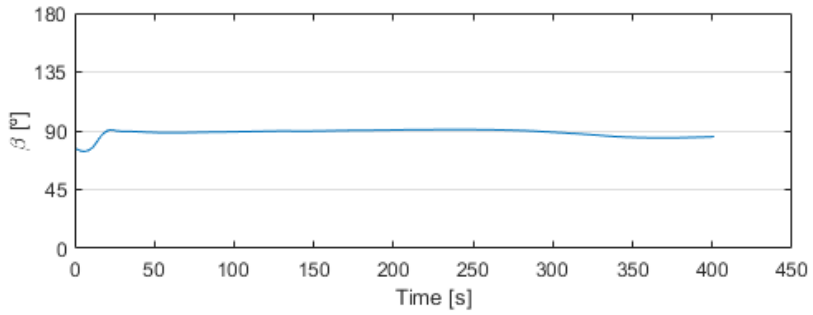
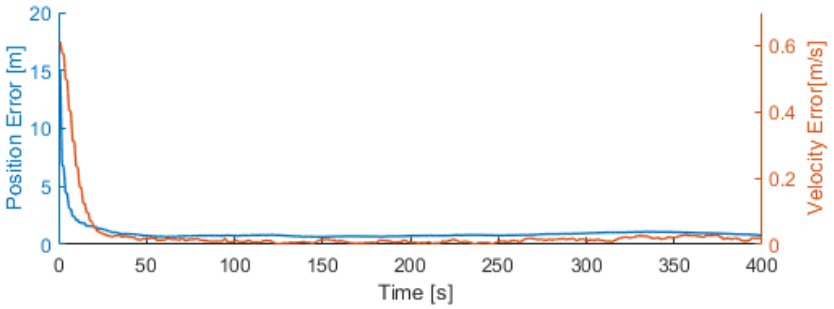


Figure 6.4: 3D-Trajectories of two trackers (blue and cyan) tracking a single target (red). The dashed green line represents the estimated position of the target and the black ellipsoids the uncertainty of the estimation



(a) Angle β between the two trackers and the target



(b) Position and velocity estimation errors

Figure 6.5: Two trackers single target simulation results (model B)

In this simulation setup two trackers trying to localize a moving target with range measurements were considered. It can be seen from fig. 6.3a and fig. 6.4 that, for both scenarios A and B, the proposed strategy works well, since the position estimation error and the velocity estimation error converge to zero after a short period of time, less than a minute in both cases.

From fig. 6.3b and fig. 6.5a it is possible to see that the angle between the relative position vectors from the trackers to the target, given by the angle β, converge to $90°$. These results are in line with the ones presented in section 4.4.2.B, where it was shown that in order to obtain the maximal range information an ideal trajectory for the trackers is to maintain the relative position vectors from them to the target orthogonal. Thus, it is proven that this MPC strategy is capable of generating trajectories for the trackers which approximate those that theoretically are optimal. Comparing these results with the ones from the single-target-single-tracker simulation setup, it can be noticed that in this case the trajectories generated are much less demanding in terms of manoeuvres.

Finally, in fig. 6.3a it can be seen that despite the oscillatory movement of the target, the trackers are able to pursuit the target by practically follow a straight line. This is one of the advantages of planning the trackers trajectories with MPC. By using an appropriate prediction window, in this case $N = 6$ was used, the trackers are able to anticipate the target position and thus moving accordingly to maximize the range information during the entire window. Also, it is important to notice that, as explained before,

an energy-saving cost was introduced in the optimisation problem, thus the optimizer tends to find a solution that compromises maximising the range information about the target and keep lower linear and angular velocities in the trackers.

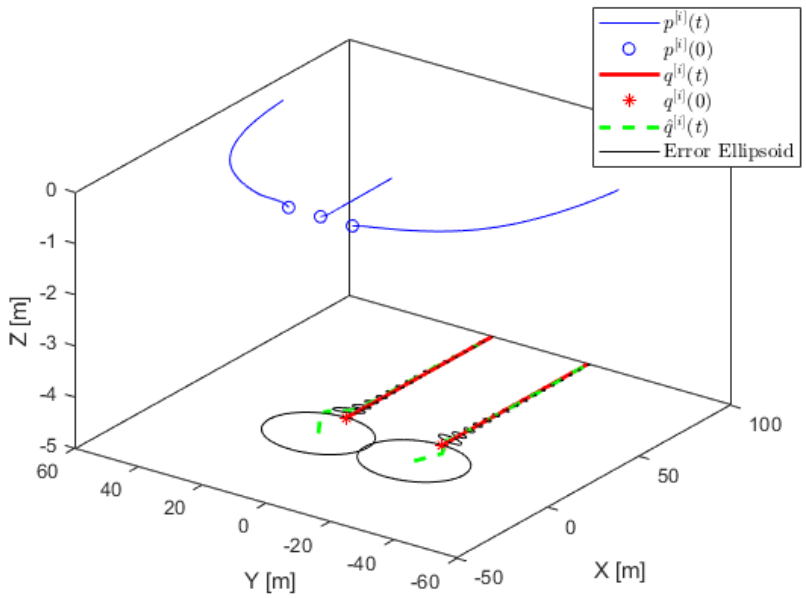
6.3 Multiple Targets Results

In this section, three trackers, $p = 3$, trying to localize two moving targets, $q = 2$, using range information were considered. In order to demonstrate the effects of assigning different weights to each target, the same simulation is conducted in two different cases: In the first, the weights associated with both targets are the same; In the second case, one of the targets is assigned with a larger weight, thus specifying more importance in tracking strategy. The goal is to analyse what are the differences in the trajectories produced for each target and if by associating a larger weight to one of the targets the determinant of correspondent Fisher Information Matrix, that encodes the quantity of the range information, is higher. Since in the two previous sections it was demonstrated that this strategy produces similar results whether the model A or the model B is considered, here, the simulations were conducted using only the model of scenario B. The simulation parameters that are common to both simulations are given in table 6.1.

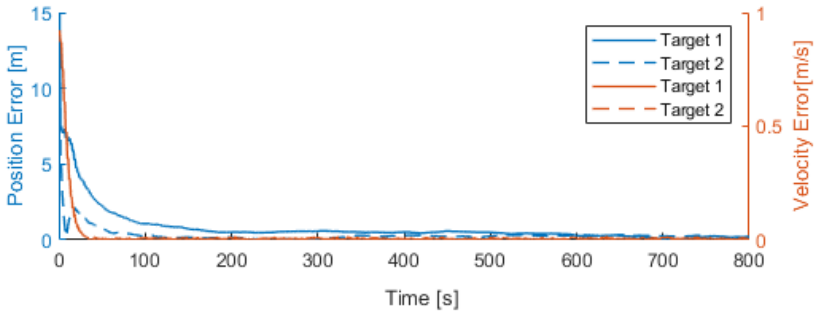
Table 6.4: Two targets three trackers simulation parameters

	Parameters	
Trackers	Initial Positions	$\mathbf{p}_0^{[1]} = [-20, 10]^T\ m$ $\mathbf{p}_0^{[2]} = [-20, 0]^T\ m$ $\mathbf{p}_0^{[3]} = [-20, -10]^T\ m$
Targets	Velocity Vector	$\mathbf{u}^{[j]} = \begin{bmatrix} 0.1 \\ 0 \end{bmatrix} m/s$; for $j \in \{1, 2\}$
	Initial Position	$\mathbf{q}_0^{[1]} = [20, 15]^T\ m$; $\mathbf{q}_0^{[2]} = [20, -15]^T\ m$
	Depth	$z_T^{[j]} = -5\ m$; for $j \in \{1, 2\}$

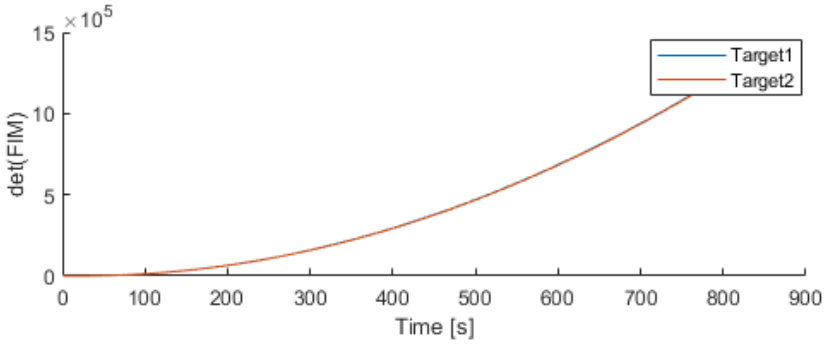
Same weights: In this simulation the weights associated with both targets are the same $\lambda_1 = \lambda_2 = 0.5$, thus both trackers are given the same importance in the tracking strategy. The target's velocity vectors are unknown to the trackers, thus they have to be estimated along with the positions. The EKF was fed with a wrong initial guess of the target states, position and velocity, $\hat{\mathbf{x}}_0^{[1]} = [5, 15, 0, 0]^T$ and $\hat{\mathbf{x}}_0^{[2]} = [5, -15, 0]^T$. The uncertainty of the positions estimation is represented in fig. 6.6a by the black ellipses that are plotted every 20s from $\hat{P}_{B,k}^{[i]}$, with $i \in \{1, 2\}$.



(a) 3D-Trajectories of three trackers $\mathbf{p}^{[i]},\ i \in \mathcal{S}$ (in blue) tracking two targets $\mathbf{q}^{[j]},\ j \in \mathcal{S}_T$ (in red). The initial positions of the trackers and targets are marked in with circles and stars respectivelly. The dashed green lines represent the estimated position of the targets and the black ellipsoids the uncertainty of the estimations.



(b) Position and velocity estimation errors



(c) Determinant of the FIM

Figure 6.6: Three trackers two targets simulation results (model B)

Different weights: In this simulation, in order to give more importance to one of the targets in the tracking strategy, different weights were associated with each target. To the first target is given a weight $\lambda_1 = 0.9$ and to the second one $\lambda_2 = 0.1$. The target's velocity vectors are unknown to the trackers, thus they have to be estimated along with the positions. The EKF was feed with a wrong initial guess of the target states, position and velocity, $\hat{\mathbf{x}}_0^{[1]} = [5, 15, 0, 0]^T$ and $\hat{\mathbf{x}}_0^{[2]} = [5, -15, 0]^T$. The uncertainty of the positions estimation is represented in fig. 6.6a by the black ellipses that are plotted every 20s from $\hat{P}_{B,k}^{[i]}$, with $i \in \{1, 2\}$.

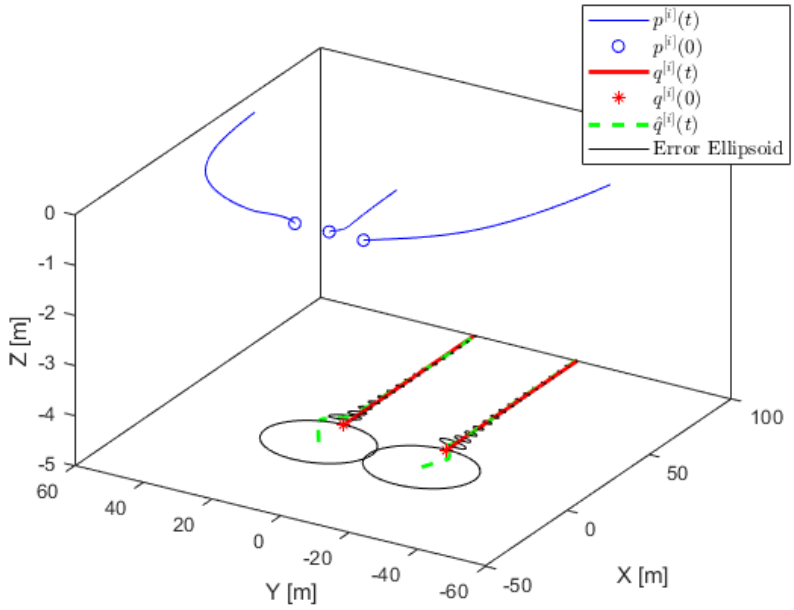
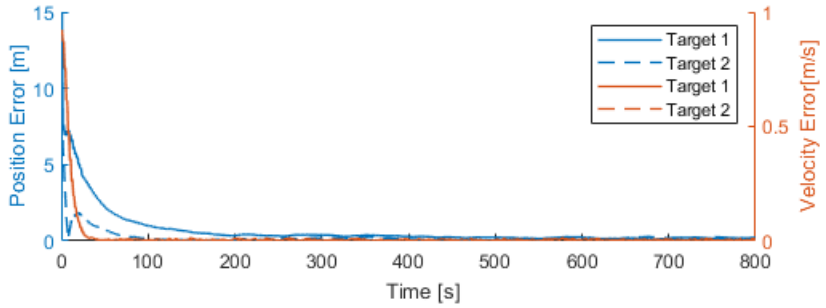


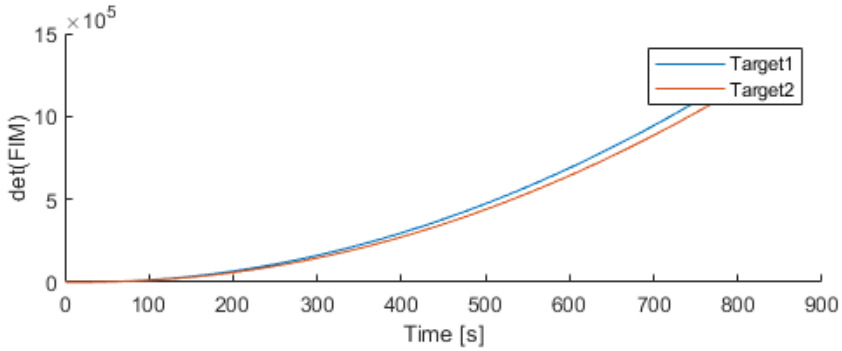
Figure 6.7: 3D-Trajectories of three trackers $\mathbf{p}^{[i]}$, $i \in \mathcal{S}$ (in blue) tracking two targets $\mathbf{q}^{[j]}$, $j \in \mathcal{S}_T$ (in red). The initial positions of the trackers and targets are marked in with circles and stars respectively. The dashed green lines represent the estimated position of the targets and the black ellipsoids the uncertainty of the estimations.

As in the previous scenarios, the proposed strategy is proven to produce good results since the position estimation error and the velocity estimation error converge to zero after a short period of time, as it is shown in fig. 6.6b and fig. 6.8a.

In section section 4.4.2.C it was discussed that an ideal geometry for the target tracking problem using multiple targets and multiple trackers is to maintain the trackers equally distributed in a circumference around the position of the targets. From the results depicted in figures fig. 6.6a it can be seen that after a short period of time the trackers move in such way that they tend to create a circumference around the projection of the targets estimated position in the surface $(z = 0)$.This shows that the strategy is again capable of generating trajectories which approximate the optimal ones for the target tracking problem. Here is important to notice that in the optimisation problem, besides the Fisher Information related cost, a cost that penalises the distance from the trackers to the targets was also introduced. For that reason,



(a) Position and velocity estimation errors



(b) Determinant of the FIM

Figure 6.8: Three trackers two targets simulation results (model B)

the trajectories produced are able to simultaneously maintain the trackers in the vicinity of the targets while maximising the range information about the target's position.

Finally, It can be seen from figure fig. 6.6a that when the same weight is given to each target, the trackers tend to be equally distributed around the position of each target, and by doing it a symmetric geometry between the targets and trackers positions is created. As the Fisher Information Matrix determinant is only dependent on the relative position between targets and trackers, from fig. 6.6c it can be seen that the determinant of the FIM is the same for both targets during the entire simulation as expected. When one of the targets is assigned with a larger weight than the other as it is in the second simulation, this symmetric geometry is no longer generated by the MPC strategy. From fig. 6.7, it can be seen that because target 1 is given a larger weight than target 2, that is $\lambda_1 = 0.9$ and $\lambda_2 = 0.1$, the trackers tend to approximate the target 1 (the one on the left in fig. 6.7), and to create a circumference around the projection of the target 1 estimated position in the surface $(z = 0)$. This movement leads to a geometry where more range information is collected about the position of the target with larger weight as it can be seen in fig. 6.8b, where the determinant of the FIM associated with the first target increases faster with time. Nevertheless, the estimates of the position and velocity converge quickly to the real value for both targets.

7

Conclusions

The main aim of this thesis work was to design and study a navigation system for range-based underwater target localization and tracking using single and multiple autonomous surface vehicles. In this setup, the targets operate underwater. As a result the trackers, that move in the surface, can only collect information about the target's position by taking range measurements to them. Therefore, with the available information, the ASV's must be able to simultaneously move in such way that they maximize the range information about the position of the targets while keep themselves in the vicinity of the AUV's.

Inspired by the results demonstrated in [20], in chapter 4, the ideal trajectories for the target tracking problem were discussed for a different number of targets and trackers. First, the Fisher Information Matrix was derived as an encoder of the quantity of information in the range measurements about the position of the targets. Then, using its determinant as a criterion to be maximized, ideal trajectories for a different number of surface trackers and underwater targets were discussed. These results are the baseline to compare with those to be obtained with the proposed MPC strategy, as they represent the best performance achievable for the localization task.

In chapter 5, the proposed MPC strategy to solve the underwater target localization and tracking problem was presented. Contrary to what was discussed in section 4.4.2, here, the MPC strategy is able to simultaneously find optimal trajectories for the trackers while respecting the constraints of the tracker's motion model, and estimating the actual position of the targets, instead of the latter being known without uncertainty. The determinant of the FIM defined in chapter 4 was again used as the criterion to be maximized. The strategy incorporates an optimal motion planing stage, a motion control stage based on the planned trajectories and an estimation stage which aims at produce estimates of the position of the targets. The EKF was used as state estimator and the Multiple Shooting method was used to solve the optimal problem that arises in the MPC strategy. The proposed strategy extends the one proposed in [20] with the possibility of assigning different weights to each target that capture the relative importance attached to each of the targets in terms of desired positioning accuracy.

Finally, in chapter 6 the proposed MPC strategy is tested in simulations in different scenarios. The simulations reveal promising results as they show that the main goal of the strategy, which is to localize underwater targets, can be achieved with surface vehicles. In addition, the simulations also show that the planned trajectories for the ASV's are identical to the ones that provide the best possible performance achievable for the target tracking problem. The MPC strategy reveals its versatility since it is capable of generating good results for the two different proposed target models even when the prior information about the target states is incorrect and there is a large uncertainty. The results also show the usefulness of the functionality of assigning different weights to different targets, showing that more range information is collected about the positions of the targets with larger associated weights.

The simulation results work as a proof of concept as they show good results in the underwater target localization and tracking task. However there are some extensions that can be made to the current work

before moving to a real system implementation.

7.1 Future Work

In this work, external disturbances such as ocean currents, noise sources on the actuators or the sensors were not considered. A straightforward extension of this work would be to consider a non-ideal scenario with these external influences [36]. The incorporation of other constraints such as obstacle avoidance and vehicle collision avoidance are other possible developments.

One of the assumptions made in this strategy is that the trackers can always measure ranges to all the targets. A second assumption is that the ranges are all available at the same time. Further research may focus on a filter design in order to add some relaxations to this requirements by dealing explicitly with such issues as asynchronous data acquisition and non-periodic sampling [16]. Furthermore, in order to obtain better estimates of the states of the targets, future work may consider, in addition to the range-only measurements, also range-rate measurements [37].

Another research path is to decentralising the MPC strategy in order to make it more extensible for scenarios with many trackers. Also, solving MPC resorting to shooting methods is time costly, and for multiple vehicles might no be able to run in real time. Collocation methods [38] might be investigated and state-of-the-art fast solvers [39] could be tested before moving to experimental operations.

Bibliography

[1] C. Kirches, Fast Numerical Methods for Mixed-Integer Nonlinear Model-Predictive Control. Vieweg, 2011.

[2] N. Crasta, D. Moreno-Salinas, B. Bayat, A. Pascoal, and J. Almansa, "Range-based underwater target localization using an autonomous surface vehicle: Observability analysis," in 2018 IEEE/ION Position, Location and Navigation Symposium (PLANS), 04 2018.

[3] J. F. Vasconcelos, C. Silvestre, P. Oliveira, P. Batista, and B. Cardeira, "Discrete time-varying attitude complementary filter," in 2009 American Control Conference, June 2009, pp. 4056–4061.

[4] A. Pascoal, I. Kaminer, and P. Oliveira, "Navigation system design using time-varying complementary filters," IEEE Transactions on Aerospace and Electronic Systems, vol. 36, no. 4, pp. 1099–1114, Oct 2000.

[5] J. M. Lemos, Controlo no Espaço de Estados. IST Press, 2019.

[6] M. I. Ribeiro, "Kalman and extended kalman filters: Concept, derivation and properties," 04 2004.

[7] H. L. V. Trees, Detection, Estimation, and Modulation Theory: Radar-Sonar Signal Processing and Gaussian Signals in Noise. Melbourne, FL, USA: Krieger Publishing Co., Inc., 1992.

[8] A. Alcocer, P. Oliveira, and A. Pascoal, "Study and implementation of an ekf gib-based underwater positioning system," IFAC Proceedings Volumes, vol. 37, no. 10, pp. 383 – 390, 2004, iFAC Conference on Computer Applications in Marine Systems - CAMS 2004, Ancona, Italy, 7-9 July 2004.

[9] T. Glotzbach, D. Moreno-Salinas, A. Pascoal, and J. Aranda, "Optimal sensor placement for acoustic range-based underwater robot positioning," IFAC Proceedings Volumes, vol. 46, no. 33, pp. 215 – 220, 2013, 9th IFAC Conference on Control Applications in Marine Systems.

[10] B. Porat and A. Nehorai, "Localizing vapor-emitting sources by moving sensors," in Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers, vol. 2, Oct 1994, pp. 765–769 vol.2.

[11] S. Martínez and F. Bullo, “Optimal sensor placement and motion coordination for target tracking,” Automatica, vol. 42, no. 4, pp. 661–668, 2006.

[12] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, Estimation with Applications to Tracking and Navigation. New York, NY, USA: John Wiley & Sons, Inc., 2002.

[13] M. Diehl, H. Bock, H. Diedam, and P.-B. Wieber, “Fast direct multiple shooting algorithms for optimal robot control,” Lecture Notes in Control and Information Sciences, vol. 340, 07 2007.

[14] J. T. Betts, Practical Methods for Optimal Control and Estimation Using Nonlinear Programming, 2nd ed. USA: Cambridge University Press, 2009.

[15] D. Moreno-Salinas, N. Crasta, M. Ribeiro, B. Bayat, A. Pascoal, and J. Aranda, “Integrated motion planning, control, and estimation for range-based marine vehicle positioning and target localization,” IFAC-PapersOnLine, vol. 49, no. 23, pp. 34 – 40, 2016, 10th IFAC Conference on Control Applications in Marine SystemsCAMS 2016.

[16] N. Crasta, D. Moreno-Salinas, A. Pascoal, and J. Almansa, “Multiple autonomous surface vehicle motion planning for cooperative range-based underwater target localization,” Annual Reviews in Control, vol. 46, 12 2018.

[17] N. Crasta, D. Moreno-Salinas, A. Pascoal, and J. Aranda, “Range-based cooperative underwater target localization,” IFAC-PapersOnLine, vol. 50, pp. 12 366–12 373, 07 2017.

[18] A. Alessandretti, A. P. Aguiar, and C. N. Jones, “Optimization based control for target estimation and tracking via highly observable trajectories,” in CONTROLO’2014 – Proceedings of the 11th Portuguese Conference on Automatic Control, A. P. Moreira, A. Matos, and G. Veiga, Eds. Cham: Springer International Publishing, 2015, pp. 495–504.

[19] R. Jain, A. Alessandretti, A. P. Aguiar, and J. Sousa, “A nonlinear model predictive control for an auv to track and estimate a moving target using range measurements,” in ROBOT, 2017.

[20] N. T. Hung, N. Crasta, D. Moreno-Salinas, A. M. Pascoal, and T. A. Johansen, “Range-based target localization and pursuit with autonomous vehicles: An approach using posterior crlb and model predictive control,” Robotics and Autonomous Systems, vol. 132, p. 103608, 2020.

[21] R. Praveen Jain, A. Pedro Aguiar, and J. B. d. Sousa, “Target tracking using an autonomous underwater vehicle: A moving path following approach,” in 2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV), 2018, pp. 1–6.

[22] R. P. Jain, A. P. Aguiar, A. Alessandretti, and J. Sousa, “Moving path following control of constrained underactuated vehicles: A nonlinear model predictive control approach,” in 2018 AIAA Information Systems-AIAA Infotech @ Aerospace, 01 2018.

[23] N. Crasta, M. Bayat, A. P. Aguiar, and A. M. Pascoal, "Observability analysis of 2d single beacon navigation in the presence of constant currents for two classes of maneuvers," IFAC Proceedings Volumes, vol. 46, no. 33, pp. 227 – 232, 2013, 9th IFAC Conference on Control Applications in Marine Systems.

[24] R. P. Jain, A. P. Aguiar, J. B. de Sousa, A. Zolich, T. A. Johansen, J. A. Alfredsen, E. Erstorp, and J. Kuttenkeuler, "Localization of an acoustic fish-tag using the time-of-arrival measurements: Preliminary results using exogenous kalman filter," in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018, pp. 1695–1702.

[25] H. Nguyen, F. Rego, N. Crasta, and A. Pascoal, "Input-constrained path following for autonomous marine vehicles with a global region of attraction," IFAC-PapersOnLine, vol. 51, pp. 348–353, 01 2018.

[26] N. T. Hung, A. M. Pascoal, and T. A. Johansen, "Cooperative path following of constrained autonomous vehicles with model predictive control and event-triggered communications," International Journal of Robust and Nonlinear Control, vol. 30, no. 7, pp. 2644–2670, 2020.

[27] D. Moreno-Salinas, A. Pascoal, and J. Aranda, "Optimal sensor placement for acoustic underwater target positioning with range-only measurements," IEEE Journal of Oceanic Engineering, vol. 41, no. 3, pp. 620–643, 2016.

[28] P. Tichavsky, C. H. Muravchik, and A. Nehorai, "Posterior cramer-rao bounds for discrete-time nonlinear filtering," IEEE Transactions on Signal Processing, vol. 46, no. 5, pp. 1386–1396, 1998.

[29] D. Moreno-Salinas, A. Pascoal, and J. Almansa, "Optimal sensor placement for multiple target positioning with range-only measurements in two-dimensional scenarios," Sensors (Basel, Switzerland), vol. 13, pp. 10 674–710, 08 2013.

[30] K. B. Howell, Principles of fourier analysis. Boca Raton, FL, USA: CRC Press, 2001.

[31] S. Thrun, W. Burgard, and D. Fox, Probabilistic robotics., ser. Intelligent robotics and autonomous agents. MIT Press, 2005.

[32] D. Moreno-Salinas, A. Pascoal, and J. Almansa, "Multiple underwater target positioning with optimally placed acoustic surface sensor networks," International Journal of Distributed Sensor Networks, vol. 14, p. 155014771877323, 05 2018.

[33] S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge University Press, 2004.

[34] D. Moreno-Salinas, A. Pascoal, and J. Aranda, "Optimal sensor placement for multiple underwater target localization with acoustic range measurements," IFAC Proceedings Volumes, vol. 44, no. 1, pp. 12 825 – 12 832, 2011, 18th IFAC World Congress.

[35] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, “CasADi – A software framework for nonlinear optimization and optimal control,” Mathematical Programming Computation, vol. 11, no. 1, pp. 1–36, 2019.

[36] M. Bayat, N. Crasta, A. P. Aguiar, and A. M. Pascoal, “Range-based underwater vehicle localization in the presence of unknown ocean currents: Theory and experiments,” IEEE Transactions on Control Systems Technology, vol. 24, no. 1, pp. 122–139, 2016.

[37] B. Ristic, S. Arulampalam, and J. McCarthy, “Target motion analysis using range-only measurements: algorithms, performance and application to isar data,” Signal Processing, vol. 82, no. 2, pp. 273 – 296, 2002.

[38] O. Von Stryk, “Numerical solution of optimal control problems by direct collocation,” Optimal Control Theory and Numerical Methods, vol. 111, 04 1998.

[39] B. Houska, H. J. Ferreau, and M. Diehl, “Acado toolkit—an open-source framework for automatic control and dynamic optimization,” Optimal Control Applications and Methods, vol. 32, no. 3, pp. 298–312, 2011.