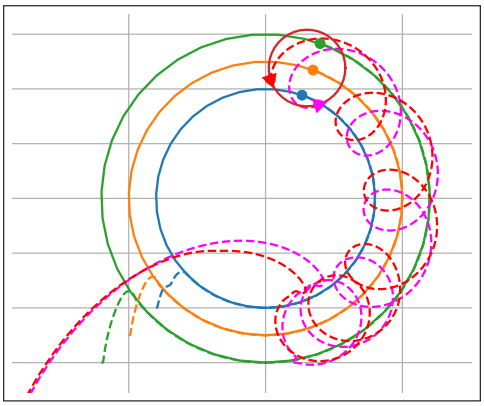
TÉCNICO LISBOA



Cooperative Marine Vehicle Navigation and Control with Applications to Geotechnical Surveying

Francisco dos Reis Centeio dos Santos Branco

Thesis to obtain the Master of Science Degree in

Electrical and Computer Engineering

Supervisor: Prof. António Manuel dos Santos Pascoal

Examination Committee

Chairperson: Prof. João Manuel de Freitas Xavier
Supervisor: Prof. António Manuel dos Santos Pascoal
Member of the Committee: Prof. David Alexandre Cabecinhas

June 2022

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 Practises of the Universidade de Lisboa.

Dedicated to my family, Alda, João and Valdemar...

Acknowledgements

As I am writing this, it is almost surreal how it all culminated to this moment. And what a journey it has been, these past years learning at *Instituto Superior Técnico*. I would like to especially thank my supervisor, Professor António Pascoal, for accompanying me in these last few years. Without his guidance and input, this work would have not been possible, and my interest in the ocean, its exploration and associated emerging technologies would not have been earnestly nurtured.

The great pillar supporting me throughout my years of existence is my family, specially my parents, Alda and João, who have always been there for me, bear the chaos of life with me, and endured my lunacy from time to time. They are the reason I have made it this far, and if i can be proud of myself it is because of them. To my grandfather, Valdemar, I dedicate this thesis to him, for nurturing my interest in electronics and my natural curiosity of wanting to know how things work. From a youngster who completed his correspondence electronics course, to one of the first electronic technicians at Philips in Portugal, to the greatest grandfather, he accompanied me through my journey almost until the end. My borrowed family, my godparents, Laura and Paulo, have also always supported me in this endeavour.

To my friends, I thank you all for the tremendous support during these trying times, specially my best friend, Pedro Quintas, who has consistently driven me towards the path of progress, keeping me in check if I ever slacked off or doubted myself, and my close musical group Camerata Appassionata, where the sense of camaraderie never leaves.

To my colleagues who have accompanied me within the DSOR Laboratory, Marcelo Jacinto, David Souto, Francisco Rego, João Quintas, André Potes and Hung Nguyen, I am extremely grateful for the time and effort you took to teach me and check all sorts of nooks and crannies of my work. I want to also thank the rest of the DSOR team, Luís Sebastião, Helena Santana, Manuel Rufino, João Botelho and Professor David Cabecinhas, for the great exposure to autonomous marine related topics, and the welcoming spirit presented from the first day I was introduced to the team.

Last but not least, I would like to thank the projects EU Marine Robotics Research Infrastructure Network [ID731103], and European Multidisciplinary Seafloor and Water Column Observatory (Portugal) [LISBOA-01-0145-FEDER-022157], for partially funding this work through Institute for Systems and Robotics (ISR).

Resumo

Os ambientes marinhos apresentam condições desafiantes para os humanos, dificultando a sua exploração minuciosa. O levantamento sísmico é uma das actividades realizadas no âmbito da exploração marinha, normalmente desempenhada recorrendo a grandes embarcações equipadas com fontes acústicas, rebocando redes massivas de cabos e receptores acústicos. Propõe-se a separação entre a fonte e os receptores acústicos, com recurso a uma rede de Veículos Autónomos Marinhos (AMV), permitindo recolher dados mais fiáveis face ao sistema convencional.

O conceito de Seguimento de Caminho (PF) serviu como unidade básica de construção de todo o sistema, seguido da cooperação entre veículos, introduzindo o conceito de Seguimento de Caminho Cooperativo (CPF). Considerando as limitações das comunicações dentro de água, um mecanismo de Comunicações Activadas por Eventos (ETC) foi concebido para colmatar esta questão. Assim que se alcança um grupo cooperativo de veículos, introduz-se a estratégia de Seguimento de Caminho Móvel (MPF), na qual um grupo de veículos segue um caminho circular fixo num alvo móvel.

O sistema final contempla um grupo de Veículos Autónomos Submersos (AUV) que executa a técnica de CPF sobre um caminho desejado, e um grupo de Veículos Autónomos de Superfície (ASV) que circula à volta do grupo de AUVs, recorrendo à abordagem de MPF. O Controlo Cooperativo de Múltiplas Formações (CMFC) é assim estabelecido entre os dois grupos de veículos, tendo em conta possíveis atrasos na coordenação geral entre veículos e formações. Uma aplicação de filtros também foi implementada, para prestar apoio à localização do AUV debaixo de água.

Palavras-chave: Veículos Autónomos Marinhos, Seguimento de Caminho Cooperativo, Caminho Móvel, Estimação de Veículos, Cooperação de Múltiplas Formações

Abstract

Marine environments bear challenging conditions for humans, making explorations of these areas very difficult. Seismic surveying is usually performed using large vessels equipped with sparkers, towing massive grids of hydrophones, or acoustic antennas. This work proposes a separation of the acoustic source and the antennas using a network of Autonomous Marine Vehicles (AMVs), where a system with multiple layers of cooperation allows for an enhanced capture of seismic data.

The concept of Path Following (PF) serves as the basic block of the entire system. The technique used lets a vehicle follow a certain path with a desired speed profile. Next, cooperation between vehicles is established, introducing Cooperative Path Following (CPF), enabling vehicle coordination between different paths, acquiring a certain desired formation. To overcome underwater communication constraints, an Event-Triggered Communications (ETC) mechanism was devised. Once a cooperative group of vehicles is attained, the Moving Path Following (MPF) strategy is introduced, where a group of vehicles follows a circular path fixed on a moving target.

The final system yields a group of Autonomous Underwater Vehicles (AUVs) which perform a simple CPF technique over a desired path, and a group of Autonomous Surface Vehicles (ASVs) which in turn perform the encircling manoeuvre around the group of AUVs, using the MPF approach. The Final layer of Cooperative Multiple Formation Control (CMFC) is formed between the two groups of vehicles, taking into account delays on the overall coordination. A filter application is also implemented to help the AUV underwater localisation.

Keywords: Autonomous Marine Vehicles, Cooperative Path Following, Moving Path, Vehicle Estimation, Cooperation of Multiple Formations

Contents

Resumo ix
Abstract xi
List of Tables xv
List of Figures xvii
Acronyms xix

1 Introduction **1**
1.1 Background 1
1.2 Problem Description and Objectives 2
1.3 State of the Art 4
1.4 Motivation 6
1.5 Main Contributions and Thesis Outline 7

2 Autonomous Marine Vehicle Model **9**
2.1 Notation and Reference Frames 9
2.2 Kinematics 11
2.3 Dynamics 11
2.4 Simplified Equations 13
2.5 MEDUSA-class Vehicles 14
2.6 System Implementation 16
2.6.1 Vehicle Model 16
2.6.2 Software Review 17

3 Path Following **19**
3.1 Historical Overview 19
3.1.1 Line-of-Sight for Straight-Line Paths 19
3.1.2 Line-of-Sight for Circular Paths 22
3.1.3 LOS Controller Definition 24
3.1.4 Simplified Line of Sight (P. Maurya) 25
3.1.5 Lyapunov-Oriented Path Following Control 27
3.2 Nonsingular Path Following Control 32
3.2.1 Kinematics 32

3.2.2 Nonlinear Controller Design . . . 35
3.2.3 Simulation Results . . . 36

4 Cooperative Path Following **38**
4.1 Cooperation with Continuous Communications . . . 39
4.1.1 Graph Theory . . . 39
4.1.2 Control Law Formulation . . . 39
4.1.3 Simulation Results . . . 41
4.2 Cooperation with Discrete Communications . . . 42
4.2.1 Event-Triggered Communications . . . 44
4.2.2 Simulation Results . . . 45

5 Moving Path Following **48**
5.1 Non-Rotating Path . . . 48
5.2 Rotating Path . . . 49
5.3 Simulation Results . . . 50
5.3.1 Simple Moving Path Following . . . 50
5.3.2 Cooperative Moving Path Following . . . 51

6 Filter Application **53**
6.1 Extended Kalman Filter . . . 54
6.1.1 Simulation Results . . . 55
6.2 Complementary Kalman Filter . . . 56
6.2.1 Simulation Results . . . 59

7 Cooperative Multiple Formation Control **62**
7.1 Smart Cooperative Path Following . . . 65
7.2 Simulation Results . . . 66
7.2.1 Cooperative Multiple Formation Algorithm . . . 66
7.2.2 Filter Application . . . 67

8 Conclusions **71**
8.1 Future Work . . . 72

List of Tables

2.1 Notation used for marine vehicles, adapted from Ribeiro [19]. 10
2.2 Model parameters for the MEDUSA-class vehicle, adapted from Sanches [20]. 15

3.1 Values for the parameters used in the PF manoeuvre. 37

4.1 Values for the parameters used in the ETC mechanism. 46

7.1 Path Following values. 68
7.2 Cooperative Path Following values. 69
7.3 Event-Triggered Communications values. 69
7.4 Cooperative Multiple Formation Control and Smart Cooperative Path Following values. . . 69

List of Figures

1.1 Relevant background projects. [2] 2
1.2 Vehicle Position plot of a simulation of the CMFC system. 4
1.3 Large ship (acoustic source) towing receiver streamers. [16] 6
1.4 Illustration of the proposed system. 7

2.1 Body-fixed and Inertial reference frames. [17]. 9
2.2 MEDUSA dimensions (in mm). [17] 14
2.3 MEDUSA vehicles in *Sesimbra*, after a full day of trials. [17] 16
2.4 Developed Dashboard for plotting analysis. 18

3.1 Straight line LOS. 20
3.2 Both Enclosure-based and Lookahead-based steering methods demonstrated. 22
3.3 Circular LOS Path Following. 23
3.4 Description of the simplified PF problem and variables of interest. 25
3.5 Control scheme for the explained simplified guidance method. [23] 27
3.6 Example of graphical representation of vehicle projection and variables of interest. [22] . 28
3.7 Vehicle parameters and frame definitions. [7] 33
3.8 PF block diagram for a single vehicle. 36
3.9 Plotting vehicle position and PF Error of the PF simulation. 37

4.1 Example of the type of coordination between vehicles required for the problem in question. 38
4.2 Example of three fully connected vehicle network topology represented in graph form. . . 40
4.3 CPF system block diagram. 41
4.4 Plotting simulation of vehicle continuous communications CPF. 42
4.5 Example of a network of vehicles and its neighbours estimates. [10] 44
4.6 CPF system block diagram with ETC mechanism. 46
4.7 Plotting simulation of vehicle discrete communications CPF with ETC mechanism. 47

5.1 Moving Path Following illustration for non-rotating path. 49
5.2 Moving Path Following illustration for rotating path. 49
5.3 MPF system block diagram. 51
5.4 Plotting vehicle position and PF Error of the simple MPF simulation. 51

5.5 Plotting the simulation results of the CMPF system. 52

6.1 Illustration of the filter application. 53
6.2 EKF included in the overall system represented in block diagram. 56
6.3 Plotting of the simulation results of the EKF implemented in the system. 57
6.4 Complementary Filter block diagram. 58
6.5 CKF included in the overall system represented in block diagram. 60
6.6 Plotting of the simulation results of the CKF implemented in the system. 61

7.1 Illustration of the CMFC system. 63
7.2 CMFC mechanism implemented on the overall system, represented in a block diagram. . 66
7.3 Plotting of the simulation for the CMFC system. 67
7.4 CMFC mechanism implemented on the overall system, represented in a block diagram. . 68
7.5 Plotting of the simulation for the CMFC system, with filter application. 70

Acronyms

AMV Autonomous Marine Vehicle. ix, xi, 1, 3, 5, 7, 8, 19, 32, 48

ASV Autonomous Surface Vehicle. ix, xi, 1, 3, 4, 6, 8, 17, 48, 51–54, 57, 59, 62–67, 69, 71, 72

AUV Autonomous Underwater Vehicle. ix, xi, 1–4, 6, 8, 17, 25, 48, 51–53, 57, 59, 62, 64–69, 71–73

CKF Complementary Kalman Filter. xviii, 8, 17, 53, 58–61, 72

CMFC Cooperative Multiple Formation Control. ix, xi, xvii, xviii, 4, 5, 7, 8, 17, 62–68, 70, 72

CMPF Cooperative Moving Path Following. xviii, 52

CPF Cooperative Path Following. ix, xi, xvii, 3, 5, 7, 8, 17, 38, 39, 41–43, 45–48, 62, 64, 65, 71, 73

DSOR Dynamical Systems and Ocean Robotics Laboratory. 1

DVL Doppler Velocity Log. 4, 5, 53, 57, 60, 72

EKF Extended Kalman Filter. xviii, 3–5, 8, 17, 53–57, 59, 72, 73

ETC Event-Triggered Communications. ix, xi, xv, xvii, 3, 8, 17, 44–48, 51, 62, 65, 71, 73

GPS Global Positioning System. 3

ISR Institute for Systems and Robotics. vii, 1, 9, 14

IST *Instituto Superior Técnico*. 9, 14

KF Kalman Filter. 53–55, 57, 58

LARSyS Laboratory of Robotics and Systems in Engineering and Science. 14

LOS Line-of-Sight. xvii, 19–23, 25

MORPH Marine Robotic System of Self-Organising, Logically Linked Physical Nodes. 1, 2

MPF Moving Path Following. ix, xi, xvii, 3–5, 7, 8, 17, 48–52, 56, 64, 71

PF Path Following. ix, xi, xv, xvii, 1, 3–5, 7, 8, 17, 19, 21, 24, 25, 27–30, 32, 33, 36–38, 41, 48, 51, 56, 63, 65, 71, 73

PID Proportional-Integral-Derivative. 24

SCPF Smart Cooperative Path Following. 65, 72

USBL Ultra-Short Baseline. 15

WiMUST Widely scalable Mobile Underwater Sonar Technology. 2

Chapter 1

Introduction

For the past few decades, the Autonomous Marine Vehicles (AMVs) scene has borne many technical feats [1], in the areas of Guidance, Navigation and Control. The WiMUST project [2] is a compelling example of the application of Path Following (PF) and cooperative motion control algorithms involving heterogeneous autonomous marine agents. Similarly within the challenge of seismic surveying, this thesis presents an approach to the problem of control and cooperation needed to realise a system capable of performing such missions. The proposed methods encompass control algorithms that ensure the convergence of vehicles to certain desired paths and cooperation between them, even if in different formations or groups. Ultimately, the aim of this work is to achieve a system of networked AMVs capable of enhancing the current resolution of surveys of the ocean floor, as well as attaining all the advantages observed from cooperation between AMV from previous projects in the same field.

The topics developed in this document bear many similarities with projects conducted at the Dynamical Systems and Ocean Robotics Laboratory (DSOR) of the Institute for Systems and Robotics (ISR), truly serving as an inspiration to this thesis.

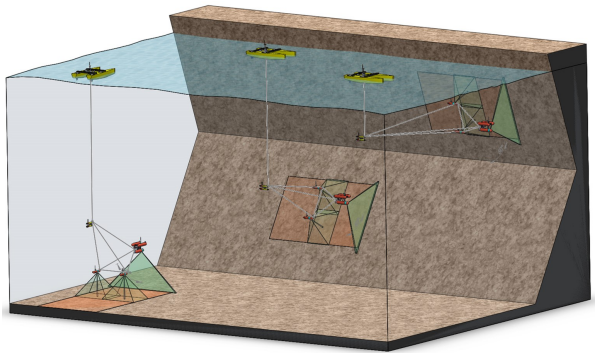
1.1 Background

As previously stated, the field of cooperation between AMVs has gathered quite an amount of investigation effort over the past few years [1]. The seabed and its resources are taking ever growing attention and, as a consequence, building up a strong motivation for the exploration of these uncharted territories. One of the pioneers in this area was the Marine Robotic System of Self-Organising, Logically Linked Physical Nodes (MORPH) project [3], which prompted the mapping of different deep ocean surfaces, including those with slopes, by deploying groups of Autonomous Underwater Vehicles (AUVs) and Autonomous Surface Vehicles (ASVs), working in cooperation and with different roles. In other words, each of the vehicles involved had a specific task at hand. The formation was composed of one ASV, to transmit GPS information to the underwater network of vehicles, thus helping the vehicle localisation problem, and four AUVs. There were two anchor vehicles, one of them leading the pack and the other one behind the formation, establishing a connection with the ASV, to rectify the overall underwater positioning of the

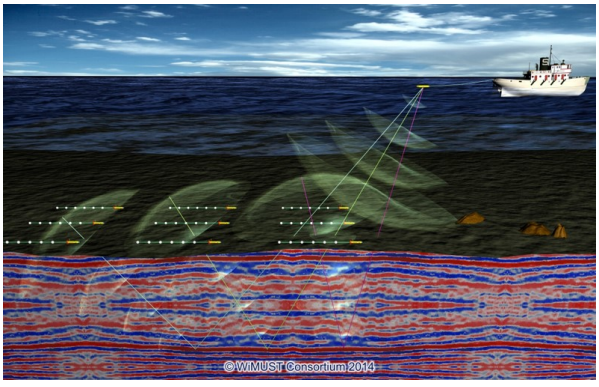
network as a whole. The last two AUVs had the job of actually mapping the seafloor, moving side by side. In Figure 1.1(a) it is possible to see the illustration of the MORPH vehicle formation. Each of the mapping nodes of the network had a camera for this purpose, creating detailed 3D imaging of the ocean floor, as well as vertical cliffs. This project led to more sophisticated methods for seabed mapping.

Another relevant work carried out in the field of Cooperative Autonomous Marine Vehicles for geotechnical surveys was the Widely scalable Mobile Underwater Sonar Technology (WiMUST) project [4]. One of the main motives behind this engineering achievement was the general amplification of the quality of seismic reflection surveys. This project was conceptualised as a flexible network of acoustic antennas, easy to set up while greatly reducing the cost of this type of missions. Up until this point, this type of surveying, for many different kinds of applications (windfarms, oil exploitation, mine exploration, etc.), was done using ships big enough to tow massive nets of streamers (cables equipped with hydrophones). In this context, the towing ship is equipped with acoustic sources called sparkers that emit acoustic waves which, upon reflection and refraction in the different layers of the seabed, are picked up by the streamers. However, this mission format presents great disadvantages, since the support ship and streamers are extremely expensive and difficult to deploy.

Consequently, the WiMUST project recorded impressive results, in the sense that it was possible to gather more and better data from these surveys, thanks to the decoupling of the source and the streamers. The AUVs would follow the acoustic source, along a desired path, towing a line of streamers. The conjugation of the various AUVs made it possible to change the shape of the acoustic underwater antenna during the mission, in addition to approximating the streamers to the seafloor. All these factors allowed for a better overall resolution obtained with the seismic reflection surveys. In Figure 1.1(b) it is possible to observe the illustration of the WiMUST project concept and vehicle distribution during a representative mission.



(a) MORPH project (2012-2016).



(b) WiMUST project (2015-2018).

Figure 1.1: Relevant background projects. [2]

1.2 Problem Description and Objectives

Cooperative Marine Vehicle Control has turned into a very attractive topic of research since it has been the inspiration for many projects within both the academy and industry environments. When it comes to

the use of AMVs for geotechnical surveys, the typical formations of deployed cooperative vehicles are purposefully designed in order to increase the overall accuracy and precision of the survey. However, the approach developed in this thesis brings a new perspective to the existing standards. The end goal is to have ASVs circulating the submerged centre of mass of a group of AUVs (supposedly towing acoustic antennas or, in other words, streamers), and following them in this circular motion. The premise to justify this logic comes from geophysical studies around the field of seismic reflection surveying. By dividing the ocean floor in cells, it was noticed that when each cell was hit by the acoustic wave from different angles, the accuracy of the mapping was substantially increased, making it possible to derive an acoustic impedance from each cell at different depths [5]. Following this line of thought, by having multiple ASVs acting as acoustic sources while circulating the AUVs carrying the streamers, hitting the ocean floor with acoustic waves, the latter can pick up the waves that ricochet of each cell at different depths, gaining an overall better resolution on the mapping of the seabed layers.

After describing the general problem, one must become familiar with the technical aspects of the problem. The project is divided in many sub-problems to simplify its development. The first issue to be addressed is Path Following (PF), the core functionality of the system under study, where the orientation of the vehicle is controlled so that it converges to a certain path. One could think of Trajectory Tracking instead, although these techniques not only are not necessary, but also complicate the development of the desired system, since time constraints are not a requirement. Another disadvantage of Trajectory Tracking, which requires a certain vehicle to be at a desired place at a desired time, is that the resulting motions may exhibit considerable ”excitation” and be quite demanding in terms of energy and actuator precision and coordination required to perform it. Besides, PF generally yields far smoother vehicle motions.

Secondly, there is the problem of Cooperative Path Following (CPF), where AMVs cooperate to converge to a path, reducing their speed if they are too ahead and, similarly, increasing it if they are too far behind. This allows the vehicles to move as a group or, to put it another way, a formation. The idea behind this method is quite simple, layering perfectly on top of what has been developed up until this point. The problem arises when one takes into account the discrete nature of these transmissions to establish cooperation between AMVs, as well as the low-bandwidth of the water column for acoustic communications when compared, for example, to the usual electromagnetic communications. This is where Event-Triggered Communications (ETC) come into play, addressing both these issues and equipping the AMVs with a way to communicate more scarcely, preventing jammed transmissions and unnecessary flow of messages between vehicles or, in other words, nodes.

Since part of the scope of this project is to have the ASVs performing a circular motion around the target AUVs, introducing the method known as Moving Path Following (MPF) was a natural step. The system now yields two groups of vehicles, a group of tracker ASVs that execute MPF on a circular path around the target group of AUVs. Each formation encompasses CPF within that specific formation, ensuring cooperation, just as mentioned before.

To ensure more accurate AUV localisation, considering that there is no GPS information while underwater, the tracker ASVs perform an estimation of the target AUVs’ position using an Extended Kalman

Filter (EKF) and successive range-measurements over the submerged vehicles. This data is then transmitted to the AUVs which in turn blend it with the velocity readings from the Doppler Velocity Log (DVL) sensor.

As of this stage, the system is able to operate as a target pursuit between two formations, estimating the group of AUVs while using it as the centre of the circular path defined during the MPF. However, as stipulated before, the aim of this work is to have full cooperation between formations, taking into account if one formation is ahead of the other, or even if there is a malfunction during the mission. Therefore, the final step is to assemble the Cooperative Multiple Formation Control (CMFC), so that all the ASVs and AUVs move together in synchronism, slowing down or speeding up in reaction to the behaviour of the neighbouring vehicles and formations. The plot in Figure 1.2 shows the general idea behind the CMFC, illustrating a simulation where the technique is performed.

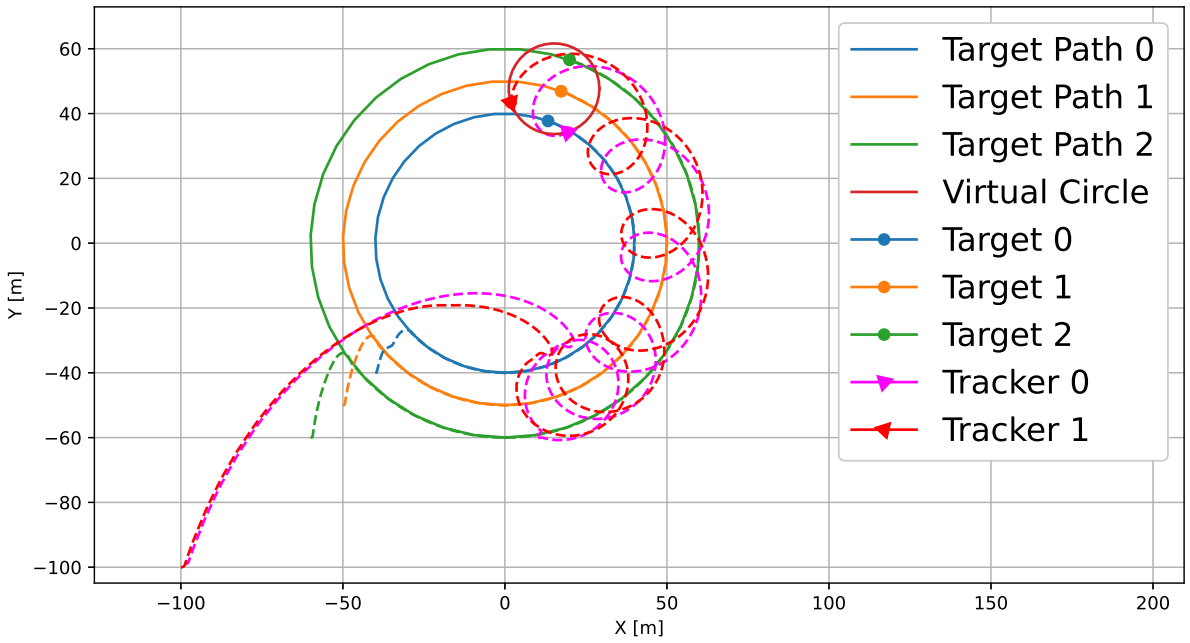


Figure 1.2: Vehicle Position plot of a simulation of the CMFC system.

1.3 State of the Art

One of the major influences for this thesis was the work accomplished in Teixeira [6], and as a result the objective of this thesis did not grow distant from it. Similarly, the end goal was to have separate formations of vehicles acting together, although the approaches taken during the design process and the final system derived from them were fairly different, as explained later on.

The main concept that enacts as the backbone of this system is the PF. It is the basic method from which the final iteration of the system was created. It is then natural that the right algorithm is carefully chosen in order to have a well designed control structure over the vehicles.

After some thought and review, the algorithm formulated in Lapierre et al. [7] was the right fit for the problem at hand, for its global convergence properties and accessible path parameter variable, which

indicates the current position of the vehicle on its assigned path. This becomes very useful during the next design stages, for example, the implementation of the CPF algorithm, meticulously documented in Ghabcheloo et al. [8]. While the PF technique generates a control law for the yaw rate of the vehicle, the CPF remains completely decoupled, generating in turn a velocity law that makes up for vehicles too ahead or behind relative to the general state of the group of vehicles, with respect to the path they are following.

The necessary use of discrete communications forces the evolution from the continuous case defined beforehand to a more compliant system. A functional discrete network of vehicles is presented in Vanni [9] along with a control law more considerate of the actual requirements. The transition to the final step regarding vehicle inter-communication is then even smoother, clearly stated in Rego et al. [10]. Two threshold functions (one of them time dependent and the other state dependent) were designed as a means to determine when it is pertinent to share messages between vehicles, which were introduced in Hung et al. [11] along with proof of convergence to the desired formation. Each vehicle holds estimates for the coordination state of their neighbours, as well as their own, so that it is possible to determine when the error between its estimation of itself and the real coordination state is bigger than the threshold. If this condition holds true, the vehicle broadcasts its state and estimates to its neighbours.

The MPF strategy was built upon the approach adopted in Teixeira [6]; this made sense since the approach was already implemented in the logic of AMVs. The original idea came from the well exposed concept presented in Oliveira et al. [12], where an MPF law was devised for an unmanned aircraft. A translation and rotation are applied on the regularly generated PF law, comprising the movement of the moving path, thus successfully performing the desired circulating manoeuvre around a target.

At this point, the target's state is unknown, so the process described earlier is unrealistic. To address this issue, a target pursuit is implemented using a position estimation from an EKF, which is a classic method to solve the problem. This technique can later be replaced with the cooperation between different formations proposed in the CMFC. Nonetheless, as seen in Hung et al. [13], taking range measurements as a sensor input, the collected data can then be used to update the current estimated position. This idea can be taken even further by sending the information gathered with the EKF to the target, facilitating vehicle underwater localisation, blending the target estimation from the EKF with the DVL sensor data. However, the approach to this data fusion was inspired by the complementary filter architecture proposed in Pascoal et al. [14], greatly simplifying the assign of inputs/outputs of the system. Using this set-up, the DVL sensor data is dealt as an input and the EKF estimation as a sensor measurement with a certain variance matrix, working just like a standard Kalman Filter method as seen in Ribeiro [15].

The final stage of CMFC was also influenced by the work in Teixeira [6] and the general idea behind CPF, as an effort to create an elegant solution for the problem of cooperation between different vehicle formations, which might already have a level of cooperation between vehicles within the formation, and taking in consideration the aforementioned underwater communication constraints.

1.4 Motivation

Since the great nautical explorers started venturing through the oceans, there has been an ever-lasting curiosity about these seemingly remote places. Unbeknownst to mankind, there was another world to discover down under in the oceans' depths.

For the past few years, these almost uncharted territories have attracted widespread attention because of the opportunities they represent due to their natural resources. However, exploring these areas can be quite the challenge, since direct human intervention is nearly impossible due to the increasing pressure of the water column as the depth increases, along with other harsh environmental conditions. To carry out a thorough study of what lies beyond the ocean's surface, it is useful to create tools and methodologies based on what current technology can support.

One very useful study method in this field is the monitoring and mapping of the ocean floor. There are many different approaches to this problem, as described before. The most refined method that has been used throughout the years is the towing of massive nets of streamers which receive the acoustic waves emitted by a source, usually the ship that tows the said streamers, as seen in Figure 1.3. This method is categorised as a seismic reflection survey, where the reflection of acoustic waves on the various layers of the ocean floor makes it possible to determine its acoustic impedance, thus having a better understanding of the constitution of the ocean floor and beneath it.

Figure 1.3: Large ship (acoustic source) towing receiver streamers. [16]

The Cooperation of Autonomous Marine Vehicles comes as an alternative to this traditional method, with many advantages. Firstly, there is the decoupling of the streamer from the acoustic source, giving more freedom to manipulate the acoustic antenna relative to the ocean floor and the acoustic source (Figure 1.1(b)). Secondly, the costs and time duration of each mission are greatly decreased when this new implementation is used. In the end, there are more different types and better quality of data gathered, overall, implying a better seismic reflection survey resolution. For the acoustic source, either a leading vessel or ASVs can be used, while AUVs tow acoustic streamers.

The reason why this technology is worth investing in is the wide possible application spectrum in current times. From a scientific point of view, this can mean a better understanding of the seafloor from

its various substrate layers to the resident marine life which populates its extension. From an industrial point of view, there are many and extremely challenging topics in the areas of offshore wind farms, oil, gas and exploration of mineral resources, that justify the various investments in the field of study of Cooperative AMVs for geotechnical surveying. These topics are quite relevant at the moment, specially with the emergence of the Blue Economy.

This project specifically targets the amplification of the resolution and general quality of seismic reflection surveys. From a geophysical standpoint, it was deduced that whilst mapping [5], dividing the ocean floor in different cells and firing acoustic waves at them, which were generated from one or more sources at different angles, it is possible to obtain better results for these types of surveys. Thus a system where the sources move around (that is, encircle) the underwater vehicles moving as a group, performing CPF dictated by a given reference path, was proposed, as can be observed in Figure 1.4.

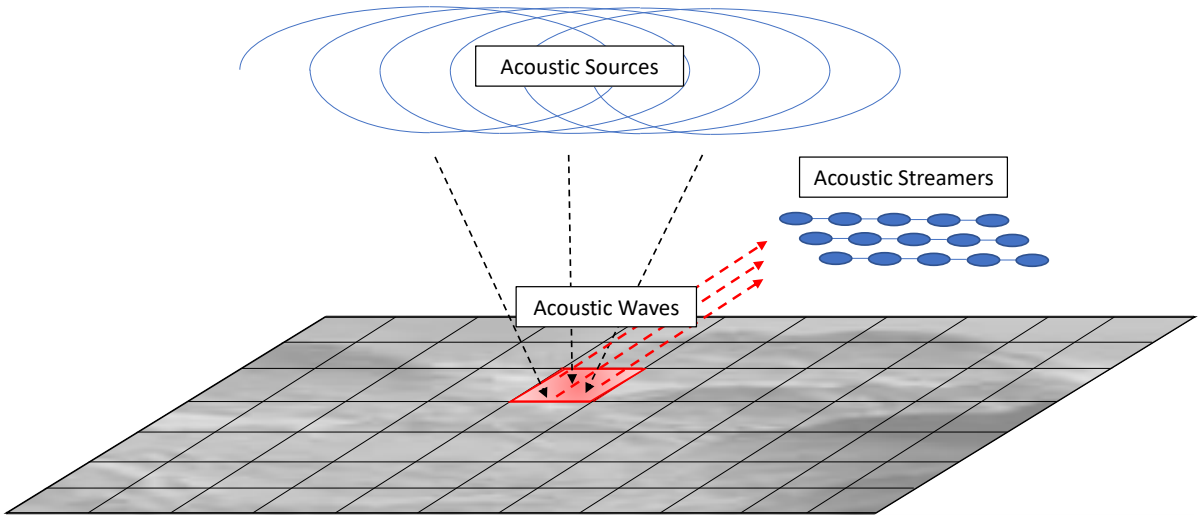


Figure 1.4: Illustration of the proposed system.

1.5 Main Contributions and Thesis Outline

Throughout this thesis, a relevant in-depth analysis of the concepts addressed in the design and development of the desired system is presented, resorting to both theory and intuition when exposing the different subjects. The PF, CPF and MPF concepts, although not new by any means, have been diligently explored in order to make the development of Chapter 7 possible. Here, the novel control mechanism CMFC is explained, conceived to integrate a cooperation layer between different vehicle formations in the context of the motivation behind this project. A new technique was also created regarding the PF and CPF, to care for a better performance of both these algorithms when working together. Another important implemented feature is an ad-hoc filter strategy to take into account the underwater positioning and localisation.

Considering all this, the key contributions of this thesis can then listed as:

- The study and implementation of an integrated system for Cooperative Multiple Formation Control

(CMFC) of marine vehicles under stringent positioning and communication constraints.

- The development of a realistic simulator to evaluate the performance of the designed system; this was made possible with a complex and well fractioned code stack, where each concept of subsystem is developed in different modules to facilitate the performance analysis and the overall implementation.

In light of this, the document is structured in eight chapters, following the reasoning presented in section 1.2. Chapter 2 deals with the model of AMV the work was based on, along with the adaptations made to facilitate the system analysis. Following this, Chapter 3 addresses the PF problem, with an overview of previously used algorithms, and a comprehensive description of the algorithm that was actually applied to the problem at hand. With Chapter 4, the CPF problem is engaged, looking upon the basic time-continuous algorithm and also moving to more real-like implementations, taking into account discrete communications, more precisely ETC.

In Chapter 5, the MPF problem is explained, and the elegant solution for a control law while having a moving path. Here, more than one group of vehicles starts to be considered, getting closer to the final envisioned system. Chapter 6 introduces two filter ideas to help approximate the system to the real case, performing target localisation within the ASVs, with the help of an EKF, and subsequently sharing and blending the information with the AUVs' Complementary Kalman Filter (CKF).

Finally, in Chapter 7, the novelty idea of CMFC is proposed, for a group of ASVs and a separate group of AUVs in the context of seismic reflection surveys for seafloor mapping and exploration. The last but not least, Chapter 8 presents the outcome of this thesis and remarks on what can be done in the future to further study and develop this type of systems.

Chapter 2

Autonomous Marine Vehicle Model

In this section, a detailed explanation of the mathematical model of a representative autonomous marine vehicle is given, specially regarding kinematics (Section 2.2) and dynamics (Section 2.3) of the vehicle, as well as a suited adaptation to the MEDUSA-class type of vehicle in Section 2.5, developed at Institute for Systems and Robotics (ISR) of the *Instituto Superior Técnico* (IST) of Lisbon, Portugal. The notation, the equations and the reasoning are borrowed from the paper of Ribeiro et al. [17], and a deeper understanding of the marine vehicle model is available in Fossen [18, chapter 2].

2.1 Notation and Reference Frames

Firstly, the base of the notation must be defined: the Earth-fixed inertial frame $\{U\}$, composed by the orthogonal axes x_U, y_U, z_U, and the Body-fixed frame $\{B\}$, composed by the axes $\{x_B, y_B, z_B\}$, as shown in Figure 2.1:

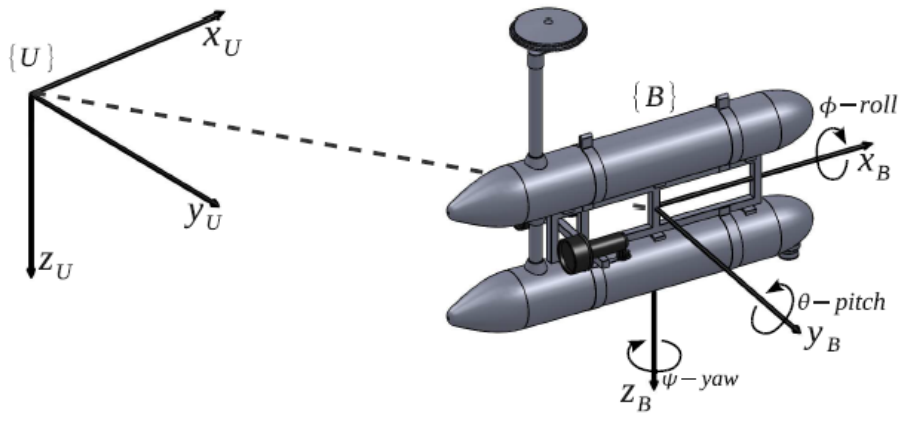


Figure 2.1: Body-fixed and Inertial reference frames. [17].

The inertial frame follows the North-East-Down (NED) convention, denoting the North and East di-

rections for x_U and y_U, respectively and z_U pointing towards the centre of the Earth.

On the other hand, the body axes are defined as x_B being the longitudinal axis (directed from the stern of the vehicle to fore), y_B being the transversal axis (directed from port to starboard) and z_B being the normal axis (directed from top to bottom). The origin of the body-fixed frame $\{B\}$ usually coincides with the centre of mass of the vehicle. The subsequent motion control of said vehicle, in other words, of $\{B\}$, is described relative to the inertial frame $\{U\}$.

Since the normal implementation of these types of vehicles yield six degrees of freedom (to move in the ocean water column), the evolution of the position and orientation of the vehicle is given by six independent coordinates, three position coordinates (x, y, z) and orientation given by three Euler angles (ϕ, θ, ψ). The motion components are defined as *surge*, *sway*, *heave*, *roll*, *pitch* and *yaw*, respectively. This means that the three motion components are relative to the variation of each position coordinate of the vehicle and the last three are relative to each orientation angle coordinate of the vehicle. We adopt the SNAME[1] notation, defined as

- $\eta_1 = [x, y, z]^T$: position of the origin of $\{B\}$ expressed in $\{U\}$, North (x), East (y), Down (z),
- $\eta_2 = [\phi, \theta, \psi]^T$: orientation of $\{B\}$ with respect to $\{U\}$, Roll (ϕ), Pitch (θ), Yaw (ψ),
- $\nu_1 = [u, v, w]^T$: linear velocity of the origin of $\{B\}$ relative to $\{U\}$, expressed in $\{B\}$,
- $\nu_2 = [p, q, r]^T$: angular velocity of $\{B\}$ relative to $\{U\}$, expressed in $\{B\}$,
- $\tau_1 = [X, Y, Z]^T$: actuating forces expressed in $\{B\}$,
- $\tau_2 = [K, M, N]^T$: actuating moments expressed in $\{B\}$.

The previous list and Table 2.1 are deducted from Ribeiro [19].

Table 2.1: Notation used for marine vehicles, adapted from Ribeiro [19].

Degree of Freedom	Forces and moments	Linear and angular velocities	Position and Euler angles
1. Motions in the x-direction (*surge*)	X	u	x
2. Motions in the y-direction (*sway*)	Y	v	y
3. Motions in the z-direction (*heave*)	Z	w	z
4. Rotation about the x-axis (roll)	K	p	ϕ
5. Rotation about the y-axis (pitch)	M	q	θ
6. Rotation about the z-axis (yaw)	N	r	ψ

In compact form, the following generalised vectors are derived from the previous list and Table 2.1, and given as

$$\eta = [\eta_1^T, \eta_2^T]^T \,, \tag{2.1}$$

$$\nu = [\nu_1^T, \nu_2^T]^T \,, \tag{2.2}$$

$$\tau_{RB} = [\tau_1^T, \tau_2^T]^T \,. \tag{2.3}$$

[1]The Society of Naval Architects & Marine Engineers - http://www.sname.org/

2.2 Kinematics

Since the kinematics deal only with the geometrical aspects of motion while relating the velocities variables with the position variables, the kinematics equations, according to the notation established in Section 2.1, can be expressed as

$$\dot{\eta} = J(\eta)\nu\,, \tag{2.4}$$

where

$$J(\eta) = \begin{bmatrix} {}^{U}_{B}R(\Theta) & 0_{3\times3} \\ 0_{3\times3} & T_{\Theta}(\Theta) \end{bmatrix} \tag{2.5}$$

and

$${}^{U}_{B}R(\Theta) = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi s\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix}, \quad s\cdot = sin(\cdot), \quad c\cdot = cos(\cdot)\,, \tag{2.6}$$

is the rotation matrix from the body-fixed frame $\{B\}$ to the inertial frame $\{U\}$. The latter is defined on the basis of three successive rotations according to the zyx-convention, that is,

$${}^{U}_{B}R(\Theta) := R_{z,\psi}R_{y,\theta}R_{x,\phi}\,. \tag{2.7}$$

In what follows,

$$T_{\Theta}(\Theta) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & \frac{s\phi}{c\theta} & \frac{c\phi}{c\theta} \end{bmatrix}, \quad t\cdot = tan(\cdot), \quad \theta \neq \pm 90^{\circ}\,, \tag{2.8}$$

is the Euler attitude transformation matrix that relates the body-fixed angular velocities (p, q, r) with the roll ($\dot{\phi}$), pitch ($\dot{\theta}$) and yaw ($\dot{\psi}$) rates. Since there is a singularity for $\theta = \pm 90^{\circ}$ in Equation 2.8, another implementation of the system can be done by using the quaternion representation, as seen in Fossen [18], with a more in-depth approach of this topic. However, due to physical constraints of the real model of the vehicle, the angles for pitch and roll will operate far from this singularity ($\theta \approx 0^{\circ}$ and $\phi \approx 0^{\circ}$), thus having the opportunity to use the above Euler representation.

2.3 Dynamics

In order to derive the equations of dynamics of the model, one may resort to Newton's law, and since the hydrodynamic forces and moments are generated by the relative motion between the body and the fluid, the natural representation of these variables is on the body-fixed frame $\{B\}$. Therefore, the rigid-body equation can be formulated as

$$M_{RB}\dot{\nu} + C_{RB}(\nu)\nu = \tau_{RB}\,, \tag{2.9}$$

where M_{RB} is the rigid-body inertia matrix, C_{RB} represents the Coriolis and centrifugal terms and τ_{RB} is a generalised vector of external forces and moments, decomposed as

$$\tau_{RB} = \tau + \tau_A + \tau_D + \tau_R + \tau_{dist}\,. \tag{2.10}$$

The term τ represents a vector of forces and torques related to thrusters/surfaces, usually for propulsion, and can be viewed as the control input. The term τ_A represents the vector of forces and moments due to the hydrodynamic added mass, and can be described as

$$\tau_A = -M_A\dot{\nu} - C_A(\nu)\nu\,, \tag{2.11}$$

where M_A is the added inertia matrix and $C_A(\nu)$ is the hydrodynamic added Coriolis and centripetal contribution matrix.

The term τ_D represents the vector of hydrodynamic due to lift, drag, skin friction, etc., and can be formulated as

$$\tau_D = -D(\nu)\nu\,, \tag{2.12}$$

where $D(\nu)$ denotes the hydrodynamic damping matrix, which must be positive definite. The term τ_R represents the vector of restoring forces and torques due to gravity and fluid density, and can be formulated as

$$\tau_R = -g(\eta)\,, \tag{2.13}$$

where $g(\eta)$ represents a vector dependent on the variables of η.

The final term τ_{dist} represents all the environmental and external forces, in other words, external disturbances (from waves, wind, etc.). Manipulating the previous equations, it is possible to group the various terms of the Equation 2.9, so that the dynamics equation is formulated as

$$\begin{aligned} \underbrace{M_{RB}\dot{\nu} + C_{RB}(\nu)\nu}_{\text{rigid-body terms}} + \underbrace{M_A\dot{\nu} + C_A(\nu)\nu + D(\nu)\nu}_{\text{hydrodynamic}} + \underbrace{g(\eta)}_{\text{restoring term}} &= \tau + \tau_{dist}\,, \\ M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + g(\eta) &= \tau + \tau_{dist}\,, \end{aligned} \tag{2.14}$$

where

$$M = M_{RB} + M_A\,, \tag{2.15}$$

is the combination of the rigid-body inertia matrix and the added inertia matrix, and

$$C(\nu) = C_{RB}(\nu) + C_A(\nu)\,, \tag{2.16}$$

is the combination of the Coriolis and centripetal terms and the hydrodynamic added Coriolis and centripetal contribution matrix.

2.4 Simplified Equations

In this section the equations which capture the motions of the vehicle are deduced based on the previous Sections 2.2 and 2.3. Since the scope of the project is operating the vehicles in the horizontal plane, there are only three degrees of freedom $[x, y, \psi]$, and the kinematics equations take the simple form

$$\begin{aligned} \dot{x} &= u\ cos(\psi) - v\ sin(\psi)\ , \\ \dot{y} &= u\ sin(\psi) - v\ cos(\psi)\ , \\ \dot{\psi} &= r\ . \end{aligned} \tag{2.17}$$

To obtain the external force in *surge* and the external torque to make the vehicle turn around the z-axis (*yaw*), the following equations were derived

$$\begin{aligned} \tau_u &= F_{sb} + F_{ps}, \\ \tau_r &= l \cdot (F_{ps} - F_{sb}), \end{aligned} \tag{2.18}$$

where the F_{sb} is the starboard thruster force, the F_{ps} is the port-side thruster force, and l is the length of the arm attached to the thrusters with respect to the centre of mass of the vehicle. The term τ_u corresponds to the common mode, where the two thrusters work together in *surge* motion, and the term τ_r corresponds to the differential mode, where the two thrusters work against each other, creating a rotating motion for the vehicle around the z-axis. Since the vehicle moves on the horizontal plane, it is possible to neglect roll, pitch and heave motions, thus formulating the equations for the velocity components of the vehicle (u, v, r), while not considering disturbances, given by

$$\begin{aligned} m_u\dot{u} - m_v vr + d_u u &= \tau_u, \\ m_v\dot{v} + m_u ur + d_v v &= 0, \\ m_r\dot{r} - m_{uv} uv + d_r r &= \tau_r, \end{aligned} \tag{2.19}$$

where

$$\begin{aligned} m_u &= m - X_{\dot{u}} \quad d_u = -X_u - X_{|u|u}|u|\ , \\ m_v &= m - Y_{\dot{v}} \quad d_v = -Y_v - Y_{|v|v}|v|\ , \\ m_r &= I_z - N_{\dot{r}} \quad d_r = -N_r - N_{|r|r}|r|\ , \\ m_{uv} &= m_u - m_v\ . \end{aligned} \tag{2.20}$$

The terms m_u, m_v, m_r and m_{uv} are mass and hydrodynamic added mass, while the terms d_u, d_v and d_r represent hydrodynamic damping effects.

If ocean currents are considered, some changes to the previous equations have to be made. For a constant irrotational ocean current $[v_c, u_c]^T$, forming an angle ψ_c with respect to the inertial frame, the kinematics become $v = v_r + v_c$ and $u = u_r + u_c$, where u_r and v_r are the components of the vehicle velocity with respect to the current and u_c and v_c are the components of the ocean current velocity in

the body-fixed frame. The new equations are then derived from the previous dynamic Equations 2.19, as defined as

$$\begin{aligned} m_u\dot{u} - m_v(v_r + v_c)r + d_u u_r &= \tau_u \,, \\ m_v\dot{v} + m_u(u_r + u_c)r + d_v v_r &= 0 \,, \\ m_r\dot{r} - m_{uv}(u_r + u_c)\cdot(v_r + v_c) + d_r r &= \tau_r \,, \end{aligned} \tag{2.21}$$

where

$$\begin{aligned} m_u &= m - X_{\dot{u}} \quad d_u = -X_u - X_{|u|u}|u_r| \,, \\ m_v &= m - Y_{\dot{v}} \quad d_v = -Y_v - Y_{|v|v}|v_r| \,, \\ m_r &= I_z - N_{\dot{r}} \quad d_r = -N_r - N_{|r|r}|r| \,, \\ m_{uv} &= m_u - m_v \,. \end{aligned} \tag{2.22}$$

Notice that the *sway* motion in not controllable, thus simplifying $\tau_v = 0$. Finally, it is important to denote that the restoring force and moment vector $g(\eta)$ is not relevant for the formulation of the previous equations, since the vehicle is neutrally buoyant and the centre of gravity and the centre of buoyancy are located vertically on the z-axis of the body-fixed frame $\{B\}$.

2.5 MEDUSA-class Vehicles

In this section, the MEDUSA-class of vehicles is discussed, so that the previous model is adapted to this type of vehicles and subsequent testing is made within the scope of the project.

The MEDUSA are autonomous marine robotic vehicles developed at the Laboratory of Robotics and Systems in Engineering and Science (LARSyS)/ISR of the IST of Lisbon, Portugal.

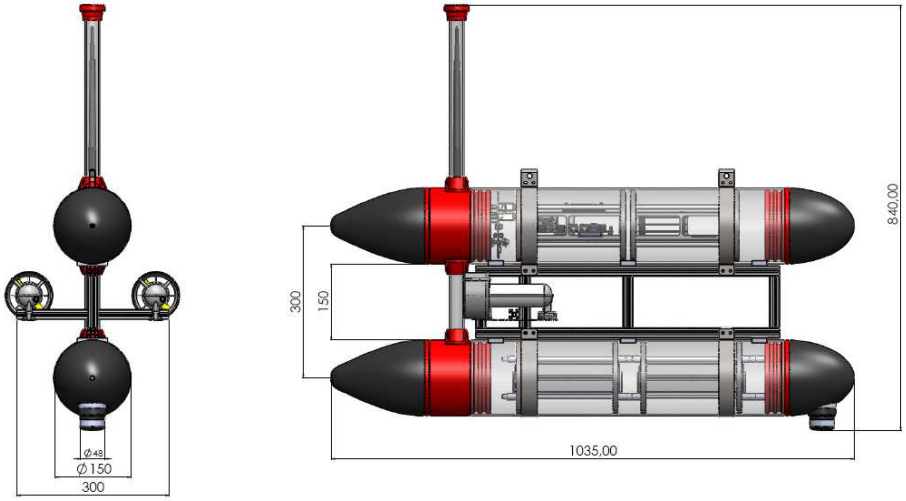


Figure 2.2: MEDUSA dimensions (in mm). [17]

Each vehicle of this class, as a whole, weighs approximately $17\ kg$ (current model, the version

described in Ribeiro [19] weighed $30\ kg$) and consists of two acrylic housing tubes of size $0.15\ m$ by $1.035\ m$ ($diameter \times length$) with aluminium end caps. These two tubes are attached to a central aluminium frame, this way composing a rigid structure, and setting the distance between the two tubes as $0.15\ m$ (Figure 2.2). The tubes align vertically, as seen in Figure 2.2, and the lower tube contains the two packs of 7-cell lithium polymer batteries together with thruster electronics, an underwater camera, an acoustic modem and other minor sensors. The upper body of the MEDUSA contains the main computer unit together with navigation sensors and GPS, for the case of surface vehicles. However, these vehicles are highly modular, making it possible to mount different hardware modules on them, depending on the type of mission. A technology which proves to be very useful in these types of missions is USBL positioning to estimate the pose of the vehicle using acoustic communications. Inter-vehicle communications are enabled using predominantly an acoustic modem network, and possibly via Wi-Fi between different surface vehicles.

There are two stern thrusters, one on starboard and another on port, which control directly the *surge* and yaw motions. As mentioned before, these thrusters can be used in common mode, to control the *surge* motion, or in differential mode, to control the yaw motion around the z-axis. To formulate these equations more precisely, it is possible to redefine these motions, from Equation 2.18, as the following torques

$$\begin{aligned} \tau_u &= F_s + F_p\,, \quad \text{for the common mode,} \\ \tau_r &= l \cdot (F_s - F_p)\,, \quad \text{for the differential mode,} \end{aligned} \tag{2.23}$$

where F_s is the starboard relative force, F_p is the port relative force and $l = 0.15\ m$ is the length of the arm which connects the thrusters to the MEDUSA vehicle. The diving/submersible MEDUSA-class vehicles have additional thrusters for the *heave* motion to control depth.

As denoted before, in the scope of this project roll, pitch and *heave* motions do not need to be actuated, as the buoyancy of the vehicle guarantees a large enough metacentric height.

The vehicle model parameters were already attained in Ribeiro [19]. The updated parameters are shown in Table 2.2.

Table 2.2: Model parameters for the MEDUSA-class vehicle, adapted from Sanches [20].

X		Y		N	
$X_{\dot{u}}$	$-20\ kg$	$Y_{\dot{v}}$	$-30\ kg$	$N_{\dot{r}}$	$-8.69\ kg.m^2$
X_u	$-0.2\ kg/s$	Y_v	$-50\ kg/s$	N_r	$-4.14\ kg.m/s$
$X_{\|u\|u}$	$-25\ kg/m$	$Y_{\|v\|v}$	$0.01\ kg/m$	$N_{\|r\|r}$	$-6.23\ kg.m$

These are the most recent values for the current MEDUSA model. They can be found in Sanches [20], based on previous model identification efforts. In light of this information, these parameters must be identified for each MEDUSA-class vehicle physical modification, where some parameters can be easily

Figure 2.3: MEDUSA vehicles in *Sesimbra*, after a full day of trials. [17]

attained directly from simple proportional and difference calculations, while others must be identified from successive experiments in controlled environments.

Now that these values are available, they can be used in Equations 2.21 and 2.22 in order to implement the mathematical model on the real system.

2.6 System Implementation

This section gives some insight on the vehicle model used, along with the actual implemented system to execute the simulations.

2.6.1 Vehicle Model

The system in question is implemented solely on a kinematics level, in order to evaluate the control algorithms at their core. The following model is then defined as

$$\begin{cases} \dot{x} &= v \cdot \cos\theta\,, \\ \dot{y} &= v \cdot \sin\theta\,, \\ \dot{\theta} &= \omega\,, \end{cases} \tag{2.24}$$

where x, y is the vehicle's position, and therefore $\dot{x}, \dot{y}$ is the derivative of the position with respect to time. The velocity is given by v, which is controlled by velocity laws, as it will be seen in later chapters, and θ is the yaw angle of the vehicle. Note that in the previous sections in Chapter 2 the variable v denoted

the sway motion of the vehicle and u the actual surge. To streamline the notation used throughout this thesis, even more when considering the other modules from the concepts of, for instance, PF and CPF, the variable v is considered to be the surge motion of the vehicle throughout this thesis. The yaw control is given by ω, which is a control input to be designed, more specifically by the PF. Another matter worth noting is the absence of sideslip, in other words, the sideslip is assumed $\beta = 0$ throughout the development of this work.

2.6.2 Software Review

The simulations are done in a Python 3 virtual environment, in a specially developed stack, which can be found in [21]. The main structure of the code is composed of three files and a library folder. The *pathgeneration* module generates all sorts of paths with known curvature. The *utils* module assists the user with various tools for the different concepts used in the implementation. The *main* module selects which simulation the user wants to run, presenting the option of either running a previously saved simulation or executing a new one. The library folder *lib* contains all the modules of the addressed concepts in this thesis, along with the folders relative to each simulation. Within each simulation folder, a *run* module is created with the initial conditions and other parameters specific to the simulation, where a system defined in the *systembuild* module is executed, and evaluated by the user through a series of plots setup in the corresponding *plot* module. To this end, a dashboard was developed in order to better understand the simulations (Figure 2.4), plotting all the variables of interest. The rest of the modules are relative to the implemented concepts of the vehicle kinematics model, the PF, the CPF, the MPF, the filter application (EKF and CKF) and the CMFC.

A brief description of Figure 2.4 is presented next, where some of the functionalities of the developed simulation environment are described, which will be discussed in detail later. A simulation of one thousand seconds is shown, contemplating the final system described in Section 1.2 of CMFC between a group of three AUV targets and a group of two ASV trackers. Six different plots are presented, where the top left one is related to the position of the ASV trackers and the AUV targets. The top right one presents the computed control output for the angle rate of the kinematic model. The Velocity plot shows the velocity of each vehicle at every time instance, generated by control laws, and systematically fed to the kinematic model. The ETC Broadcasting plot demonstrates the moments when each vehicle broadcasts its coordination state within the AUV network and the ASV network. The Vehicle Path Progression plot shows the coordination states of all vehicles and how they achieve coordination. The Cooperative Formation Control plot displays the coordination states of the elected target, in this case, the AUV Target 1, and the virtual circle coordination state, clearly shown in the Position plot, and it's clearly observed that coordination is also reached. Notice that once the coordination states γ reach the value of one, it loops around to zero, since the path in question is a circle that is completely followed through more than once.

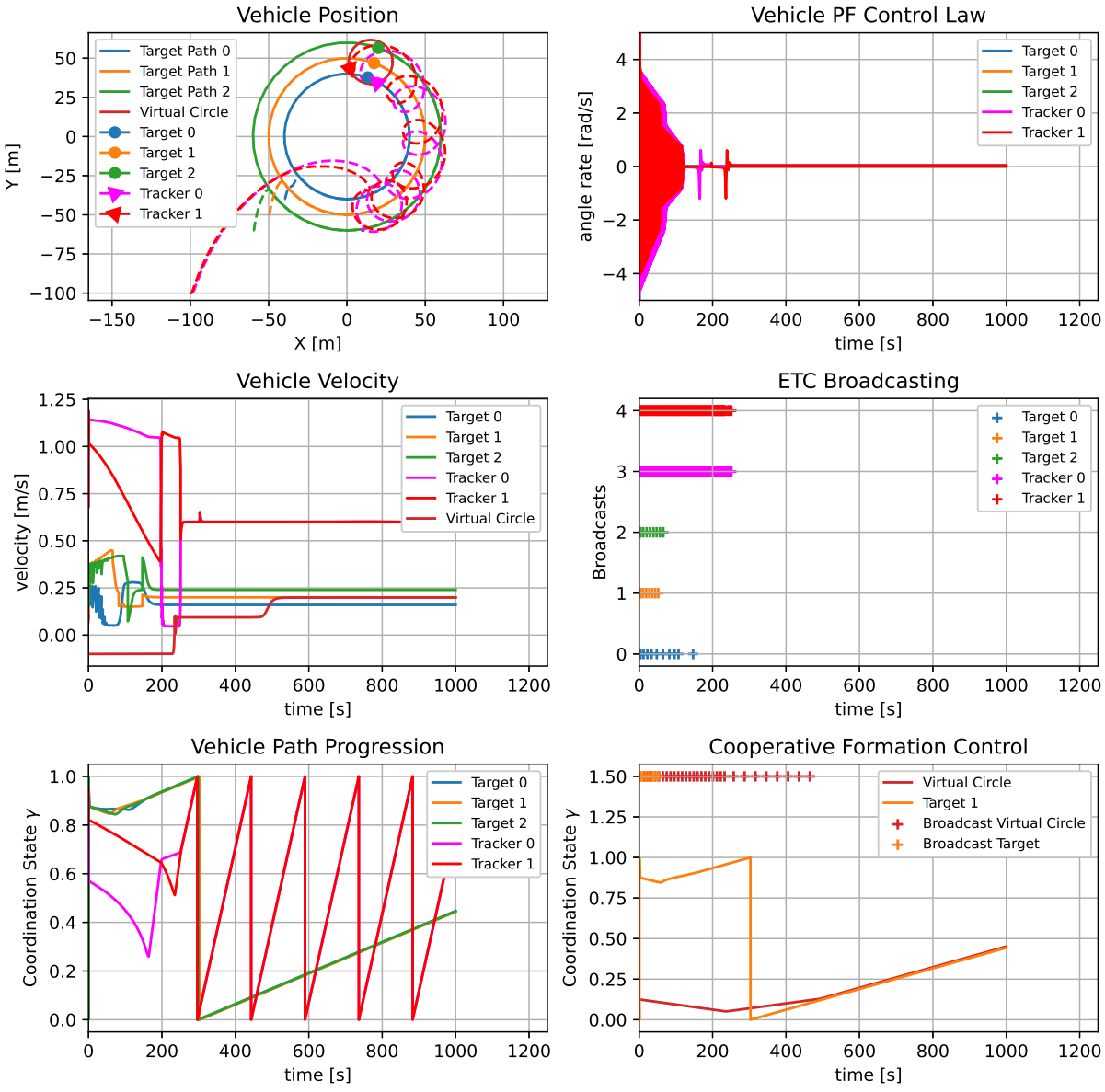


Figure 2.4: Developed Dashboard for plotting analysis.

Chapter 3

Path Following

In the context of the system developed throughout this thesis, the Path Following (PF) acts as the fundamental building block. A fair assumption within the description in Section 1.2 is to consider the problem in 2D, since the control of the vehicle in the vertical axis is out of the scope of this project. Following this line of thought, the PF allows a vehicle to follow a path chosen beforehand, unencumbered by time constraints, as opposed to Trajectory Tracking techniques. In this section a review of the most fundamental PF algorithms for AMVs is presented, more precisely, the Line-of-Sight (LOS) algorithm for straight lines, circular paths and a simplified useful version of this algorithm. After analysing these algorithms, it is possible to move on to more robust methods, such as the one discussed in Micaelli et al. [22] for a new approach to the parameterisation of the path in space and the subsequent convergence of the vehicle to the path. Finally, it is shown an analysis of the methods described in Lapierre et al. [7] for another interesting method to yield convergence of the vehicle to the desired path, projecting the position of the vehicle on the path at different points of said path according to the needs of the system, while solving some issues found in the previous algorithm.

3.1 Historical Overview

LOS is one of, if not the most notable vehicle PF algorithm, due to its widespread use and general applicability to many different scenarios. It has been continuously employed and refined over time, thus proving to be a useful tool for autonomous vehicles' engineers, even when applying other more complex algorithms. The LOS makes for an excellent starting point for understanding PF at its core, as it is observed in the next sections.

3.1.1 Line-of-Sight for Straight-Line Paths

In a nutshell, the LOS computes the desired steering angle of the vehicle based on a measure of distance selected *a priori*, to a certain target or position in sight. In other words, the LOS algorithm targets a point on the path using a desired measurement method, as seen in Figure 3.1. As a means to achieve this goal, two different methods will be analysed, the **Enclosure-based steering** and **Lookahead-based**

steering, as it was formulated in Fossen [18, chapter 10]. This approach is waypoint based, though it is possible to implement these methods without specific waypoints, instead with auxiliary points.

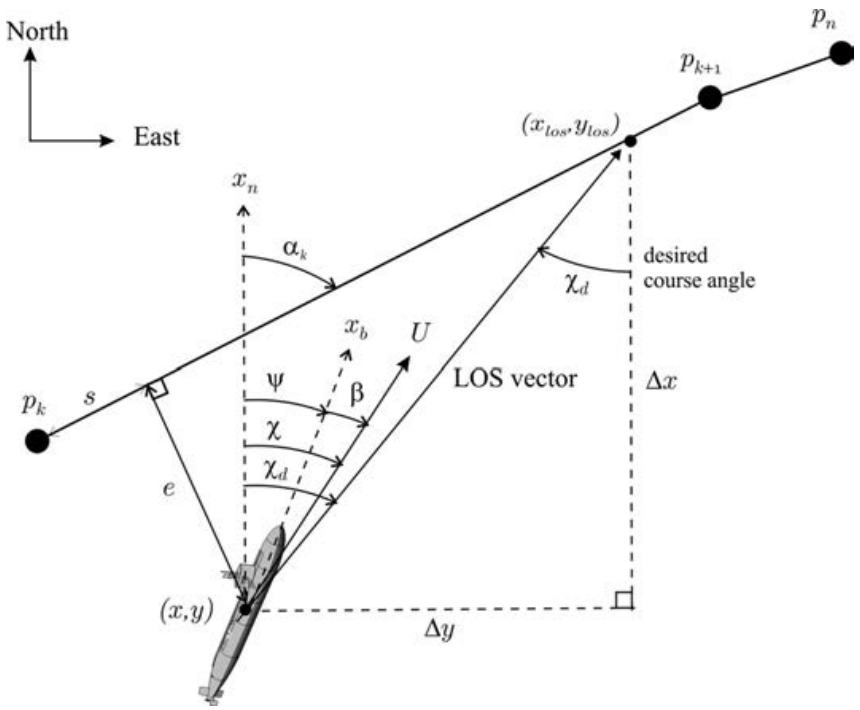


Figure 3.1: Straight line LOS. [18]

Enclosure-based steering

The **Enclosure-based steering** focuses on tracing a circumference of a desired radius *R* enclosing the vehicle, large enough so that it intersects the path at two points. Assuming that the direction of the path to follow is known, the right intersection point is chosen accordingly.

From Figure 3.1, it is possible to determine the desired course angle χ_d, since

$$tan(\chi_d(t)) = \frac{\Delta y(t)}{\Delta x(t)} = \frac{y_{los} - y(t)}{x_{los} - x(t)}\,, \tag{3.1}$$

where $\Delta y(t) = y_{los} - y(t)$ and $\Delta x(t) = x_{los} - x(t)$.

From this point on, let $\boldsymbol{p} = [x(t), y(t)]^T$ be the vehicle's position and $\boldsymbol{p}_{los} = [x_{los}, y_{los}]^T$ be its LOS intersection point. The desired course angle can then be computed as

$$\chi_d(t) = atan2(y_{los} - y(t), x_{los} - x(t))\,. \tag{3.2}$$

By solving the set of Equations in 3.4 and 3.3, the LOS intersection point $\boldsymbol{p}_{los}$ is attainable from

$$[x_{los} - x(t)]^2 + [y_{los} - y(t)]^2 = R^2\,, \tag{3.3}$$

$$tan(\alpha_k) = \frac{y_{k+1} - y_k}{x_{k+1} - x_k} = \frac{y_{los} - y_k}{x_{los} - x_k} = constant\,. \tag{3.4}$$

When it comes to the implementation of this project, the Equation 3.4 can be modified to

$$tan(\alpha_c) = \frac{y_{los} - y_c}{x_{los} - x_c}\,, \tag{3.5}$$

where $\boldsymbol{p}_c = [y_c, x_c]^T$ is the path's closest point to the vehicle and α_c is the path's angle.

After solving the Equations 3.4 and 3.3, we get two set of equations for each case of $\mid \Delta x \mid > 0$ and $\Delta x = 0$, respectively

$$\begin{aligned} x_{los} &= \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}\,, \\ y_{los} &= d(x_{los} - x_k) + y_k\,, \end{aligned} \tag{3.6}$$

and

$$\begin{aligned} x_{los} &= x_k = x_{k+1}\,, \\ y_{los} &= \begin{cases} y + \sqrt{R^2 - (x_{los} - x)^2}, & \text{if } \Delta y > 0\,, \\ y - \sqrt{R^2 - (x_{los} - x)^2}, & \text{if } \Delta y < 0\,, \end{cases} \end{aligned} \tag{3.7}$$

where

$$\begin{gathered} d := (\tfrac{\Delta y}{\Delta x}), \quad e := x_k, \quad f := y_k, \quad g := f - d \cdot e = y_k - (\tfrac{\Delta y}{\Delta x}) \cdot x_k, \\ a := 1 + d^2, \quad b := 2(d \cdot g - d \cdot y - x), \quad c := x^2 + y^2 + g^2 - 2g \cdot y - R^2 \quad , \end{gathered}$$

and the case for $\Delta x = 0$ and $\Delta y = 0$ is not an option, since it would mean that the vehicle was on top of the LOS generated point , which is impossible considering the condition set for this point.

Lookahead-based steering

This method consists on finding the desired course angle χ_d through an auxiliary lookahead distance which defines $\boldsymbol{p}_{los}$, the LOS intersection point on the path. After calculating $\boldsymbol{p}_c$, the point on the path which is closest to the vehicle, the lookahead distance is used to find $\boldsymbol{p}_{los}$ on the path ahead of $\boldsymbol{p}_c$.

For the **Lookahead-based steering**, the desired course angle χ_d is assigned by two components

$$\chi_d(e) = \chi_p + \chi_r(e)\,, \tag{3.8}$$

where $\chi_p = \alpha_k$ is the same angle that was discussed in the **Enclosure-based sterring** method, in other words, the angle of the straight line α_c, while

$$\chi_r(e) = arctan\left(\frac{-e}{\Delta}\right) = arctan(-K_p e)\,, \tag{3.9}$$

where $K_p = 1/\Delta > 0$ and works like a proportional gain. This feature can be quite useful since the angle is always restricted between $\chi_r(e) \in [-\pi/2, \pi/2]$. To ensure good behaviour of the PF, the control law

can be modified to

$$\chi_r(e) = arctan\left(-K_p e - K_i \int_0^t e(\tau)d\tau\right), \tag{3.10}$$

where $K_i > 0$ is an integral gain to regulate the integral part of the control law. The integral action is helpful when there is sideslip in underactuated vehicles, while also avoiding overshoots and windup effects, if necessary. The cross-track error $e(t)$ observed in the previous equations can be extracted from the error vector

$$\boldsymbol{\epsilon}(t) = \begin{bmatrix} s(t) \\ e(t) \end{bmatrix} = \boldsymbol{R}_p(\alpha_k)^T(\boldsymbol{p}(t) - \boldsymbol{p}_k), \tag{3.11}$$

where

$$\boldsymbol{R}_p(\alpha_k) := \begin{bmatrix} cos(\alpha_k) & -sin(\alpha_k) \\ sin(\alpha_k) & cos(\alpha_k) \end{bmatrix}. \tag{3.12}$$

.

Now the desired course angle χ_d is perfectly attainable by substituting the cross-track error $e(t)$ in Equation 3.10. In Figure 3.2, it is shown a clear representation of the two discussed methods, the enclosure circle of radius R and the lookahead distance Δ.

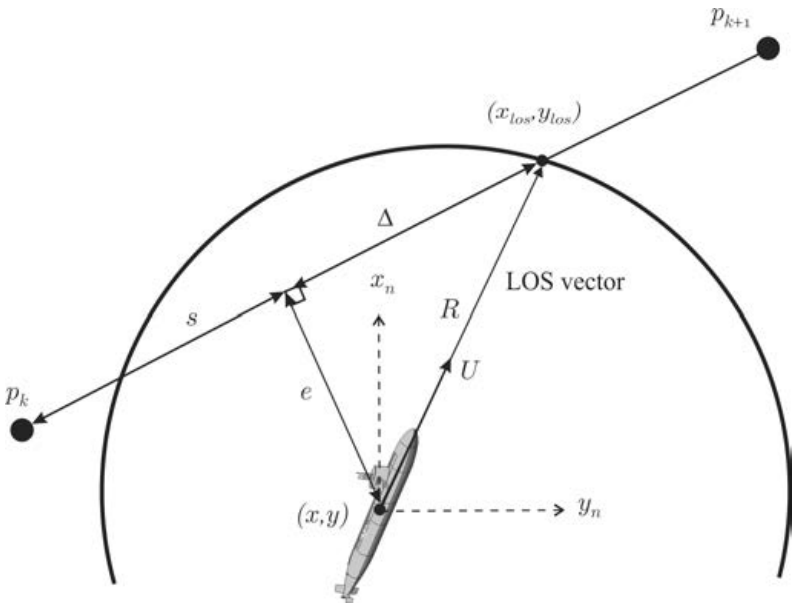


Figure 3.2: Both **Enclosure-based** and **Lookahead-based steering** methods demonstrated. [18].

3.1.2 Line-of-Sight for Circular Paths

In this section, a similar method to **Lookahead-based steering** is applied. However, there are some differences worth mentioning. Firstly, it is assumed that the curvilinear characteristics of each point of the path are known. For simplicity's sake, this section will consider the case of a simple known arc of circumference. Secondly, the vehicle position (the centre of mass of the vehicle $\boldsymbol{p} = [x, y]^T$) and the LOS point $\boldsymbol{p}_{los}$ must be defined so that the LOS vector $\boldsymbol{vec}_{los} = [d(t), e(t)]^T$ is well determined. First, the closest point of the path to the vehicle $\boldsymbol{p}_c$ is computed. Then, using the tangential angle θ, the point

$\boldsymbol{p}_c$ on the path and the chosen lookahead distance d, the LOS point $\boldsymbol{p}_{los}$ is easily determined as

$$\begin{aligned} \boldsymbol{p}_{los} &= \boldsymbol{p}_c + \begin{bmatrix} d \cdot cos\left(\theta + \frac{\pi}{2}\right) \\ d \cdot sin\left(\theta + \frac{\pi}{2}\right) \end{bmatrix}, \\ \boldsymbol{p}_c &= \boldsymbol{C} + \begin{bmatrix} R \cdot sin(\theta) \\ R \cdot cos(\theta) \end{bmatrix}, \end{aligned} \tag{3.13}$$

where the angle θ is calculated as

$$\theta = arctan\left(\frac{y - C_y}{x - C_x}\right), \tag{3.14}$$

with $\boldsymbol{C} = [C_x, C_y]^T$ as the centre of the arc of circumference.

In Figure 3.3 it can be seen that some notions change, for example, one relevant angle previously stated was the path angle α_k, but now we use a different varying angle which depends on the position of the vehicle in relation with to the circumference arc and subsequently the LOS point $\boldsymbol{p}_{los}$ dictated by the lookahead value d, denoted as σ. This is because the lookahead distance is no longer completely set on the path, but built on the point on the path closest to the vehicle.

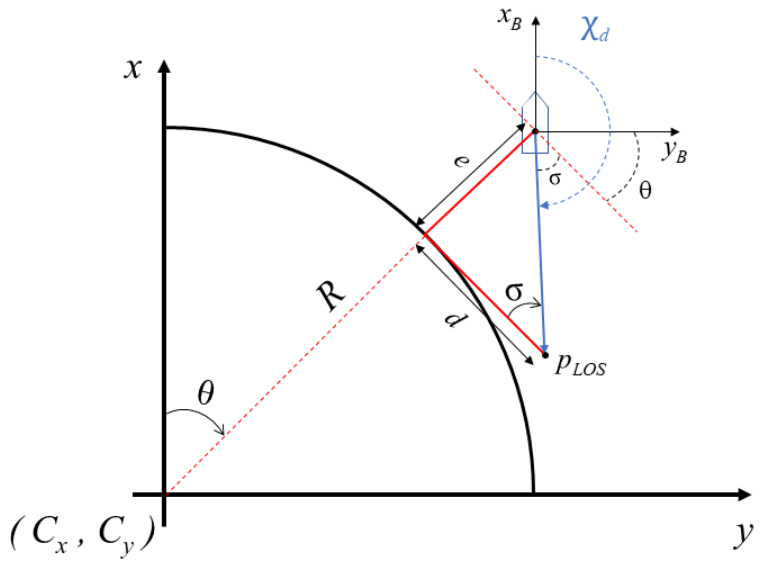


Figure 3.3: Example sketch for circular LOS Path Following.

To design the control law, the desired course angle χ_d is required, which in turn can be formulated as

$$\chi_d = \frac{\pi}{2} + \theta + \sigma, \tag{3.15}$$

as it can be seen from Figure 3.3. The angle σ is calculated as

$$\sigma = arctan\left(\frac{e(t)}{d(t)}\right), \tag{3.16}$$

where $e(t)$ is the cross-track error and d is the lookahead distance. The error vector $\boldsymbol{\epsilon}(t)$ can therefore be expressed as

$$\boldsymbol{\epsilon}(t) = \begin{bmatrix} d(t) \\ e(t) \end{bmatrix} = \mathbf{R}_p(\phi)(\boldsymbol{p}_{los}(t) - \boldsymbol{p}(t)) \,, \tag{3.17}$$

where $\mathbf{R}_p$ is the rotation matrix defined in the previous section and $\phi = -\left(\theta + \frac{\pi}{2}\right)$ is a rotation angle to get the LOS vector and the components $d(t)$ and $e(t)$ of the vector aligned with the xx axis and the yy axis, respectively.

To finally compute the desired course angle χ_d, one must finish the calculation of the cross-track error $e(t)$. After some simple algebra, the following expression for $e(t)$ is attainable

$$e(t) = \left(R \cdot cos\theta + d \cdot cos\left(\theta + \frac{\pi}{2}\right) + C_x - x\right) sin\phi + \left(R \cdot sin\theta + d \cdot sin\left(\theta + \frac{\pi}{2}\right) + C_y - y\right) cos\phi \,. \tag{3.18}$$

Now that the cross-track error $e(t)$ is calculated, the angle σ is possible to compute and subsequently so is the desired course angle χ_d for the final control law

$$\chi_d = \frac{\pi}{2} + arctan\left(\frac{y - C_y}{x - C_x}\right) + arctan\left(\frac{e}{d}\right) \,. \tag{3.19}$$

The previous $arctan$ used for computing θ and σ can be replaced with the function $atan2()$ so that the angles are calculated with respect to the four quadrants.

3.1.3 LOS Controller Definition

As mentioned in [18], the objective of the controller is to minimise the ahead tracking error so that the vehicle has a fitting behaviour, in this case, path convergence. The error is then defined as

$$\tilde{\psi} = \psi - \psi_d \,, \tag{3.20}$$

where ψ is the measured heading of the vehicle and ψ_d is the desired heading dictated by the PF algorithm fed to the vehicle's autopilot or other navigation system. Consequently, a simple Proportional-Integral-Derivative (PID) type control law would be

$$\tau(t) = -K_p\tilde{\psi}(t) - K_d\dot{\tilde{\psi}}(t) - K_i \int_0^t \tilde{\psi}(\lambda) d\lambda \,, \tag{3.21}$$

where τ is the input of the vehicle's system. In case there is the presence of sideslip, in other words, $\beta \neq 0$, the desired heading will be $\psi_d = \chi_d - \beta$ thus shaping the error

$$\tilde{\psi} = \psi - \psi_d = \psi - \chi_d + \beta \,. \tag{3.22}$$

If the velocity for the kinematic model of the vehicle is known, the sideslip can be computed as

$$\beta = arcsin\left(\frac{v}{U}\right) \,. \tag{3.23}$$

3.1.4 Simplified Line of Sight (P. Maurya)

The following method was proposed in Maurya et al. [23] with the end goal of simplifying the implementation of the previously formulated controllers. To achieve this, an **inner-outer loop control** was sought so that the implementation of the guidance system would not directly depend on the characteristics of the vehicle's dynamics and autopilot, this is, the inner loop. This implementation is especially useful since it accounts for ocean currents without being necessary to estimate said forces.

For simplicity's sake, and similar to the previous LOS sections, the demonstration of the method is done on a 2D environment, for a single vehicle or AUV whose objective is to solely follow a vertical line, leading the cross-track error $e(t)$ to zero. In Figure 3.4 it is possible to observe the setting for the demonstration of the presented method.

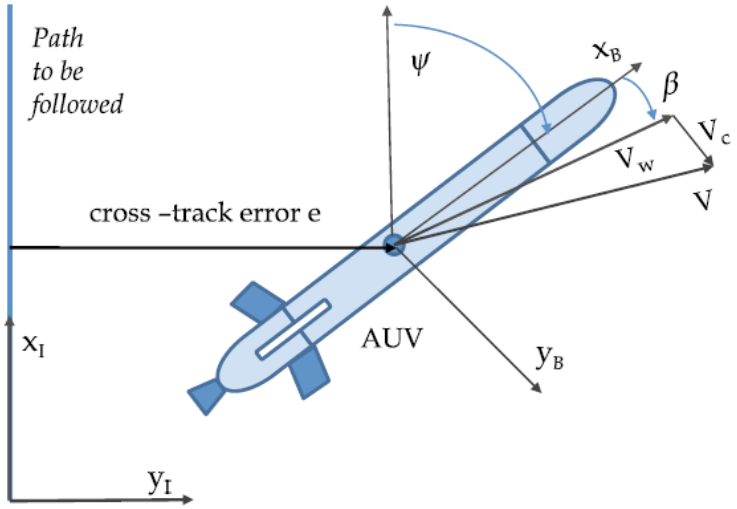


Figure 3.4: Description of the simplified PF problem and variables of interest. [23]

The inertial frame is described in Figure 3.5 as $\{\boldsymbol{I}\} = \{x_I, y_I\}$, as well as the body-fixed frame as $\{\boldsymbol{B}\} = \{x_b, y_b\}$. The variable $\boldsymbol{V}_w$ represents the velocity of the vehicle relative to the water expressed in the body-fixed frame $\{\boldsymbol{B}\}$, $\boldsymbol{V}_c = [v_{cx}, v_{cy}]^T$ is the velocity of the current in the inertial frame $\{\boldsymbol{I}\}$ and $\boldsymbol{V}$ is the total inertial velocity of the vehicle also expressed in the inertial frame $\{\boldsymbol{I}\}$. As usual, ψ and β represent the heading and sideslip of the vehicle, respectively. The total speed of the vehicle is attainable when the vehicle stops rotating and settles on a certain heading. If $\boldsymbol{P}$ denotes the position of the centre of mass of the vehicle in the inertial frame $\{\boldsymbol{I}\}$, the vehicle's velocity $\boldsymbol{V}$, or the derivative of $\boldsymbol{P}$ with respect to time, can be characterised as

$$\dot{\boldsymbol{P}} = \boldsymbol{R}(\psi)\boldsymbol{V}_w + \boldsymbol{V}_c\,, \tag{3.24}$$

where $\boldsymbol{R}(\psi)$ denotes the rotation matrix from $\{\boldsymbol{B}\}$ to $\{\boldsymbol{I}\}$ according to the angle ψ. If the sideslip angle

is included, for instance, due to currents, the derivative of the vehicles position can be expressed as

$$\dot{\boldsymbol{P}} = \boldsymbol{R}(\psi + \beta) \begin{bmatrix} ||\boldsymbol{V}_w|| \\ 0 \end{bmatrix} + \boldsymbol{V}_c \,. \tag{3.25}$$

From this equation it is possible to extract the essential cross-track error $e(t)$ from the y coordinate of the vehicle's position vector $\boldsymbol{P}$, and thus extending it to its derivative with respect to time

$$\dot{e} = sin(\psi + \beta)||\boldsymbol{V}_w|| + v_{cy} \,, \tag{3.26}$$

where v_{cy} is the y component of the current velocity vector $\boldsymbol{V}_c$, along the unit vector y_I. To simplify the problem, it is assumed that the velocity $||\boldsymbol{V}_w|| = U > 0$ is constant and subsequently extended to the case of $||\boldsymbol{V}_w|| > ||\boldsymbol{V}_c||$. To make the error $e(t)$ converge to zero, the derivative of the error $\dot{e}$ and input u are defined as

$$\begin{aligned} \dot{e} = U \cdot sin(\psi) + v_{cy} \quad \wedge \quad u = sin(\psi) \Leftrightarrow \\ \Leftrightarrow \dot{e} = U \cdot u \quad \wedge \quad u = sin(\psi) \,. \end{aligned} \tag{3.27}$$

If it is possible to manipulate the virtual input u, the choice of control law for the virtual input $u = -\left(\frac{K_1}{U}\right) e$ can be done, simplifying one step further the problem at hand, making the error asymptotically exponentially converge to the origin. The introduction of an integral term in the virtual input u comes from considering a possible constant bias like the v_{cy}, to cancel the effects of current, rewritten as

$$u = -\frac{1}{U}\left(K_1 e - K_2 \int_0^t e(\tau) d\tau\right) \,. \tag{3.28}$$

Consequently, the dynamics of e become

$$\dot{e} + K_1 e + K_2 \int_0^t e(\tau) d\tau = 0 \,. \tag{3.29}$$

By making a variable substitution, the following second order system is deduced as

$$\ddot{\varsigma} + K_1 \dot{\varsigma} + K_2 \varsigma = 0 \quad \wedge \quad \varsigma = \int_0^t e(\tau) d\tau \,, \tag{3.30}$$

where the gains K_1 and K_2 can be chosen accordingly so that a desired natural frequency and damping vector are obtained. After these computations, the desired heading has to be written as

$$\psi_d = arcsin(sat(u)) \,, \tag{3.31}$$

where $sat(u)$ is the process of saturating the input so it lies in the interval $[-1, 1]$ for the operation of

$arcsin$. The saturation of the input can be described as

$$sat(u) = \begin{cases} l_s, & \text{if } u \geq l_s\,, \\ u, & \text{if } |u| < l_s\,, \\ -l_s & \text{if } u \leq -l_s\,, \end{cases} \tag{3.32}$$

where $l_s \in [0, 1]$ is a variable to guarantee the saturation. The final control law deduced from this line of thought in Equation 3.31 must account for integral windup, and so the final form of the control law ψ_d will involve a new definition of u. It is formulated in terms of the operator $f : e \rightarrow \psi_d$ as

$$\begin{aligned} \psi_d &= arcsin(sat(u))\,, \\ u &= \left(-\frac{K_1}{U}e - \frac{K_2}{U}\varsigma\right)\,, \end{aligned} \tag{3.33}$$

where ς is the result of the dynamical system designed for the implementation of the integral antiwindup mechanism, defined as the dynamical system $f_{aw}(e) : e \rightarrow \varsigma$, described as

$$\dot{\varsigma} = \begin{cases} e, & \text{if } |u| < l_s\,, \\ 0, & \text{otherwise}\,. \end{cases} \tag{3.34}$$

The convergence of the previous control law can be shown using Lyapunov-based analysis tools, as stated in Maurya et al. [23], if the vehicle heading ψ and the desired heading reference ψ_d are the same. The resulting control scheme is presented in Figure 3.5, where it can be clearly seen that the heading autopilot (inner loop) only depends on the desired heading angle ψ_d.

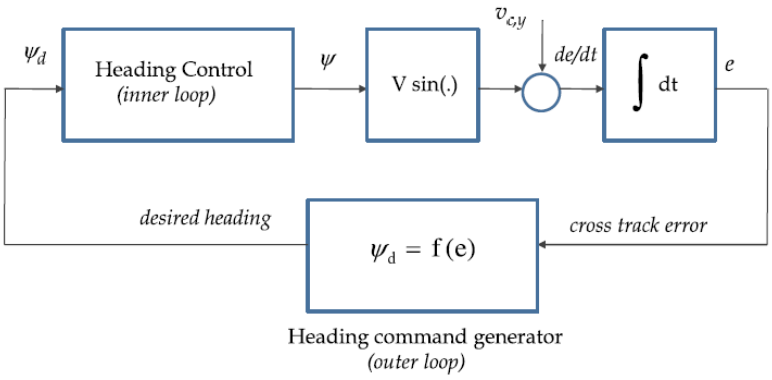


Figure 3.5: Control scheme for the explained simplified guidance method. [23]

3.1.5 Lyapunov-Oriented Path Following Control

The following algorithm was proposed in Micaelli et al. [22] and brought a new approach to the PF problem. The basic idea was to parameterise the path in space and project the vehicle on the path. This projection, according to what was stated in [22], corresponds to the closest point on the path to

the vehicle. The cross-track error is then taken from the distance from the path to the vehicle. After this first analysis, the convergence conditions of the two variables of interest for the PF problem, namely the cross-track error and the desired heading angle, are tested using Lyapunov methods for non-linear stability analysis. The introduction of specially designed functions to achieve the desired control is the great differentiating factor of the technique described in [22], instead of simply using the error variables stated before, the cross-track error and the desired heading angle. Therefore, two expressions are designed to converge to zero, one in function of the cross-track error and another in function of the difference between the measured heading angle and the desired heading angle. The favourable feature of having costumed designed functions is that it is possible to tune the velocity of convergence of the cross-track error and the heading, separately.

Model Description

There are two main frames, the inertial frame $(\boldsymbol{R})$, with the unit vector pair $\{\overrightarrow{i}, \overrightarrow{j}\}$ representing the two axis of the frame, and the so called *Serret-Frenet* frame $(\boldsymbol{T})$ centred on the projection of the vehicle on the path, with the unit vector pair $\{\overrightarrow{m}, \overrightarrow{n}\}$ representing the two axis of the frame. These concepts can be easily observed in Figure 3.6, as well as a curve $(\boldsymbol{C})$, where each point on the curve can be described as $\boldsymbol{P}(s)$ (s being the path parameterisation $f : s \rightarrow \boldsymbol{P}$) with an associated angle of inclination θ_c, relative to the inertial frame $(\boldsymbol{R})$.

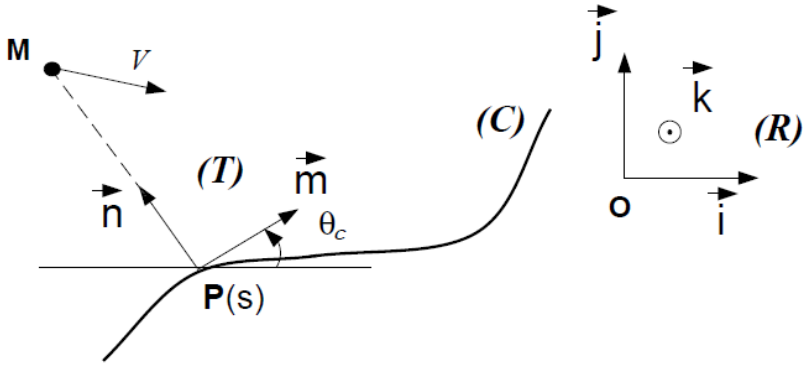


Figure 3.6: Example of graphical representation of vehicle projection and variables of interest. [22]

From classical laws of mechanics, this expression is obtained

$$\left(\frac{d\overrightarrow{OM}}{dt}\right)_R = \left(\frac{d\overrightarrow{OP}}{dt}\right)_R + \left(\frac{d\overrightarrow{PM}}{dt}\right)_T + \overrightarrow{\omega}_c \times \overrightarrow{PM}\,, \tag{3.35}$$

where $\left(\frac{d\overrightarrow{OM}}{dt}\right)_R$ denotes the time derivative with respect to the inertial frame $(\boldsymbol{R})$ of the position of the vehicle, $\left(\frac{d\overrightarrow{OP}}{dt}\right)_R$ represents the time derivative with respect to the inertial frame $(\boldsymbol{R})$ of the projection of the vehicle on the parameterised path, $\left(\frac{d\overrightarrow{PM}}{dt}\right)_T$ denotes the time derivative of the representation of the vehicle with respect to the *Serret-Frenet* $(\boldsymbol{T})$, in other words, the time derivative of the cross-track error.

Finally, the last term $\overrightarrow{\omega}_c \times \overrightarrow{PM}$ represents the cross-product of the vectors

$$\overrightarrow{\omega}_c = \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_c = c_c(s)\dot{s} \end{bmatrix}, \quad \overrightarrow{PM}_T = \begin{bmatrix} 0 \\ y \\ 0 \end{bmatrix}, \tag{3.36}$$

where $\overrightarrow{\omega}_c$ is the vehicle rotation velocity vector of the frame $(\boldsymbol{T})$ with respect to $(\boldsymbol{R})$. After analysing carefully Equation 3.35, a new expression can be deduced as

$$\begin{aligned} \boldsymbol{R}_T^R \begin{bmatrix} \dot{X} \\ \dot{Y} \\ 0 \end{bmatrix} &= \begin{bmatrix} \dot{s} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \dot{y} \\ 0 \end{bmatrix} + \begin{bmatrix} -c_c(s)y \cdot \dot{s} \\ 0 \\ 0 \end{bmatrix} \Leftrightarrow \\ \Leftrightarrow \begin{bmatrix} \dot{s} \\ \dot{y} \\ 0 \end{bmatrix} &= \boldsymbol{R}_T^R \begin{bmatrix} \dot{X} \\ \dot{Y} \\ 0 \end{bmatrix} + \begin{bmatrix} c_c(s)y \cdot \dot{s} \\ 0 \\ 0 \end{bmatrix}, \end{aligned} \tag{3.37}$$

where $\boldsymbol{R}_T^R$ is the transfer rotation matrix from the inertial frame $(\boldsymbol{R})$ to the *Serret-Frenet* frame $(\boldsymbol{T})$, denoted as

$$\boldsymbol{R}_T^R = \begin{bmatrix} cos\theta_c & sin\theta_c & 0 \\ -sin\theta_c & cos\theta_c & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.38}$$

Lastly, the following expressions for the time derivative of the projection of the vehicle s and of the cross-track error y can be formulated as

$$\begin{cases} \dot{s} = \begin{bmatrix} cos\theta_c & sin\theta_c \end{bmatrix} \cdot \begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} /(1 - c_c(s)y), \\ \dot{y} = \begin{bmatrix} -sin\theta_c & cos\theta_c \end{bmatrix} \cdot \begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix}. \end{cases} \tag{3.39}$$

Moving forward, the kinematics of the vehicle have to be defined, and so in a very simple manner, it is assumed that

$$\begin{cases} \begin{bmatrix} \dot{X} \\ \dot{Y} \end{bmatrix} &= v \cdot \begin{bmatrix} cos\theta_m \\ sin\theta_m \end{bmatrix}, \\ \dot{\theta}_m &= \omega, \end{cases} \tag{3.40}$$

where θ_m represents the heading of the vehicle, v the translational velocity of the vehicle and ω is the rotational velocity of said vehicle. For PF purposes, the translational velocity v can be specified beforehand. Using this new information on Equation 3.39, the following expressions with respect to the

parameterisation (s, y), respectively the vehicle projection and the cross-track error, are deduced as

$$\begin{cases} \dot{s} = \frac{v \cdot cos(\theta_m - \theta_c)}{1 - c_c(s)y}\,, \\ \dot{y} = v \cdot sin(\theta_m - \theta_c)\,, \\ \dot{\theta}_m = \omega\,. \end{cases} \tag{3.41}$$

After all these steps, the reasonable choice for a control variable is the angular velocity ω. Furthermore, the derivative of the the curvature with respect to time is formulated as

$$\dot{c}_c(s(t)) = g_c(s(t)) \cdot \dot{s}(t)\,. \tag{3.42}$$

Lyapunov Analysis

Lyapunov based methods are the tool par excellence for the analysis and design of nonlinear control systems (consult [24] for more information on Lyapunov analysis). As seen in Section 4 of [22], the great advantage of using these methods for this problem is the use of adequate Lyapunov functions that not only work for convergence analysis purposes, but also for the control design stage.

The initial Lyapunov function can be expressed as

$$V = \frac{1}{2}\left[y^2 + \frac{1}{\lambda_\theta}\theta^2\right]\,, \tag{3.43}$$

where y is the *cross-track error*, θ is the difference between the measured heading θ_m and the desired heading, in other words, the tangential angle on the closest point to the vehicle on the path θ_c, and λ_θ is a scaling factor. Another step taken in [22] was an improvement to the Lyapunov method in Equation 3.43, given as

$$V = \frac{1}{2}\left[f^2(y) + \frac{1}{\lambda_\theta}(\theta - \delta(y, v))^2\right]\,, \tag{3.44}$$

where the functions $f(y) :]-r, r[\to \Re$ and $\delta(y, v) : \Re \times \Re \to \Re$ are both C^1 and C^2, and such that

$$\begin{aligned} &H_1 : f(\pm r) = \pm\infty\,, \\ &H_2 : f(0) = \delta(0, v) = 0, \forall v\,, \\ &H_3 : f'_y(y) > 0, \forall y\,, \\ &H_4 : v f(y) sin(\delta(y, v)) \leq 0, \forall y, \forall v\,. \end{aligned} \tag{3.45}$$

These hypotheses have to be respected in order to have the desired control. The functions are introduced to attain a broader domain of the PF problem, not only to achieve stability but also to reach the control objectives. By computing the time derivative of the Lypunov function, an expression arises as

$$\begin{aligned} \dot{V} &= f f'_y v sin\theta + \frac{1}{\lambda_\theta}(\theta - \delta) \cdot (\dot{\theta} - \delta'_y v sin\theta - \delta'_v \dot{v}) \\ &= f f'_y v sin\delta + (\theta - \delta) \cdot \left[f f'_y v \frac{sin\theta - sin\theta\delta}{\theta - \delta} + \frac{1}{\lambda_\theta}\left(\dot{\theta} - \delta'_y v sin\theta - \delta'_v \dot{v}\right)\right]\,, \end{aligned} \tag{3.46}$$

where the dependencies have been omitted for the sake of simplicity. The term f'_y represents the derivative of $f(y(t))$ with respect to $y(t)$, and δ'_y and δ'_v represents the derivative of $\delta(y(t), v(t))$ with respect to $y(t)$ and $v(t)$, respectively. To cancel the undesirable terms, a control law is formulated as

$$\omega = c_c \frac{vcos\theta}{1 - c_c y} + \delta'_y v sin\theta + \delta'_v \dot{v} - \lambda_\theta f f'_y v \frac{sin\theta - sin\delta}{\theta - \delta} - k\lambda_\theta |v|(\theta - \delta) \quad , \quad k > 0,\ \lambda_\theta > 0\,. \tag{3.47}$$

Some details need attention, for instance, if $|y(0)| < \frac{1}{c_{cmax}}$ and if $v(t)$ and $\dot{v}(t)$ are bounded, and if $v(t)$ does not tend to zero when t tends to infinity, then the solutions $y(t)$ and $\theta(t)$ asymptotically converge to zero. Joining the two Equations, 3.46 and 3.47, if $\dot{\theta}$ is replaced by

$$\omega - c_c \frac{vcos\theta}{1 - c_c y}\,, \tag{3.48}$$

where ω is given by 3.47, the resulting expression becomes

$$\dot{V} = f f'_y v sin\delta - k|v|(\theta - \delta)^2 \leq 0\,. \tag{3.49}$$

If this condition is respected, it is possible to conclude that $V(t)$ is non-increasing, assuming that $|f(y(t))|$ and $|\theta(t) - \delta(y(t), v(t))|$ are bounded. Therefore, from the properties of f and δ previously stated, it is expected that $|y(t)| < \frac{1}{c_{cmax}} - \epsilon$ and $|\theta(t)| < \zeta$, knowing that $\exists\, \epsilon, \zeta > 0, \forall t$, thus being bounded.

It is now known that $V(t)$ converges to some limit value and, since LaSalle's principle can not be utilized because the system is autonomous, or time-variant, one must resort to Barbalat's Lemma, as explained in Liu [25], which states that given a function $f(t) \in C^1(a, \infty)$ and $\lim_{t\to\infty} f(t) = \alpha$, where $\alpha < \infty$, if its derivative f' is uniformly continuous (or its double derivative f'' is bounded), then $\lim_{t\to\infty} f' = 0$.

Using this powerful tool, the proof that $V(t) \to 0$ as $t \to \infty$ is intuitively understandable, even more if the different terms of the equation are carefully analysed.

Now that the control law is formulated, it is necessary to propose satisfactory functions for $f(y(t))$ and $\delta(y(t), v(t))$ to achieve the desired control objectives. In [22] two simple enough expressions are proposed, which respect to the conditions denoted in the paper, given as

$$\delta(y, v) = -sign(v) g_\delta(y)\,, \tag{3.50}$$

where $sign(\cdot)$ is the signum function and $g_\delta(y)$ can be represented as

$$g_\delta(y) = \theta_a \frac{e^{2k_{\delta y}} - 1}{e^{2k_{\delta y}} + 1} \quad , \quad k_\delta > 0 \quad , \quad 0 \leq \theta_a < \pi\,. \tag{3.51}$$

For the function $f(y)$, in order to keep $\theta - \delta$ small, one should maintain $f(y) f'_y(y)$ as small as possible,

in the largest possible domain. Thus, another expression is suggested as

$$f(y) = \begin{cases} \frac{y/k_1}{\left(1+(y/k_2)^2\right)^{1/3}}\,, & \text{if } c_{cmax} = 0 \\ \frac{g_y(y)/k_1}{\left(1+(g_y(y)/k_2)^2\right)^{1/3}}\,, & \text{if } c_{cmax} \neq 0 \end{cases} \text{, with } k_1 > 0,\ k_2 > 0\,, \tag{3.52}$$

where

$$g_y(y) = \frac{r}{2} Log \frac{r+y}{r-y} \quad , \quad r > 0\,. \tag{3.53}$$

A final remark for the choice of these functions is that the linearisation of the system's closed loop equations about $y = 0$ and $\theta = 0$, when v keeps the same sign, is given as

$$y'' + k_{v_y} y' + k_{p_y} y = 0\,, \tag{3.54}$$

where

$$\begin{aligned} k_{v_y} &= k_\delta \theta_a + k\lambda_\theta\,, \\ k_{p_y} &= \lambda_\theta \left(\frac{1}{k_1^2} + k k_\delta \theta_a \right)\,. \end{aligned} \tag{3.55}$$

These equations can be used to determine the constants referred before as λ_θ, k, k_δ and k_1 from values specified beforehand of k_{p_y} and k_{v_y}.

3.2 Nonsingular Path Following Control

In this section, a new approach was taken to the PF problem, as seen in Lapierre et al. [7]. Instead of having a simple projection of the vehicle on the path, representing the closest point on the path to the vehicle, it was thought of another more versatile way to represent this target. By having a movable virtual target on the path, and by controlling its rate of progression, the constraints for the problem become more flexible, specially in terms of initial conditions. For instance, the initial values for the vehicle variables are no longer a constraint, as the PF is now globally asymptotic. Another great advantage is the possibility of a design for the desired velocity along time and the respective convergence of the the real velocity of the vehicle to this value.

3.2.1 Kinematics

The technique presented in this work is for a mobile robot vehicle model, similar to what was done in [7], although it is perfectly adaptable for the AMV model. In order to compute the equations of motion and control laws, the *Serret-Frenet* frame $\{\boldsymbol{F}\}$ is utilised. However, as mentioned before, the frame will be centred on the virtual target of sorts, which in turn will be itself controlled by designed laws. Let $\boldsymbol{P}$ be the point on the path representing the virtual target and $\boldsymbol{Q}$ be the centre of mass of the vehicle. The abscissa of $\boldsymbol{P}$ along the path will be denoted as the curvilinear coordinate s, similar to what was proposed in the

previous section, the path is parameterised so that s is comprised in the interval $s \in [0,\ Len]$, where *Len* represents the total length of the path.

As observed in Figure 3.7, the vehicle's position $\boldsymbol{Q}$ can be expressed as $\mathbf{q} = (X, Y, 0)$ in the inertial frame $\{\boldsymbol{I}\}$ or as $(s_1, y_1, 0)$ in the *Serret-Frenet* frame $\{\boldsymbol{F}\}$. Returning to the problem at hand, and for the sake of simplicity, this PF strategy was designed in 2D, and so $\boldsymbol{Q}$ can be expressed as (X, Y) in $\{\boldsymbol{I}\}$ or (s_1, y_1) in $\{\boldsymbol{F}\}$. Moreover, the point $\boldsymbol{P}$ in $\{\boldsymbol{I}\}$ is thence represented by the vector $\mathbf{p}$.

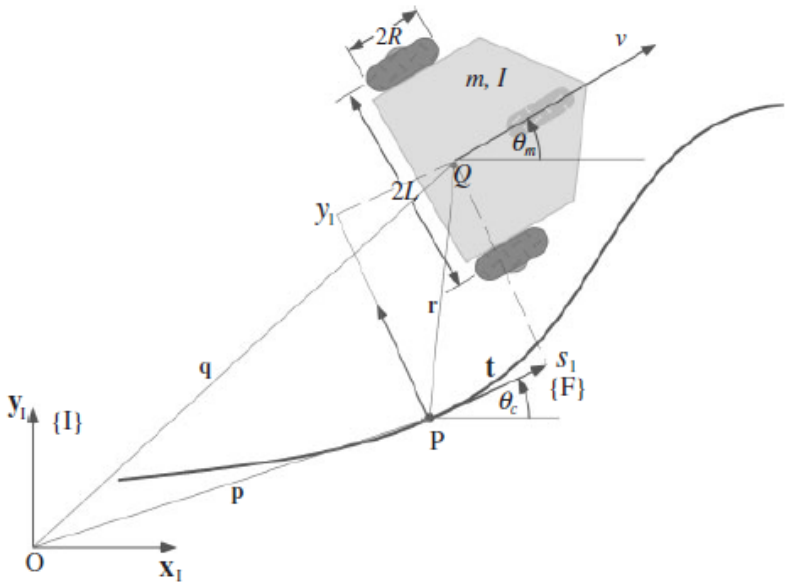


Figure 3.7: Vehicle parameters and frame definitions. [7]

In order to compute all the needed equations and to uphold the transformations from frame $\{\boldsymbol{I}\}$ to $\{\boldsymbol{F}\}$, a rotation matrix $\boldsymbol{R}$ is required, expressed as

$$\mathbf{R}(\theta_c) = \begin{bmatrix} cos\theta_c & sin\theta_c & 0 \\ -sin\theta_c & cos\theta_c & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{3.56}$$

where θ_c denotes the angle formed between $\{\boldsymbol{I}\}$ and $\{\boldsymbol{F}\}$. The derivative of θ_c with respect to time is given by

$$\dot{\theta}_c = \omega_c = c_c(s)\dot{s}\,, \tag{3.57}$$

where $c_c(s)$ is the path curvature. The velocity of the virtual target $\boldsymbol{P}$ in $\{\boldsymbol{I}\}$ can defined in $\{\boldsymbol{F}\}$ as

$$\left(\frac{\partial \mathbf{p}}{\partial t}\right)_F = \begin{bmatrix} \dot{s} \\ 0 \\ 0 \end{bmatrix}. \tag{3.58}$$

Using the equations deduced from classical laws of mechanics, it is possible to deduce the velocity

of $\boldsymbol{Q}$ in $\{\boldsymbol{I}\}$ as

$$\left(\frac{\partial \mathbf{q}}{\partial t}\right)_I = \left(\frac{\partial \mathbf{q}}{\partial t}\right)_I + \mathbf{R}^{-1}(\theta_c)\left(\frac{\partial \mathbf{r}}{\partial t}\right)_F + \mathbf{R}^{-1}(\theta_c)(\omega_c \times \mathbf{r})\,, \tag{3.59}$$

where $\mathbf{r}$ is the vector from $\boldsymbol{P}$ to $\boldsymbol{Q}$ represented in the frame $\{\boldsymbol{F}\}$. Multiplying $\mathbf{R}(\theta_c)$ on both sides, the previous equation becomes

$$\mathbf{R}(\theta_c)\left(\frac{\partial \mathbf{q}}{\partial t}\right)_I = \left(\frac{\partial \mathbf{p}}{\partial t}\right)_F + \left(\frac{\partial \mathbf{r}}{\partial t}\right)_F + \omega_c \times \mathbf{r}\,. \tag{3.60}$$

Computing each parcel of the previous equation, these equations are obtained and conveyed as

$$\left(\frac{\partial \mathbf{q}}{\partial t}\right)_I = \begin{bmatrix}\dot{X}\\ \dot{Y}\\ 0\end{bmatrix} \quad , \quad \left(\frac{\partial \mathbf{r}}{\partial t}\right)_F = \begin{bmatrix}\dot{s}_1\\ \dot{y}_1\\ 0\end{bmatrix} \quad , \quad \omega_c \times \mathbf{r} = \begin{bmatrix}0\\ 0\\ \dot{\theta_c} = c_c(s)\dot{s}\end{bmatrix} \times \begin{bmatrix}s_1\\ y_1\\ 0\end{bmatrix} = \begin{bmatrix}-c_c(s)\dot{s}y_1\\ c_c(s)\dot{s}s_1\\ 0\end{bmatrix}\,. \tag{3.61}$$

With these statements, Equation 3.60 can be rewritten as

$$\mathbf{R}(\theta_c)\begin{bmatrix}\dot{X}\\ \dot{Y}\\ 0\end{bmatrix} = \begin{bmatrix}\dot{s}(1 - c_c(s)y_1) + \dot{s}_1\\ \dot{y}_1 + c_c(s)\dot{s}s_1\\ 0\end{bmatrix}\,. \tag{3.62}$$

After solving for $\dot{s}_1$ and $\dot{y}_1$, these equations are observed

$$\begin{aligned}\dot{s}_1 &= \begin{bmatrix}cos\theta_c & sin\theta_c\end{bmatrix}\begin{bmatrix}\dot{X}\\ \dot{Y}\end{bmatrix} - \dot{s}(1 - c_c y_1)\,,\\ \dot{y}_1 &= \begin{bmatrix}-sin\theta_c & cos\theta_c\end{bmatrix}\begin{bmatrix}\dot{X}\\ \dot{Y}\end{bmatrix} - c_c\dot{s}s_1\,.\end{aligned} \tag{3.63}$$

It is important to denote that in the previous section, directly based on Micaelli et al. [22], $s_1 = 0, \forall t$, since $\boldsymbol{P}$ is the projection of $\boldsymbol{Q}$ on the path, representing the closest point on the path to the vehicle. However, it is also convenient to solve for $\dot{s}$. Though this creates a singularity at $y_1 = \frac{1}{c_c}$, by having the term $(1 - c_c y_1)$ on the denominator, thus making the initial condition problem obvious. The initial condition of $\boldsymbol{Q}$ is restricted to a tube around the path with radius less than $\frac{1}{c_{cmax}}$, where c_{cmax} denotes the maximum curvature of the path. This constraint is very conservative because it could translate in a divergence from the path when there is a large c_{cmax} in even a very small section of the path (strict constraint to the initial vehicle's position), regardless of any initial conditions.

By having $s_1 \neq 0$, the virtual target $\boldsymbol{P}$ will not coincide with the projection thought in the previous section, thus introducing an extra degree of freedom for the controller design. Consequently, the singularity at $y_1 = \frac{1}{c_c}$ is removed, since it is specified how fast the virtual target $\boldsymbol{P}$ moves. The velocity model for the vehicle is then deduced with respect to the frame $\{\boldsymbol{I}\}$ as

$$\begin{bmatrix}\dot{X}\\ \dot{Y}\end{bmatrix} = v\begin{bmatrix}cos\theta_m\\ sin\theta_m\end{bmatrix}\,, \tag{3.64}$$

where θ_m and v denote the true yaw angle and the speed in relation to the body-axis frame of the vehicle, respectively. Introducing the variable $\theta = \theta_m - \theta_c$ and joining Equation 3.63 and 3.64, a new set of equations arises and is given by

$$\begin{cases} \dot{s}_1 &= -\dot{s}(1 - c_c \cdot y_1) + v \cdot cos\theta \,, \\ \dot{y}_1 &= -c_c \cdot \dot{s} \cdot s_1 + v \cdot sin\theta \,, \\ \dot{\theta} &= \omega_m - c_c \cdot \dot{s} \,, \end{cases} \tag{3.65}$$

where $\omega_m = \dot{\theta}_m$.

3.2.2 Nonlinear Controller Design

Now it is possible to derive the control laws for the above system. The objective is similar to the one presented in Section 3.1.5 when it comes to the Lyapunov approach to nonlinear control. Therefore, the following Lyapunov function is obtained

$$V_1 = \frac{1}{2}(s_1^2 + y_1^2) + \frac{1}{2\gamma}(\theta - \delta(y_1, v))^2 \,, \tag{3.66}$$

where it is assumed that

- $\delta(0, v) = 0$,
- $y_1 \cdot v \cdot sin\delta(y_1, v) \leq 0, \ \forall y \ \forall v$,
- $\lim_{t\to\infty} v(t) \neq 0$.

The first term in Equation 3.66 $\frac{1}{2}(s_1^2 + y_1^2)$, depicts the distance between the vehicle and the path, which must converge to zero. The second term of $\frac{1}{2\gamma}(\theta - \delta(y_1, v))^2$ aims to shape the approach angle $\theta = \theta_m - \theta_c$ of the vehicle to the path as a function of the new definition for cross-track error y_1 and speed v, by forcing it to follow a desired orientation profile embodied in the function δ, thus approaching the vehicle to the path before converging the vehicle's heading to the tangential angle θ_c. See Lapierre et al. [7] for a better understanding of these words, and also Micaelli et al. [22] for an in-depth description of what the δ function should look like, just as it was addressed in Section 3.1.5.

The prior assumptions are made to guarantee a series of conditions: firstly, $\delta(0, v) = 0$ serves to assure that when the cross-track error y_1 converges to zero, the heading θ_m converges to the tangent angle θ_c; secondly, $y_1 \cdot v \cdot sin\delta(y_1, v) \neq 0, \ \forall y \ \forall v$ provides an adequate reference sign definition in order to drive the vehicle to the path (turn left when the vehicle is on the right side of the path, and turn right if vice-versa); finally, $\lim_{t\to\infty} v(t) \neq 0$ represents the fact that the vehicle's velocity must be different than zero as time goes to infinity, avoiding a state of rest.

Taking this into consideration and computing the derivative of the Lyapunov function $V_1(t)$ with re-

spect to time, the following equation is attained

$$\begin{aligned}\dot{V}_1 &= s_1 \cdot \dot{s}_1 + y_1 \cdot \dot{y}_1 + \frac{1}{\gamma}(\theta - \delta)(\dot{\theta} - \dot{\delta}) \\ &= s_1(v \cdot cos\theta - \dot{s}(1 - c_c \cdot y_1) - \dot{s}\, c_c \cdot y_1) + y_1 \cdot v \cdot sin\theta + \frac{1}{\gamma}(\theta - \delta)(\dot{\theta} - \dot{\delta}) \\ &= s_1(v \cdot cos\theta - \dot{s}) + y_1 \cdot v \cdot sin\delta + \frac{1}{\gamma}(\theta - \delta)\left(\dot{\theta} - \dot{\delta} + \gamma \cdot y_1 \cdot v \cdot \frac{sin\theta - sin\delta}{\theta - \delta}\right).\end{aligned} \tag{3.67}$$

Now it is possible to define the control laws

$$\begin{cases} \dot{s} &= v \cdot cos\theta + k_1 \cdot s_1\,, \\ \dot{\theta} &= \dot{\delta} - \gamma \cdot y_1 \cdot v \cdot \frac{sin\theta - sin\delta}{\theta - \delta} - k_2 \cdot (\theta - \delta)\,, \end{cases} \tag{3.68}$$

where k_1 and k_2 are positive gains. Substituting Equation 3.68 in Equation 3.67, nullifying undesirable terms, the derivative of the Lyapunov function becomes

$$\dot{V}_1 = -k_1 \cdot s_1^2 + y_1 \cdot v \cdot sin\delta - \frac{k_2}{\gamma}(\theta - \delta)^2\,, \tag{3.69}$$

thus respecting the Lyapunov function condition of $\dot{V}_1 \leq 0$ for the convergence of the variables of interest s_1, y_1 and $\theta - \delta$. Finally, using Barbalat's Theorem, similar to what was done in the previous Section 3.1.5, and from Micaelli et al. [22], it is possible to prove that the variables of interest really do converge to zero as time goes by.

The control output for the desired angle rate ω_m then becomes

$$\omega_m = \dot{\theta} + c_c \cdot \dot{s} \tag{3.70}$$

where $\dot{\theta}$ and $\dot{s}$ are the control laws defined in Equation 3.68.

3.2.3 Simulation Results

The projected PF for a single vehicle can be described by the block diagram in Figure 3.8.

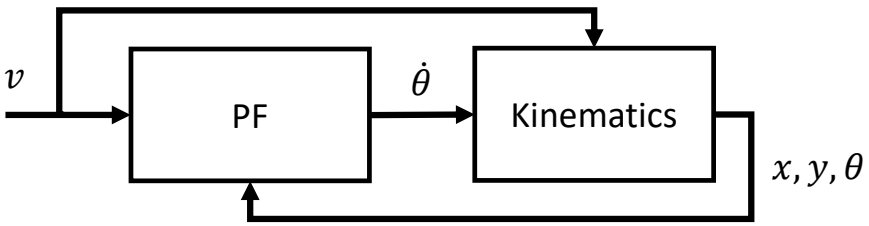


Figure 3.8: PF block diagram for a single vehicle.

The simulation in Figure 3.9 presents a single vehicle governed by the model in Equation 2.24 per-

forming the PF technique from Lapierre et al. [7] explained in the previous section. The simulation runs for two hundred seconds and it is visible how the vehicle converges to the path, effectively reducing the distance between itself and the virtual target, as it can be seen in the PF Error plot in Figure 3.9.

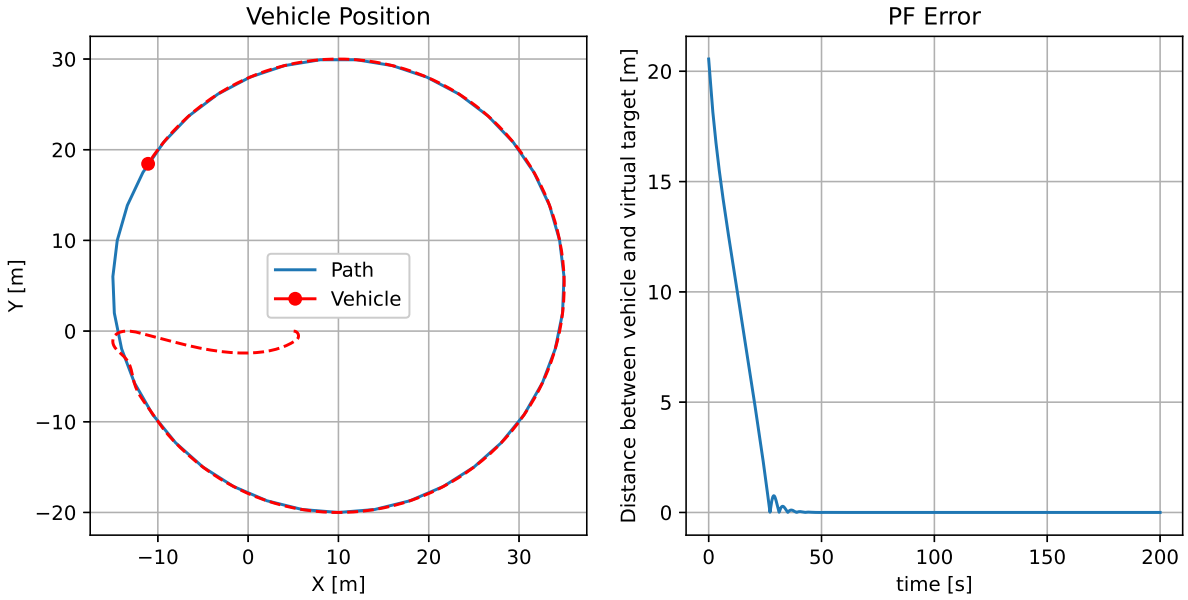


Figure 3.9: Plotting vehicle position and PF Error of the PF simulation.

The parameters used for this simulation are stated in Table 3.1. As long as the speed profile v_p assigned to the vehicle's velocity is not zero, the algorithm converges the vehicle to the path. Since the speed profile within the context of this work is constant, the derivative with respect to time of the vehicle's velocity $\dot{v} = 0$.

Table 3.1: Values for the parameters used in the PF manoeuvre.

γ	k_1	k_2	k_{delta}	θ_a	v_p
1	1	0.3	0.5	0.8	$0.8\,m/s$

Chapter 4

Cooperative Path Following

This section is dedicated to the problem of CPF within the global aim of this thesis. The CPF is an extension of the PF problem described in Chapter 3, where each vehicle must follow an assigned path while simultaneously maintaining a desired geometric configuration compatible with the assigned paths. For this purpose, communication links between vehicles must be established, so that it becomes possible to broadcast motion-related information among them, and design a control law for the velocity of each of them. The topic at hand can be divided in two sequential approaches, a first regarding time-continuous communications, and a second regarding discrete-time communications. After the PF, this is the next key step to design the system previously described in Section 1.2. Figure 4.1 roughly illustrates the idea behind the CPF with two vehicles, where they remain uncoordinated until the two coordination states γ are equal.

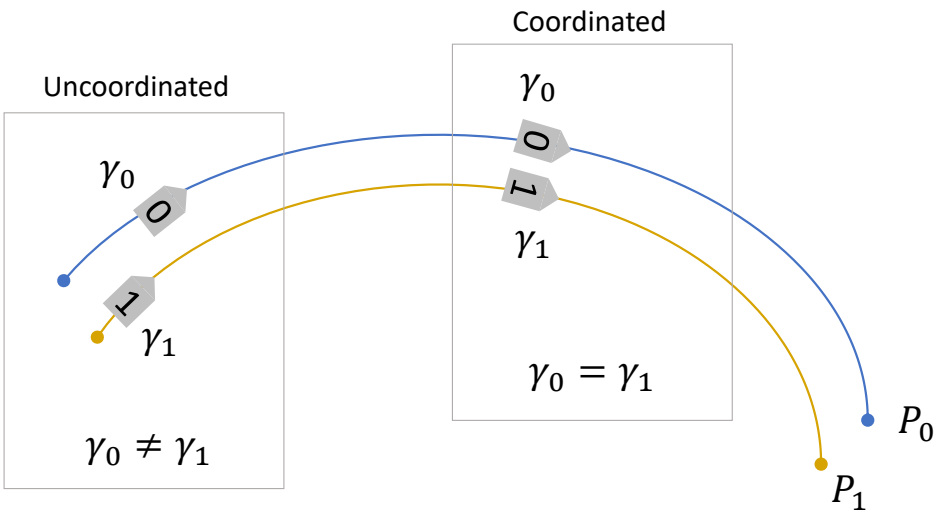


Figure 4.1: Example of the type of coordination between vehicles required for the problem in question.

4.1 Cooperation with Continuous Communications

In a nutshell, the time-continuous CPF technique consists of steering a group of vehicles along a set of spatial paths, with profile or nominal speeds that may be path or time dependent, while holding a feasible geometric pattern, assuming there is constant vehicle inter-communication. Accordingly, the speed of progression along the paths is negotiated between vehicles to compensate for rough initial conditions, unexpected environmental or external disturbances, and temporary reduced performance. The work in Ghabcheloo et al. [8] depicts perfectly these issues, proposing an adequate solution for the underlying time-continuous case of CPF, and in which the theoretical framework borrows from nonlinear control and graph theory, capturing the topology of the underlying inter-vehicle communication network.

4.1.1 Graph Theory

In order to represent the connections between vehicles, an adjacency matrix $\boldsymbol{A}$ is defined, where a link between vehicles is represented in the matrix as $a_{ij} = 1$ and the lack of a communication channel $a_{ij} = 0$, with i and j being the line and columns, respectively. For the purpose of the work in question, it is assumed that all vehicles communicate both ways, so the adjacency matrix $\boldsymbol{A}$ is always symmetric. It is now possible to define the degree matrix $\boldsymbol{D}$ that represents the out-degrees of each node in the graph that is the network of vehicles, in other words, the connections which hail from each vehicle. This is very clearly explained in Vanni [9, section 2.3]. The matrices $\boldsymbol{A}$ and $\boldsymbol{D}$ are then used to calculate the Laplacian of the graph $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}$, which in turn can be converted into the normalised Laplacian $\boldsymbol{L_D} = \boldsymbol{D^{-1}}(\boldsymbol{D} - \boldsymbol{A})$. This last element is especially useful when computing the error between path progression parameters for each vehicle, as it is seen in the next section. Figure 4.2 lays out an example of vehicle network topology, as well as the respective matrices that characterise it, like so

$$\begin{gathered} \boldsymbol{A} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \quad \boldsymbol{D} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}, \\ \boldsymbol{L} = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}, \quad \boldsymbol{L_D} = \begin{bmatrix} 1 & -1/2 & -1/2 \\ -1/2 & 1 & -1/2 \\ -1/2 & -1/2 & 1 \end{bmatrix}. \end{gathered} \tag{4.1}$$

4.1.2 Control Law Formulation

Contemplating the path parameter s defined for the virtual target in Section 3.2, it clearly depicts the progression of the vehicle relative to the path, even more once the vehicle converges to the virtual target. A normalisation of this path parameter s can be used to create the coordination state $\gamma \in [0, 1]$. Using this procedure, it is possible to compare the coordination states between vehicles assigned with path's with different lengths. The work in Rego et al. [10] depicts very clearly what is aimed to accomplish here. For this purpose, let s be the path parameter and vehicle projection from Section 3.2, and L_p the

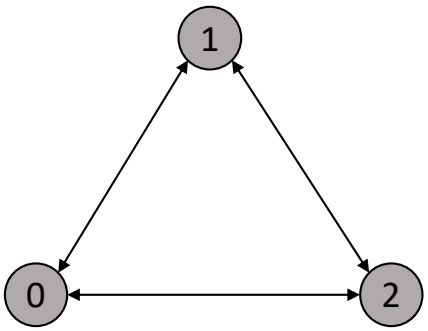


Figure 4.2: Example of three fully connected vehicle network topology represented in graph form.

path's length, the relation between s and γ for a certain vehicle i is therefore

$$\gamma^{[i]}(t) = \frac{s^{[i]}(t)}{L_p}\,. \tag{4.2}$$

To perform an evaluation on the actual state of coordination between vehicles, an error function is thought for each vehicle i and its neighbouring vehicles j, given by

$$\epsilon_i = \gamma_i - \frac{1}{|\mathcal{N}_i|}\sum_{j\in\mathcal{N}_i}\gamma_j\,, \tag{4.3}$$

where $\mathcal{N}_i$ is the group of neighbours of vehicle i. In this stage, the normalised Laplacian $\boldsymbol{L}_D$ becomes quite convenient, letting the error between coordination states be represented in matrix form. The following equation is equivalent to Equation 4.3 in regard to every vehicle in the network, in other words, the coordination error in vector form can be expressed as

$$\boldsymbol{\epsilon} = \boldsymbol{L}_D \cdot \boldsymbol{\gamma}\,, \tag{4.4}$$

where γ is the vector of coordination states for every vehicle in the network, and $\boldsymbol{L_D}$ the normalised Laplacian matrix defined earlier. By computing the coordination error vector $\boldsymbol{\epsilon}$ as observed in [9, section 5.3], the velocity control law is achievable. If each vehicle has an appointed speed profile v_p, then the desired velocity for all vehicles at every time instance is denoted as

$$\boldsymbol{v}_d = \boldsymbol{v}_p + \boldsymbol{v}_c\,, \tag{4.5}$$

where $\boldsymbol{v}_c$ is the corresponding velocity correction, which in turn is designed in accordance with the coordination error vector $\boldsymbol{\epsilon}$, and $\boldsymbol{v}_p$ the vector of speed profiles assigned to each vehicle in the network. Taking this into account, a control law can then be deduced as

$$\begin{aligned}\boldsymbol{v}_c &= -\boldsymbol{k}_\xi \cdot \tanh(\boldsymbol{\epsilon}) \\ &= -\boldsymbol{k}_\xi \cdot \tanh(\boldsymbol{L}_D \cdot \boldsymbol{\gamma})\,,\end{aligned} \tag{4.6}$$

where $\boldsymbol{k}_\xi > \boldsymbol{0}$ is the vector of tuning gains, dictating the maximum absolute value for the velocity correc-

tion $\boldsymbol{v}_c$ of each vehicle, and $\boldsymbol{\epsilon}$ the coordination error vector calculated in Equation 4.4. It is also worth mentioning that with this control law, the desired velocity v_d is bounded. If the analysis is made for a single vehicle i present in the network, joining Equations 4.3 and 4.6, a new equation is obtained and defined as

$$v_c^{[i]} = -k_\xi^{[i]} \cdot \tanh\left(\gamma_i - \frac{1}{|\mathcal{N}_i|}\sum_{j\in\mathcal{N}_i}\gamma_j\right), \tag{4.7}$$

where it is possible to verify the usefulness of the normalised Laplacian matrix $\boldsymbol{L}_D$, as it replaces the computation presented in Equation 4.7, each line of the matrix $\boldsymbol{L}_D$ corresponding to each coordination error operation computed in each vehicle i. The coordination error vector $\boldsymbol{\epsilon}$ is then replaced with the difference between the γ of vehicle i and the average of the γ of its neighbouring vehicles, with j representing each neighbour and $|\mathcal{N}_i|$ representing the total number of neighbours of vehicle i.

4.1.3 Simulation Results

In this section, the previously projected control laws are tested. Further proof of convergence for this algorithm is explicitly presented in Ghabcheloo et al. [8] and Vanni [9, section 5.3]. The resulting system for a single vehicle is represented by the block diagram in Figure 4.3. The vehicle continuously communicates with its neighbours over the Vehicle Network and performs the CPF and PF techniques, controlling the desired velocity v_d and angle rate $\dot{\theta}$ of the vehicle, respectively.

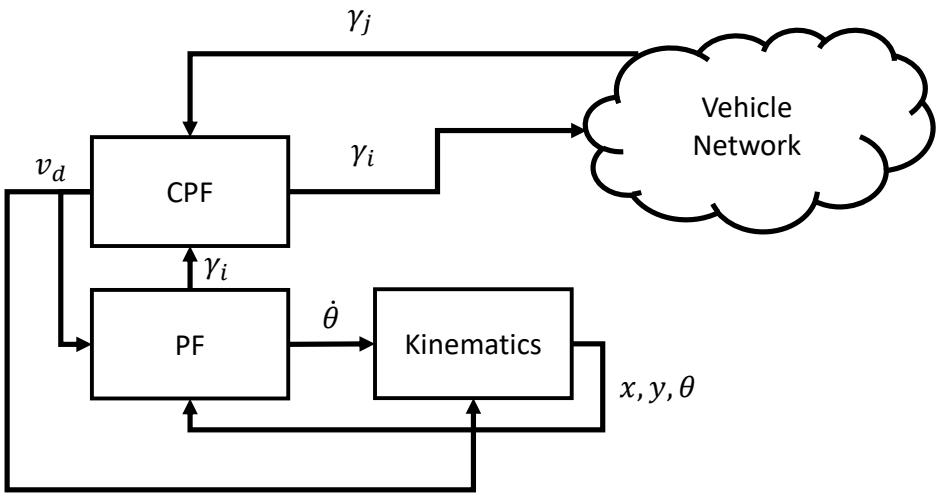


Figure 4.3: CPF system block diagram.

By joining the PF technique shown in Section 3.2 and the CPF technique, and choosing the right value for the tuning parameter k_ξ in the context of the problem at hand, the resulting plots were generated and laid-out in Figure 4.4. The velocity profile v_p assigned to each vehicle, in this specific case, should also consider that paths with different lengths are traversed at the same rate with different speeds. In this sense, in the simulation observed in Figure 4.4, the velocity profile v_p assigned to **Vehicle 0** should be higher than the one assigned to **Vehicle 1**.

Analysing the Coordination Error plot in Figure 4.4, one can observe that the vehicle effectively achieve coordination, as the error between coordination state γ converges to zero. It also important to

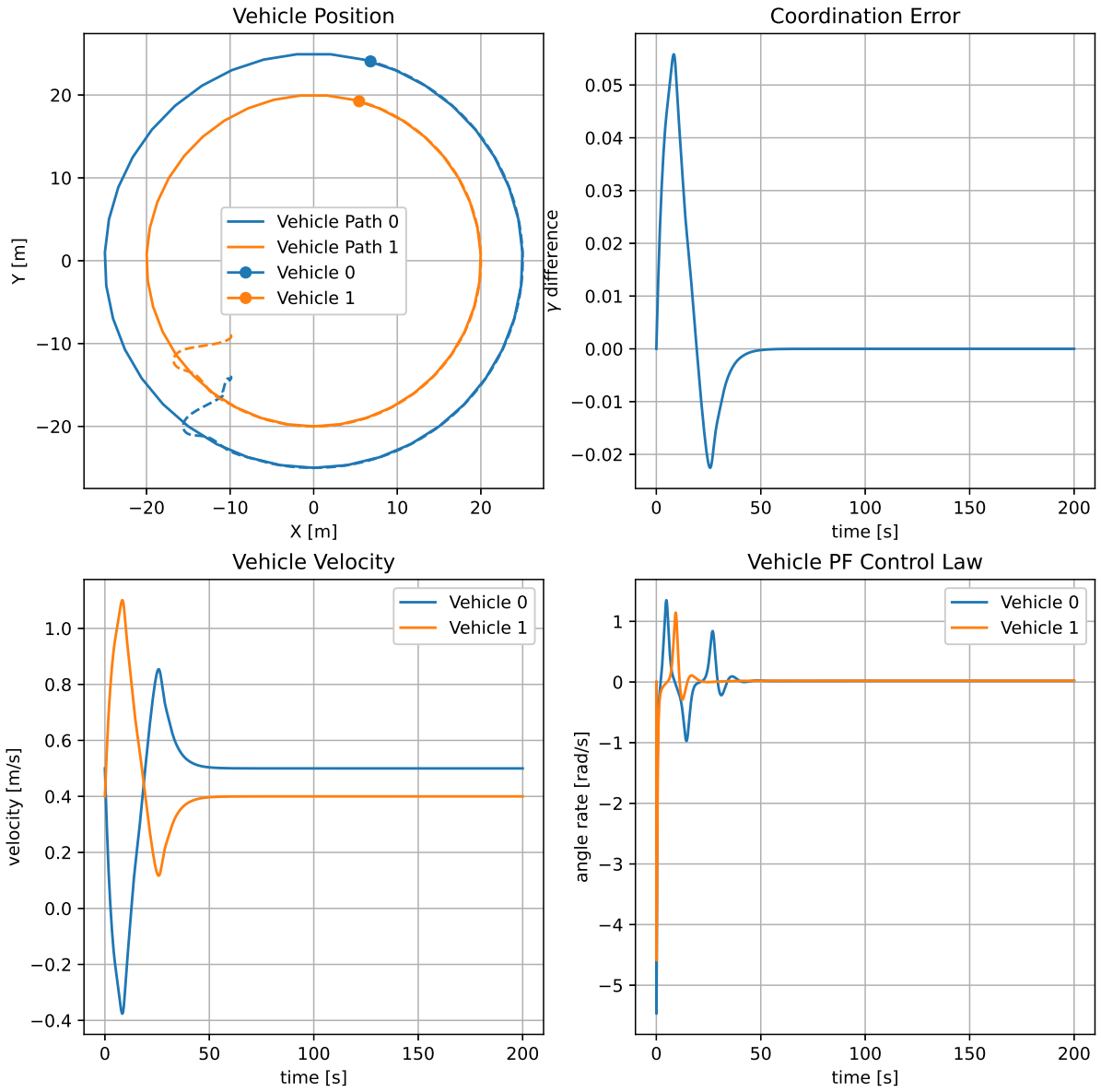


Figure 4.4: Plotting vehicle continuous communications CPF.

denote that from here on out, and for simplicity's sake, the vehicle's velocity derivative with respect to time is assumed to be $\dot{v} = 0$.

4.2 Cooperation with Discrete Communications

The communications between underwater vehicles are a non-trivial problem, there are constraints such as low bandwidth and non-negligible transmission latency making it impossible to have permanent access to the information provided by neighbouring vehicles, as it is explained in Kebkal et al. [26]. This requires the communications to be discrete instead of continuous, and resorting to triggering mechanisms as a means to reduce the frequency of message exchanging by having each vehicle decide on-line, in a decentralised manner, when to exchange information with its neighbours.

After grasping the concepts laid-out in Section 4.1, the transition to discrete communications becomes quite feasible. The estimation of the coordination state γ turns into a necessity, since there is no way of directly knowing the state of neighbouring vehicles without engaging in a difficult exchange of messages.

For this purpose, a new error function is designed, bearing in mind that each vehicle only has access to its true coordination state and the estimates of its neighbours. Consequently, the coordination error ϵ_i for a certain vehicle i from Equation 4.3 becomes

$$\epsilon_i = \gamma_i - \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \hat{\gamma}_j \,, \tag{4.8}$$

where $\hat{\gamma}_j^{[i]}$ is the coordination state estimate for each neighbouring vehicle j, calculated on vehicle i. One more time, it is interesting how the normalised Laplacian $\boldsymbol{L}_D$ can be used to perform the same computation as seen in Equation 4.8. Taking $\boldsymbol{L}_D$ apart, the following expansion of the coordination error vector in Equation 4.4 is obtained as

$$\begin{aligned} \boldsymbol{\epsilon} &= \boldsymbol{D}^{-1}(\boldsymbol{D} - \boldsymbol{A})\boldsymbol{\gamma} \\ &= \boldsymbol{\gamma} - \boldsymbol{D}^{-1} \cdot \boldsymbol{A} \cdot \boldsymbol{\gamma} \,, \end{aligned} \tag{4.9}$$

where $\boldsymbol{D}^{-1}$ is the inverse of the degree matrix and $\boldsymbol{A}$ is the adjacency matrix, defined in Section 4.1.1. The error vector $\boldsymbol{\epsilon}$ is then slightly modified, to accommodate for the coordination state estimators $\hat{\gamma}$, shown as

$$\boldsymbol{\epsilon} = \boldsymbol{\gamma} - \boldsymbol{D}^{-1} \cdot \boldsymbol{A} \cdot \hat{\boldsymbol{\gamma}} \,. \tag{4.10}$$

Considering the new error vector $\boldsymbol{\epsilon}$, Equation 4.6 can be rewritten as

$$\begin{aligned} \boldsymbol{v}_c &= -\boldsymbol{k}_\xi \cdot \tanh(\boldsymbol{\epsilon}) \\ &= -\boldsymbol{k}_\xi \cdot \tanh(\boldsymbol{\gamma} - \boldsymbol{D}^{-1} \cdot \boldsymbol{A} \cdot \hat{\boldsymbol{\gamma}}) \,. \end{aligned} \tag{4.11}$$

If a single vehicle i is contemplated, the velocity correction for that one vehicle can be expressed as

$$v_c^{[i]} = -k_\xi^{[i]} \cdot \tanh\left(\gamma_i^{[i]} - \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \hat{\gamma}_j^{[i]} \right) \,, \tag{4.12}$$

where $\gamma_i^{[i]}$ is the real coordination state of vehicle i and $\hat{\gamma}_j^{[i]}$ is the coordination state estimate calculated within vehicle i of its neighbour j. The parameter $k_\xi^{[i]}$ functions as a tuner for the CPF performance of the vehicle i. In Figure 4.5 it is observed a simple example of a network of vehicles and its neighbour's estimates.

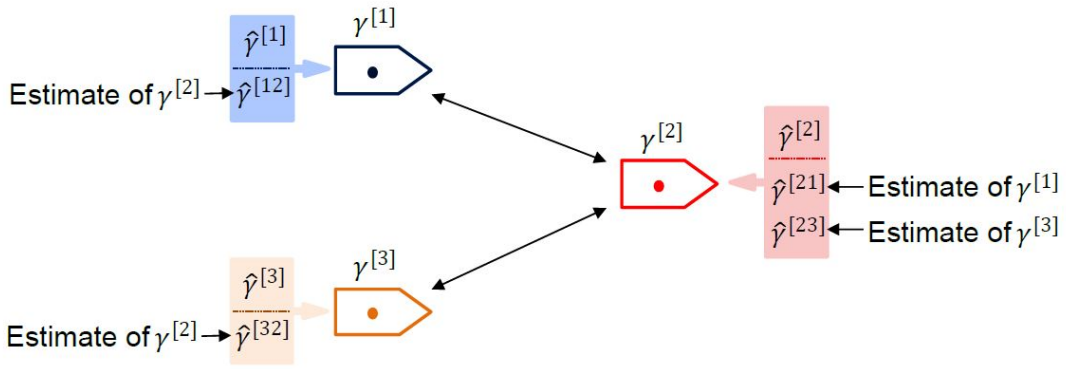


Figure 4.5: Example of a network of vehicles and its neighbours estimates. [10]

4.2.1 Event-Triggered Communications

Having the control for discrete communications laid out, the actual communication layer needs to be set up. To meet this end, the technique of ETC was implemented on every vehicle to ease the coordination and the exchange of messages. A source for these issues is Hung et al. [11], where mathematical proofs and more in-depth descriptions can be found. The fundamental idea behind this method is to have triggering functions which in turn dictate when it is appropriate to broadcast coordination information between vehicles. There are two types of triggering functions, a time-dependent and a state-dependent, as it will be further explained.

The mechanism works in tandem with the coordination state estimators mentioned in Section 4.2, and thus the estimators are finally defined as

$$\begin{aligned} \dot{\hat{\gamma}}_j^{[i]} &= \frac{v_p^j}{L_j}\,, \\ \hat{\gamma}_j^{[i]}(t_k) &= \gamma_j^{[j]}(t_k)\,, \end{aligned} \tag{4.13}$$

where v_p^j is the assigned speed profile assigned to vehicle j, L_j is the total distance of the path assigned to vehicle j, used to normalise the speed profile, and the second line of the Equation 4.13 represents the reception of a new message from a neighbouring vehicle at a given time t_k, updating the coordination state estimator $\hat{\gamma}_j^{[i]}$ of all shared neighbouring vehicles, including vehicle j, assuming no delays in the message exchange.

The error used in the triggering functions is calculated differently from before. It is in fact the difference between the real coordination state γ and an estimate of this coordination state of the actual vehicle i. The error $e(t)$ for a vehicle i is then given by

$$e^{[i]}(t) = \hat{\gamma}_i^{[i]}(t) - \gamma_i^{[i]}(t)\,, \tag{4.14}$$

where $\hat{\gamma}_i$ is the self estimate and γ_i the real coordinator state, for vehicle i. The self estimate is an important feature for this system to come to fruition. It is the result of a self coordination sate estimator

which runs along with the other neighbour estimators. The estimator is then governed by the model

$$\begin{aligned} \dot{\hat{\gamma}}_i^{[i]} &= \frac{v_p^i}{L_i}\,, \\ \hat{\gamma}_i^{[i]}(t_k) &= \gamma_i^{[i]}(t_k)\,, \end{aligned} \tag{4.15}$$

where v_p^i is the assigned speed profile assigned to vehicle i, L_i is the total distance of the path assigned to vehicle i, used to normalise the speed profile, and the second line of the Equation 4.15 represents the broadcast of a new message to every neighbouring vehicle at a given time t_k, resetting the self coordination state estimator $\hat{\gamma}_i^{[i]}$.

The triggering functions can now be introduced, whereas the time-dependent one is expressed as

$$\begin{aligned} h^{[i]}(t) &= |e^{[i]}(t)| - \eta^{[i]}(t)\,, \\ \eta^{[i]}(t) &= c_0 + c_1 \cdot e^{-c_2 \cdot t}\,, \end{aligned} \tag{4.16}$$

where $\eta(t)$ is the element which determines if the error $e(t)$ is considerable enough at a given time to trigger a broadcast from vehicle i to its neighbours. In other words, if $h(t) \geq 0$ then vehicle i broadcasts its information to its neighbours, while also updating the its **self** estimator. The constants c_0, c_1 and c_2 are tuning parameters so that the system performs at its best, coordinating with the other vehicles while exchanging as few messages as possible.

The other state-dependent triggering function is formulated as

$$h^{[i]}(t) = |e^{[i]}(t)|^2 - \left(\theta^{[i]} \frac{\sigma}{\sigma^{[i]}} \sum^{j \in \mathcal{N}_i} a_{ij} \left(\hat{\gamma}^{[i]}(t) - \hat{\gamma}^{[ij]}(t) \right)^2 + \epsilon_0 \right)\,, \tag{4.17}$$

where a_{ij} represents the entries in the adjacency matrix A defined previously. The other constants ($\theta^{[i]}$, σ, $\sigma^{[i]}$ and ϵ_0) are well-determined in Hung et al. [11]. In the next section, the performance of each triggering function is evaluated and put against the results from the time-continuous case in the Section 4.1.

4.2.2 Simulation Results

The block diagram in Figure 4.6 does not differ much from the continuous communications CPF presented in Figure 4.3. However, it accounts for the new discrete communications stipulated in this section, as it can be seen in Figure 4.6 represented by the dashed line between the *Vehicle Network* and the rest of the system.

The simulation results are displayed in Figure 4.7. By observing the Vehicle Position plot and the Coordination Error plot, it is possible to verify that coordination between vehicles was achieved with the ETC mechanism, producing a total of 107 Broadcasts between the two vehicles, shown in the Vehicle Broadcasting plot. The final velocity of each velocity is also the constant speed profile v_p assigned to each vehicle.

The values used for the ETC parameters are registered in Table 4.1.

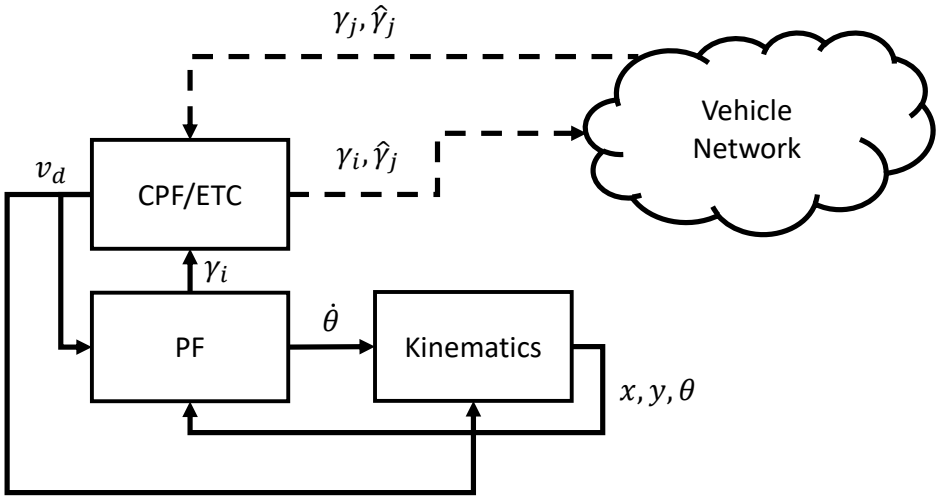


Figure 4.6: CPF system block diagram with ETC mechanism.

Table 4.1: Values for the parameters used in the ETC mechanism.

c_0	c_1	c_2	$k_\xi^{[0]}$	$k_\xi^{[1]}$
0.01	0.5	1	0.4	0.32

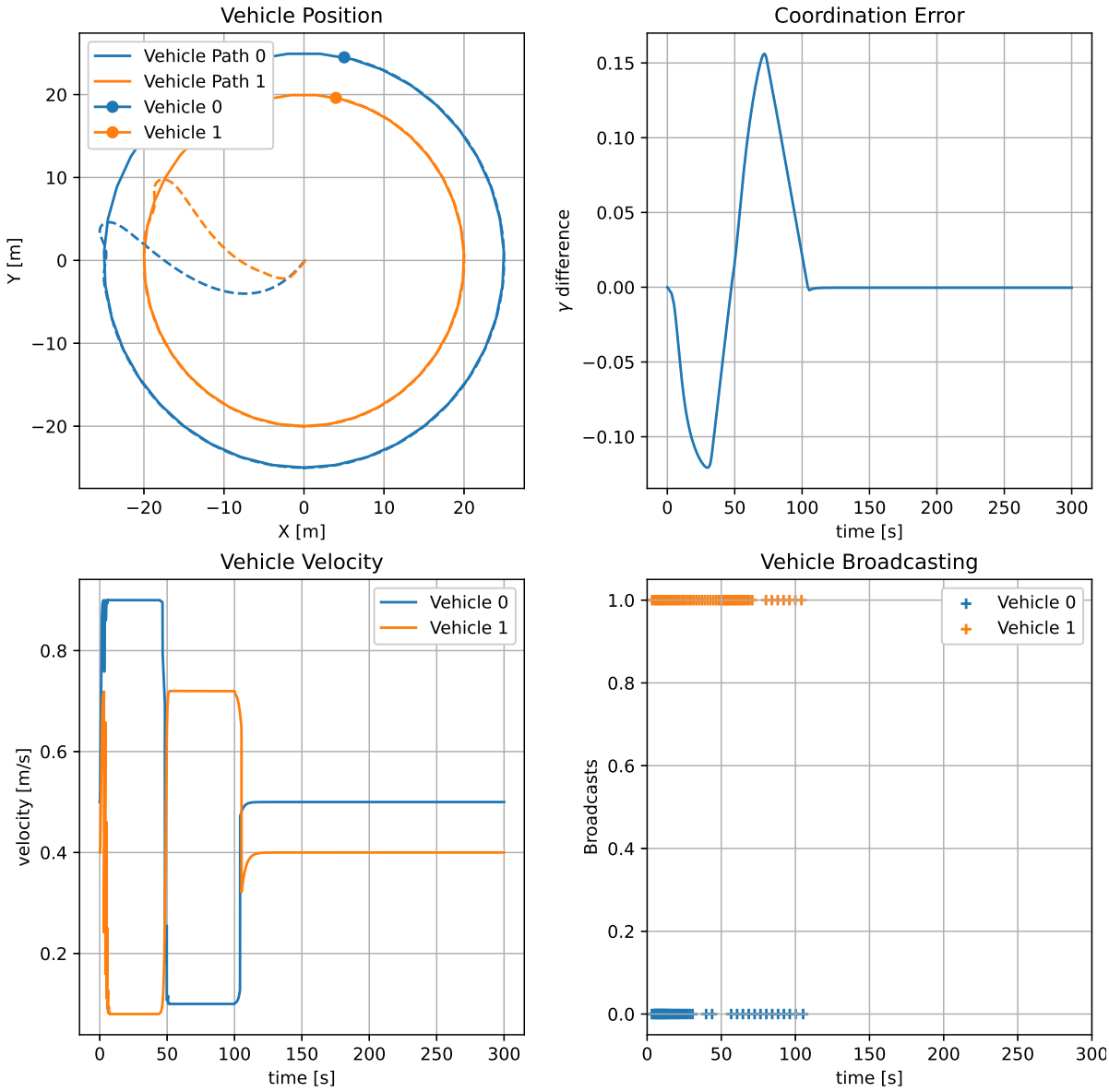


Figure 4.7: Plotting simulation of vehicle discrete communications CPF with ETC mechanism.

Chapter 5

Moving Path Following

After tackling CPF, while also allowing for ETC in order to handle the underwater exchange of messages, MPF comes as the next step. This technique was presented in Oliveira et al. [12], initially introduced for aerial vehicles, and then adapted for AMVs in Teixeira [6]. As stated in Section 1.2, the MPF is crucial so that the ASVs perform the encircling manoeuvre on the underwater group of target AUVs. A path is envisioned on top of the group of targets, moving along with them. The ASVs or trackers then shall perform PF strategies on said path.

To this end, a reference frame is placed on top of the target. In the context of this work, the 2D assumption is revisited, putting the trackers and targets in the same plane. The circular path lays static with reference to the frame, which in turn moves along, accompanying the target. Therefore, the realisation of this method has two routes, where the first considers a simple translation of the frame of reference to the target in question, and the second goes a step further taking into account the orientation of the target and thus the frame of reference itself.

5.1 Non-Rotating Path

The case of MPF with non-rotating path is fairly simple to design. The concept is illustrated in Figure 5.1, where it is possible to see the vector forms of the points of interest.

By analysing Figure 5.1, it can be deduced that the tracker's position in the inertial frame $\{\mathcal{I}\}$ is then expressed as

$$^{\mathcal{I}}\boldsymbol{p} = {}^{\mathcal{I}}\boldsymbol{O}_P + {}^{\mathcal{I}}_{\mathcal{P}}\boldsymbol{R} \cdot {}^{\mathcal{P}}\boldsymbol{p}\,, \tag{5.1}$$

where $^{\mathcal{I}}\boldsymbol{p}$ is the tracker's position in the inertial frame $\{\mathcal{I}\}$, $^{\mathcal{I}}\boldsymbol{O}_P$ is the target's position and moving origin of the reference frame $\{\mathcal{P}\}$, and $^{\mathcal{P}}\boldsymbol{p}$ is the tracker's position, also expressed in the reference frame $\{\mathcal{P}\}$. The rotation matrix ${}^{\mathcal{I}}_{\mathcal{P}}\boldsymbol{R}$ from the reference frame $\{\mathcal{P}\}$ to the inertial frame $\{\mathcal{I}\}$ can be ignored for now, considering the angle $\theta = 0$ and is constant, for there is no rotation in this case, thus the Equation 5.1 can be rewritten as

$$^{\mathcal{I}}\boldsymbol{p} = {}^{\mathcal{I}}\boldsymbol{O}_P + {}^{\mathcal{P}}\boldsymbol{p}\,. \tag{5.2}$$

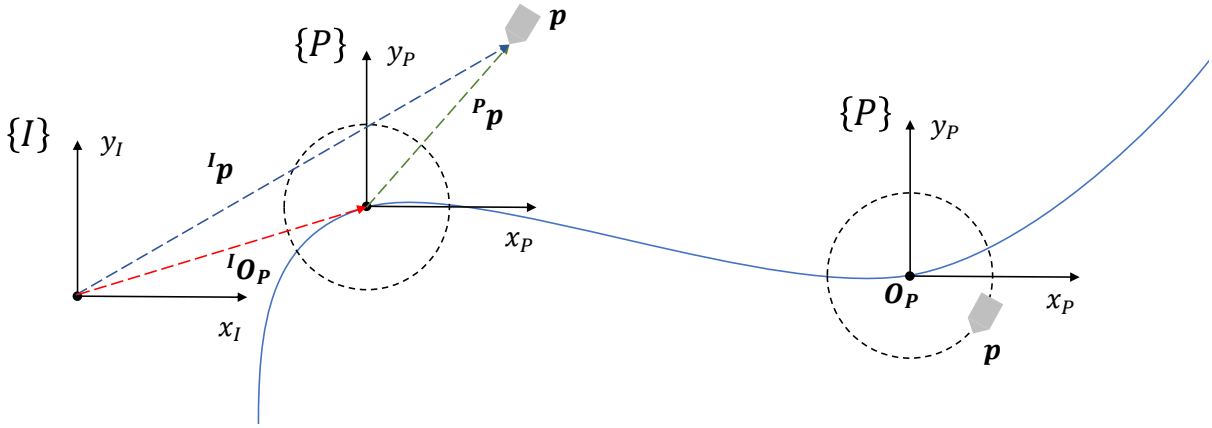


Figure 5.1: Moving Path Following illustration for non-rotating path.

Taking all these equations into consideration, the constructed MPF law can be formulated simply by differentiating the tracker's position in the inertial frame $\{\mathcal{I}\}$ as

$$ {}^{\mathcal{I}}\dot{\boldsymbol{p}} = {}^{\mathcal{I}}\boldsymbol{v}_P + {}^{\mathcal{P}}\boldsymbol{v}\,, \tag{5.3} $$

where ${}^{\mathcal{I}}\boldsymbol{v}_P$ is the velocity of the moving target or origin of the reference frame $\{\mathcal{P}\}$ represented in the inertial frame $\{\mathcal{I}\}$, and ${}^{\mathcal{P}}\boldsymbol{v}$ is the velocity of the tracker in the reference frame $\{\mathcal{P}\}$. This way, it is possible to extract the desired surge velocity and yaw from ${}^{\mathcal{I}}\dot{\boldsymbol{p}}$ in Equation 5.3.

5.2 Rotating Path

The rotating path case adds a layer of complexity to the previous problem, where a rotation angle between the reference frame $\{\mathcal{P}\}$ and the inertial frame $\{\mathcal{I}\}$ is considered, as it can be observed in Figure 5.2.

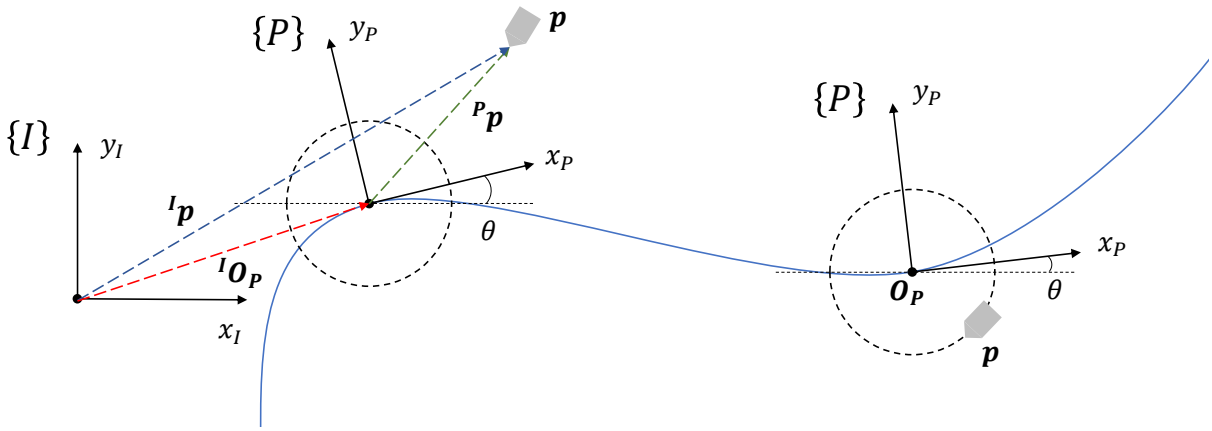


Figure 5.2: Moving Path Following illustration for rotating path.

The rotation matrix ${}^{\mathcal{I}}_{\mathcal{P}}\boldsymbol{R}$ from reference frame $\{\mathcal{P}\}$ to the inertial frame $\{\mathcal{I}\}$ that was formerly consid-

ered is no longer negligible, since the rotation angle θ is no longer equal to zero, thus being denoted as

$$ {}^{\mathcal{I}}_{\mathcal{P}}\boldsymbol{R} = \begin{bmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{bmatrix} . \tag{5.4} $$

To determine the new control law for the desired velocity expressed in both xy axis components, the Equation 5.1 is revisited. If differentiated again with respect to time, it becomes

$$ {}^{\mathcal{I}}\dot{\boldsymbol{p}} = {}^{\mathcal{I}}\boldsymbol{v}_P + {}^{\mathcal{I}}_{\mathcal{P}}\dot{\boldsymbol{R}} \cdot {}^{\mathcal{P}}\boldsymbol{p} + {}^{\mathcal{I}}_{\mathcal{P}}\boldsymbol{R} \cdot {}^{\mathcal{P}}\boldsymbol{v} \,, \tag{5.5} $$

where ${}^{\mathcal{I}}_{\mathcal{P}}\dot{\boldsymbol{R}}$ is the derivative of the rotation matrix ${}^{\mathcal{I}}_{\mathcal{P}}\boldsymbol{R}$ with respect to time. The derivative of the rotation matrix is simply defined as ${}^{\mathcal{I}}_{\mathcal{P}}\dot{\boldsymbol{R}} = {}^{\mathcal{I}}_{\mathcal{P}}\boldsymbol{R} \cdot S(\boldsymbol{\omega}_p)$, where $\boldsymbol{\omega}_P$ is the angular velocity vector of the angle between frames. Substituting this new information in Equation 5.5, and in light of the skew matrix property regarding multiplication with vectors $S(\boldsymbol{a}) \cdot \boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$, the previous result can be denoted as the cross product between the angular velocity $\boldsymbol{\omega}_p$ and the position of the tracker ${}^{\mathcal{P}}\boldsymbol{p}$ expressed in the reference frame $\{\mathcal{P}\}$. All things considered, the Equation 5.5 can be reworked and hence given as

$$ {}^{\mathcal{I}}\dot{\boldsymbol{p}} = {}^{\mathcal{I}}\boldsymbol{v}_P + {}^{\mathcal{I}}_{\mathcal{P}}\boldsymbol{R} \cdot \boldsymbol{\omega}_P \times {}^{\mathcal{P}}\boldsymbol{p} + {}^{\mathcal{I}}_{\mathcal{P}}\boldsymbol{R} \cdot {}^{\mathcal{P}}\boldsymbol{v} \,, \tag{5.6} $$

where $\boldsymbol{\omega}_P \times {}^{\mathcal{P}}\boldsymbol{p}$ represents the cross product between the angular velocity $\boldsymbol{\omega}_P = [0, 0, \omega_P]^T$ of the reference frame $\{\mathcal{P}\}$ and the position of the tracker expressed in the same reference frame $\{\mathcal{P}\}$. Notice the representation of the vectors in three dimensions even though this is a 2D case. This is because the cross product in two dimensions does not technically exist, although it is considered that the reference frame is rotating around a third axis, in an intuitive way, pointed at the person looking at 2D example. After the calculation, the result of the cross product is truncated back to a two dimensional vector, discarding the third coordinate.

Finally, as it was explained in Section 5.1, the surge velocity and yaw can likewise be extracted from the control law in Equation 5.6.

5.3 Simulation Results

For the sake of simplicity, the implementation only considered the case of a non-rotating path, seeing that the system described in Section 1.2 does not require a rotating path for the described encircling manoeuvre. The series of steps described previously can be represented by the block diagram in Figure 5.3, where the core of the MPF strategy is illustrated along with the rest of the system implemented up until now.

5.3.1 Simple Moving Path Following

The simulation of a single tracker encircling a target is presented in Figure 5.4, where the error between the tracker's virtual target and the tracker itself converges to zero in the Tracker PF Error plot, indicating

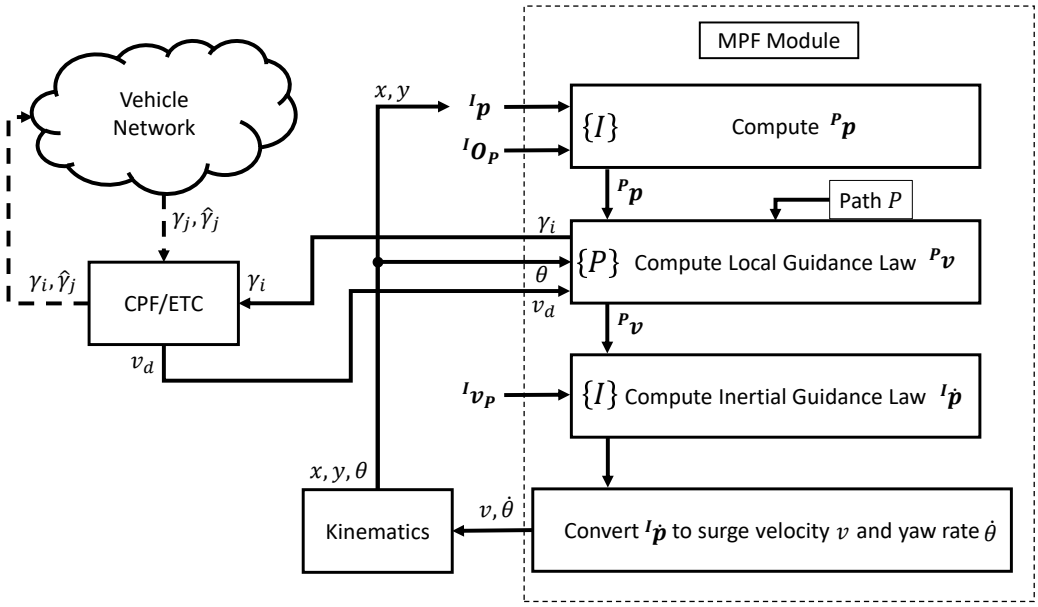


Figure 5.3: MPF system block diagram.

the MPF success around a moving path fixed on top of a target, as confirmed in the Vehicle Position plot.

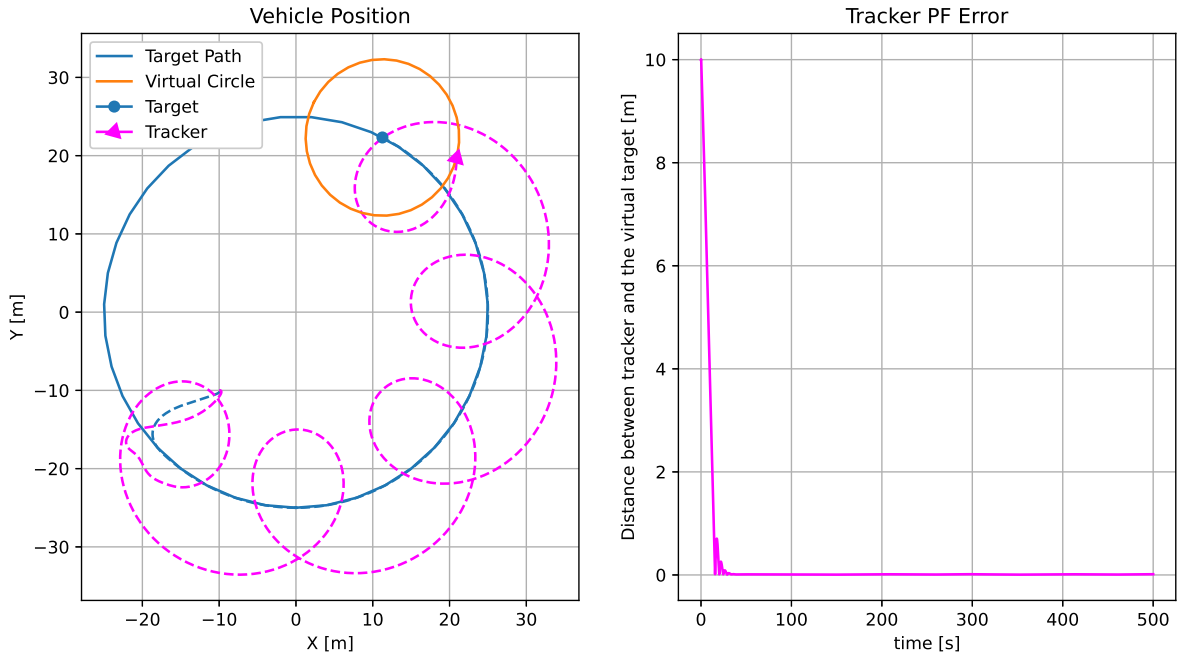


Figure 5.4: Plotting vehicle position and PF Error of the simple MPF simulation.

5.3.2 Cooperative Moving Path Following

The simulation displayed in Figure 5.5 consists on two ASV trackers cooperating and performing the MPF technique around a group of AUV targets, which in turn are cooperating internally within their own group of AUVs. The ETC broadcasts performed locally between AUVs, and locally between ASVs is

presented in the Vehicle Broadcasting plot, altogether making up for 46 broadcasts between AUVs, and 54 broadcasts between ASVs. It is also possible to see that, in the Coordination States plot, the trackers eventually achieve coordination. The coordination states are varying between zero and one cyclically due to the circular motion executed in the MPF. The two trackers are able to maintain the ninety degree angle between them because of their path set-up. At the beginning of the mission, while both circular paths, relative to the **Tracker 0** and **Tracker 1**, sit on top of each other, the paths are defined so that one starts ninety degrees after the other, effectively creating this formation.

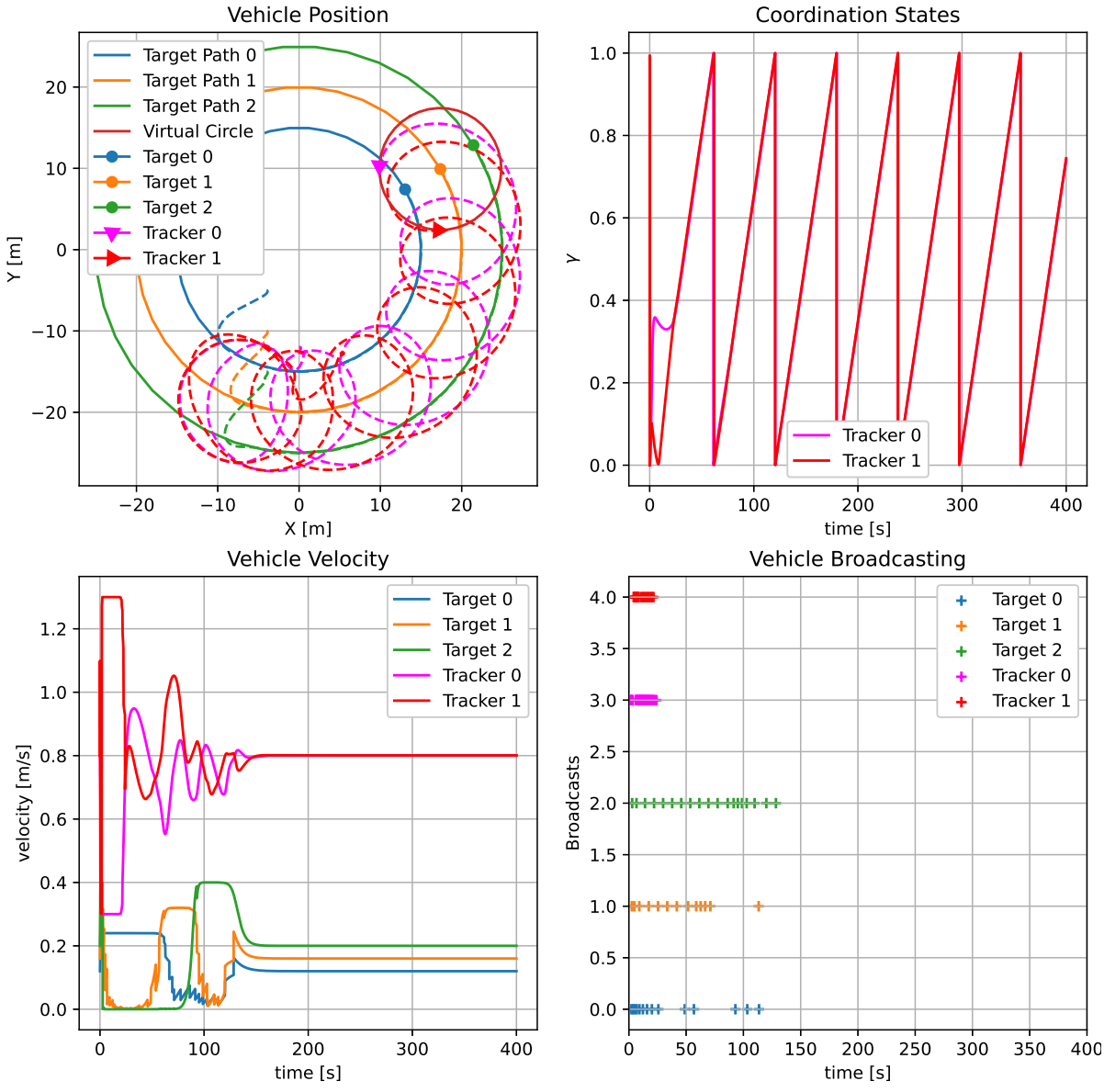


Figure 5.5: Plotting the simulation results of the CMPF system.

Chapter 6

Filter Application

The previous sections describe the primary issues of the envisioned system, the control and cooperation of each vehicle on the network. However, the system presented in Section 5.3.2 is far from an accurate portrayal of the real case. For instance, it is assumed the position of the targets, or AUVs, are known at all times by the trackers, or ASVs, which is not true since underwater localisation is not trivial. Another example of an unrealistic assumption is the AUV themselves knowing their true position underwater at any point in time. There are countless disturbances and constraints in these environments which make it very difficult to know an exact or, at least, good estimate of a position underwater. With this objective in mind, two Kalman Filters (KF) were thought: an Extended Kalman Filter (EKF) implemented on a tracker ASV to estimate the position of a target utilising periodic range measurements; and a Complementary Kalman Filter (CKF) implemented on a target AUV, which in turn receives the EKF estimate of the tracker ASV to correct its position estimate, calculated from velocity measurements performed by a DVL. A sketch of the system is presented in Figure 6.1.

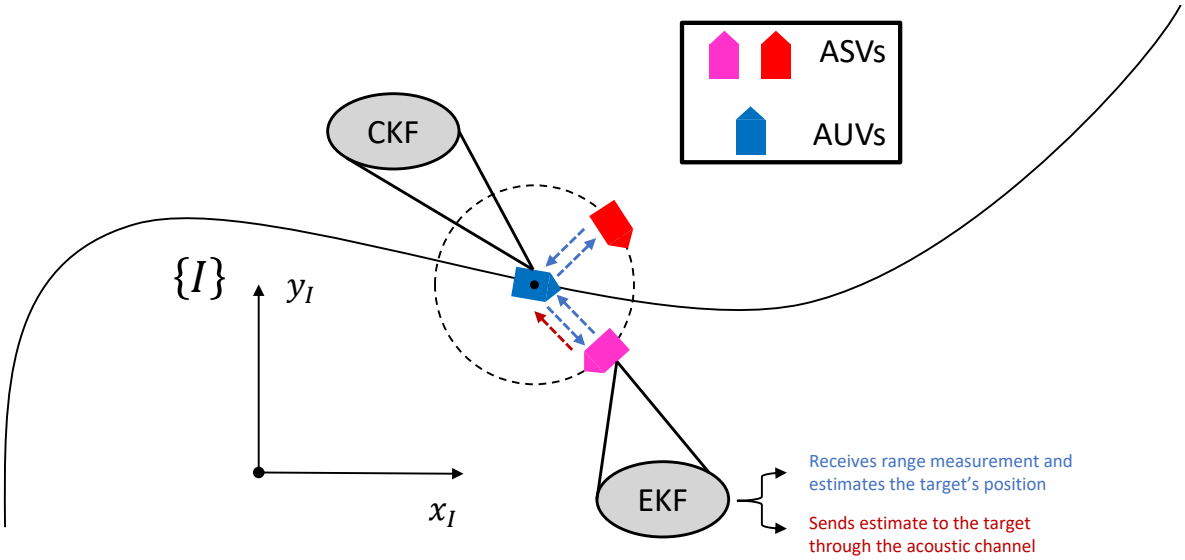


Figure 6.1: Illustration of the filter application.

6.1 Extended Kalman Filter

The structure of this filter draws from the work in Hung et al. [13], where a combination of linear and nonlinear filtering techniques were applied, in other words, a mix of regular KF (linear) and EKF (nonlinear). It is also explained in the paper that an improvement to the estimation is increased if an ninety degree (90°) angle is formed between the two ASV trackers during the encircling manoeuvre. It starts with a linear approach to KF, where a process matrix $\boldsymbol{A}$ and an input matrix $\boldsymbol{B}$ are defined. The linear system is then given as

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A} \cdot \boldsymbol{x}(t) + \boldsymbol{B} \cdot \boldsymbol{u}(t)\,, \tag{6.1}$$

where $\boldsymbol{x}$ stands for the state of the KF and $\boldsymbol{u}$ the input. For the system in question, $\boldsymbol{x} = [x, y, v_x, v_y]^T$, where x, y are the components of the position of the target, and v_x, v_y are the components of the velocity of the target; it is also assumed there are no inputs $\boldsymbol{u}$, so the second term of Equation 6.1 is ignored. The process is assumed to be corrupted by white Gaussian noise $\boldsymbol{w} \sim \mathcal{N}(0, \boldsymbol{Q}_t)$, where $\boldsymbol{Q}_t$ is the variance of said noise, in order to represent deviations which may happen inside the whole procedure. The system model can the be expressed as

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A} \cdot \boldsymbol{x}(t) + \boldsymbol{w}(t) \quad , \quad \boldsymbol{A} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}. \tag{6.2}$$

Another relevant assumption is to have the KF in discrete time, instead of continuous. Using Euler's Method, Equation 6.2 can be discretised and turns into

$$\boldsymbol{x}_{k+1} = \boldsymbol{F} \cdot \boldsymbol{x}_k + \boldsymbol{w}_k\,, \tag{6.3}$$

where

$$\boldsymbol{F} = \begin{bmatrix} \boldsymbol{I}_2 & T_s \cdot \boldsymbol{I}_2 \\ \boldsymbol{0} & \boldsymbol{I}_2 \end{bmatrix}, \tag{6.4}$$

being the discretised matrix version of the process matrix $\boldsymbol{A}$, with T_s being the sampling period and $\boldsymbol{w}_k \sim \mathcal{N}(0, \boldsymbol{Q} = T_s \cdot \boldsymbol{Q}_t)$ the process noise in the discretised system.

On the other hand, the system output, which consists on the nonlinear part of this KF, making it an EKF at its core, is given by

$$d_k = ||\boldsymbol{p}_k - \boldsymbol{q}_k||\,, \tag{6.5}$$

where d_k is the distance between the target's position $\boldsymbol{p}_k = [p_x, p_y]^T$ and the tracker's position $\boldsymbol{q}_k = [q_x, q_y]^T$, which in turn performs the range measurements y_k on the target. Since these measures are considered to be noise corrupted distances between the two vehicles, they can be defined as

$$y_k = d_k + \eta_k\,, \tag{6.6}$$

where $\eta_k \sim \mathcal{N}(0, R)$ is a Gaussian noise associated with the range measurement sensor installed on the trackers, with variance R. This is especially useful when creating the range measurement simulation.

The EKF can then start to be assembled, where two distinct stages must be defined: a **prediction** and an **update**. For an in-depth review on this subject, the report regarding KFs documented in Ribeiro [15] offers the intuitive guidance and subsequent proof for the comprehension of this matter. The prediction stage can hence be described as

$$\begin{cases} \hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{F} \cdot \hat{\boldsymbol{x}}_{k-1|k-1}\,, \\ \boldsymbol{P}_{k|k-1} = \boldsymbol{F} \cdot \boldsymbol{P}_{k-1|k-1} \cdot \boldsymbol{F}^T + \boldsymbol{Q}\,, \end{cases} \tag{6.7}$$

where $\hat{\boldsymbol{x}}_{k-1|k-1}$ is the state estimate before the prediction, $\hat{\boldsymbol{x}}_{k|k-1}$ is the state estimate after the prediction, $\boldsymbol{P}_{k-1|k-1}$ is the probability matrix before the prediction, and $\boldsymbol{P}_{k|k-1}$ is the probability matrix after the prediction. This probability matrix is the essential part of the EKF, as it is passed through prediction and update at each timestamp t_k, it represents the confidence of the estimate at each stage. As it is observed in Equation 6.7, the prediction portion of the EKF follows a linear-based model.

In contrast with the prediction stage, the update part of the EKF is nonlinear-based, since it deals with the output of the system, which is calculated nonlinearly as seen in Equation 6.5. The update stage can thus be denoted as

$$\begin{cases} d_k(\hat{\boldsymbol{x}}_{k|k-1}) = ||\hat{\boldsymbol{p}} - \boldsymbol{q}|| = \sqrt{(\hat{p}_x - q_x)^2 + (\hat{p}_y - q_y)^2}\,, \\ \boldsymbol{C}_k = \frac{d}{d\boldsymbol{x}} d_k(\hat{\boldsymbol{x}}_{k|k-1})\,, \\ \boldsymbol{K}_k = \boldsymbol{P}_{k|k-1} \cdot \boldsymbol{C}_k^T (\boldsymbol{C}_k \cdot \boldsymbol{P}_{k|k-1} \cdot \boldsymbol{C}_k^T + R)^{-1}\,, \\ \hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k (y_k - d_k(\hat{\boldsymbol{x}}_{k|k-1}))\,, \\ \boldsymbol{P}_{k|k} = (\boldsymbol{I}_4 - \boldsymbol{K}_k \cdot \boldsymbol{C}_k) \cdot \boldsymbol{P}_{k|k-1}\,, \end{cases} \tag{6.8}$$

where $d_k(\hat{\boldsymbol{x}}_{k|k-1})$ is the distance between the tracker's position $\boldsymbol{q}$ and the estimated position of the target $\hat{\boldsymbol{p}} = [\hat{p}_x, \hat{p}_y]^T = [\hat{\boldsymbol{x}}^x_{k|k-1}, \hat{\boldsymbol{x}}^y_{k|k-1}]^T$, $\boldsymbol{C}_k$ is the derivative of $d_k(\hat{\boldsymbol{x}}_{k|k-1})$ with respect to the state $\boldsymbol{x}$, $\boldsymbol{K}_k$ is the Kalman gain calculated on each update, $\hat{\boldsymbol{x}}_{k|k}$ is the state estimate after the update, and $\boldsymbol{P}_{k|k}$ is the probability matrix after the update. A feature worth noting is when there is no new range measurement y_k, which translates in the Kalman gain $\boldsymbol{K}_k = \boldsymbol{0}$, which in turn means, in an intuitive way, that the state is equal to the predicted $\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1}$, as well as the probability matrix $\boldsymbol{P}_{k|k} = \boldsymbol{P}_{k|k-1}$.

If the case of multiple range measurements is considered, the system presented in Equation 6.8 changes slightly, where the measures $\boldsymbol{y}_k$, the distances $\boldsymbol{d}_k(\hat{\boldsymbol{x}}_{k|k-1})$ and the variance matrix $\boldsymbol{R}$ are not scalars anymore, having the dimension corresponding to the number of range measurements. For instance, with two measures, $\boldsymbol{y}_k$ is 2×1, $\boldsymbol{d}_k(\hat{\boldsymbol{x}}_{k|k-1})$ is 2×1, taking into account that the position of the tracker $\boldsymbol{q}$ is dependent on the tracker which effectively performed the range measurement, $\boldsymbol{R}$ is 2×2, and $\boldsymbol{C}_k$ also changes and becomes 2×4.

6.1.1 Simulation Results

The proposed EKF interacts with the rest of the ASV tracker's system just as shown in the block diagram in Figure 6.2. At this step, a target pursuit is performed by feeding the EKF estimate of the target's

position and velocity to the tracker's MPF mechanism, bringing the system closer to the real case.

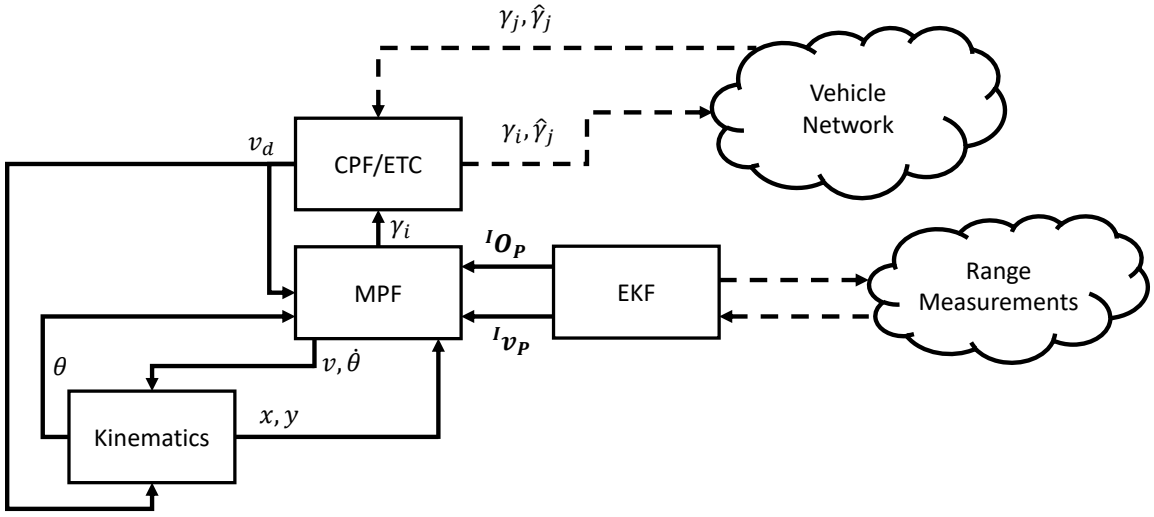


Figure 6.2: EKF included in the overall system represented in block diagram.

The results of the simulation are registered in the plots from Figure 6.3. It can be observed that the EKF estimate is successful, as the EKF Error plot, which represents the distance between the target's actual position and the EKF estimate, is very close to zero. After acquiring the target pursuit motion with a plausible EKF estimate, the range measurements stay on values near the seven meters, which in this case translates to the radius of the virtual circle. Finally, the Vehicle Position plot shows how the EKF estimate remains on top of the target throughout the mission.

The values for the variances used in this EKF are

$$\boldsymbol{Q} = 10^{-6} \cdot \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \boldsymbol{R} = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}. \tag{6.9}$$

The probability matrix $\boldsymbol{P}_{k-1|k-1}$ of the filter is initialised as the variance of the Gaussian noise corrupting the process $\boldsymbol{Q}$.

There might be some cases where the position estimate diverges from the real position at the start of the mission. A solution to this problem can be the pre-estimation of the vehicle before the mission starts. This means that before the target starts performing the PF strategy, the trackers converge around the target to better estimate it before starting the actual mission.

6.2 Complementary Kalman Filter

As stated before, underwater localisation is not a trivial problem, making it a challenge when a vehicle only has its onboard sensors to rely on, as many disturbances occur during their mission. This filter is

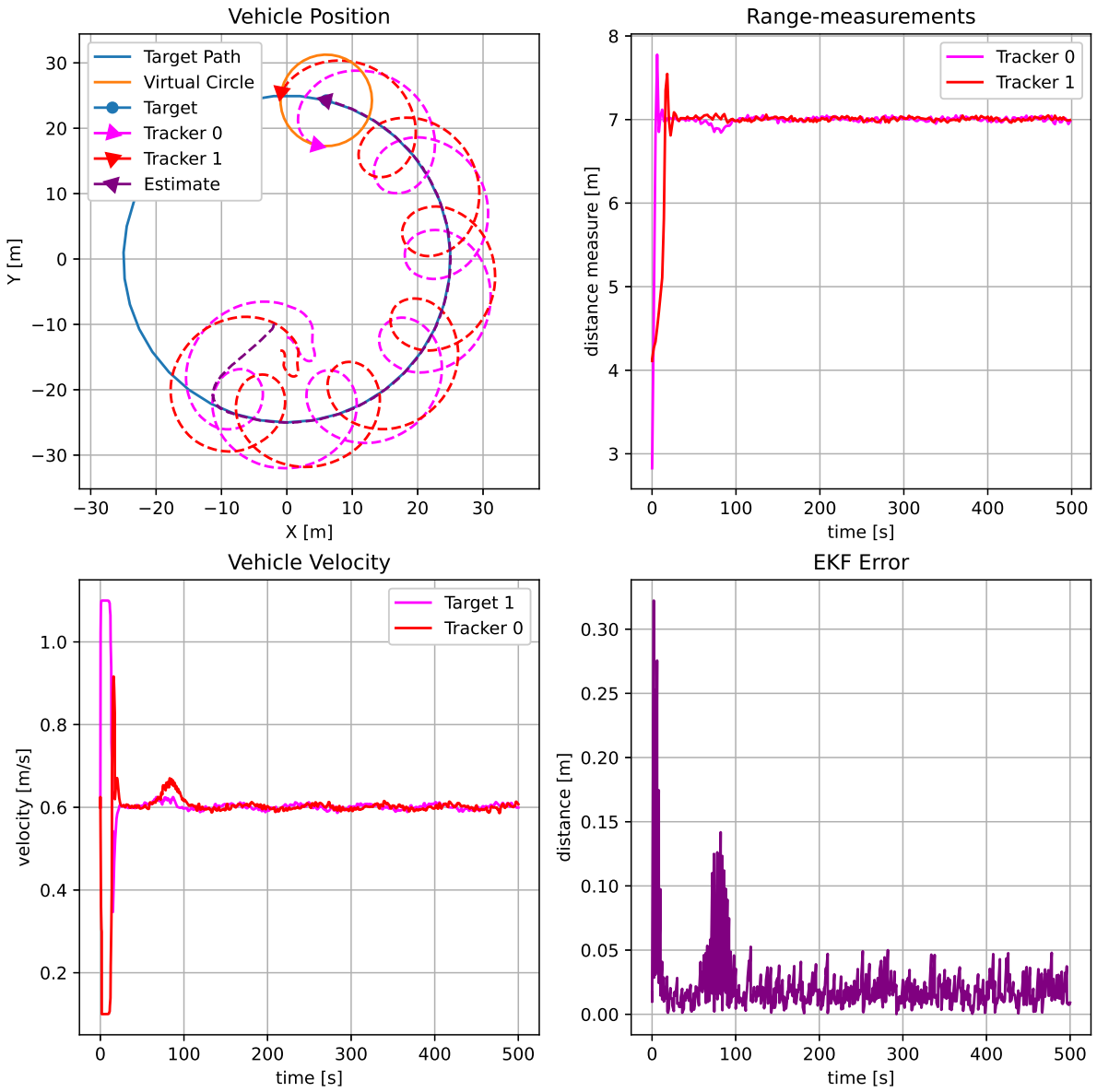


Figure 6.3: Plotting the simulation results of the EKF implemented in the system.

then purposed for the target AUVs, receiving the EKF estimate from the tracker ASVs and thus complementing its own estimate of their position. To this end, another KF type of filter is implemented on the target. The received EKF estimate is dealt like a regular sensor on the vehicle, yet with a very low work frequency. To achieve this, not only does the vehicle receive the EKF estimate, but also the probability matrix of the EKF at that point in time.

However, the approach to the design of this filter is different from the usual KF sensor fusion concept. To avoid having a more complex structure and a needless amount of Kalman gains to compute, a Complementary Filter approach was taken (see Kaminer et al. [14] for a more detailed explanation on this topic), where the velocity measures from the DVL sensor were considered inputs, as seen in Figure 6.4. The estimation model then becomes similar to the one used by an observer in classical linear control theory.

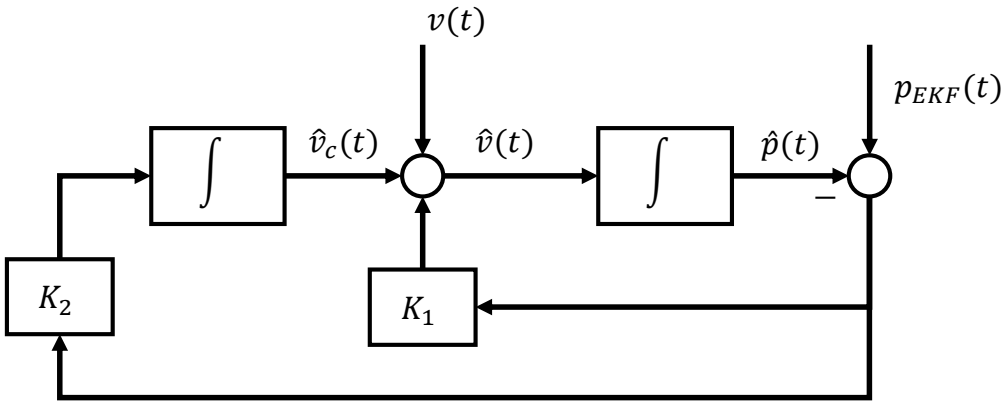


Figure 6.4: Complementary Filter block diagram.

The model illustrated in Figure 6.4, still in time-continuous, can then be defined as

$$\begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A} \cdot \boldsymbol{x}(t) + \boldsymbol{B} \cdot \boldsymbol{v}(t) + K(\boldsymbol{y}_m(t) - \hat{\boldsymbol{y}}(t)) \,, \\ \hat{\boldsymbol{y}}(t) &= \boldsymbol{C} \cdot \boldsymbol{x}(t) \,, \end{cases} \tag{6.10}$$

where

$$\boldsymbol{x} = \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{v}_c^x \\ \hat{v}_c^y \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I}_2 \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} \boldsymbol{I}_2 \\ \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{C} = \begin{bmatrix} \boldsymbol{I}_2 & \boldsymbol{0} \end{bmatrix}, \quad \boldsymbol{K} = \begin{bmatrix} \boldsymbol{K}_1 \\ \boldsymbol{K}_2 \end{bmatrix}, \quad \boldsymbol{y}_m = \boldsymbol{p}_{EKF} \,, \tag{6.11}$$

while

$$\boldsymbol{K}_1 = \begin{bmatrix} K_1^x & 0 \\ 0 & K_1^y \end{bmatrix}, \quad \boldsymbol{K}_2 = \begin{bmatrix} K_2^{v_c^x} & 0 \\ 0 & K_2^{v_c^y} \end{bmatrix} . \tag{6.12}$$

The gains in matrix $\boldsymbol{K}_1$ and matrix $\boldsymbol{K}_2$, according to the work in Kaminer et al. [14], can be deduced using classical control theory methods, such as analysing the second order system and calculating gains for an adequate pair of damping ξ and natural frequency ω_n factors. In contrast to what was determined in Section 6.1, the new state $\boldsymbol{x}$ consists on the position $\boldsymbol{p} = [x, y]^T$ of the vehicle itself, and also the current velocity $\boldsymbol{v_c} = [v_c^x, v_c^y]^T$, so that the system accounts for some bias in the estimation. Now that the Complementary Filter is defined, the next step is the conversion to a CKF, having time-variant gains instead of fixed ones.

As stated before, the application of the velocity as an input is what simplifies the KF. Moreover, the measuring rate of the sensor is quite high, so it is assumed there is a new velocity input at every time instance t_k, in the practical discretised version of the filter. In light of all this, during the design of the new system model represented in Figure 6.4, the process not only has some associated noise $\boldsymbol{w} \sim \mathcal{N}(0, \boldsymbol{W})$, but also must now account for the velocity measurement noise $\boldsymbol{\nu} \sim \mathcal{N}(0, \boldsymbol{N})$ as well. Since these two noises are considered independent, the variance to be used in the KF associated with the process noise

can be given by the sum of the two variances

$$\boldsymbol{W}_{\nu} = \boldsymbol{W} + \begin{bmatrix} \boldsymbol{N} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} , \tag{6.13}$$

where $\boldsymbol{W}_{\nu}$ is the new covariance related to the process, $\boldsymbol{W}$ is the 4×4 covariance matrix of the process without taking into account the velocity measurement noise, and $\boldsymbol{N}$ is the 2×2 covariance matrix related to the velocity measurement noise. Similar to the previous Section 6.1, the prediction stage of the CKF is defined as

$$\begin{cases} \hat{\boldsymbol{x}}_{k|k-1} = \boldsymbol{F} \cdot \hat{\boldsymbol{x}}_{k-1|k-1} + \boldsymbol{E} \cdot \boldsymbol{v}_k \, , \\ \boldsymbol{P}_{k|k-1} = \boldsymbol{F} \cdot \boldsymbol{P}_{k-1|k-1} \cdot \boldsymbol{F}^T + \boldsymbol{W}_{\nu} \, , \end{cases} \tag{6.14}$$

where

$$\boldsymbol{F} = T_s \cdot \boldsymbol{A}, \quad \boldsymbol{E} = T_s \cdot \boldsymbol{B} \, , \tag{6.15}$$

and T_s is the sampling period. The update stage of the CKF is translated as

$$\begin{cases} \boldsymbol{K}_k = \boldsymbol{P}_{k|k-1} \cdot \boldsymbol{C}^T (\boldsymbol{C} \cdot \boldsymbol{P}_{k|k-1} \cdot \boldsymbol{C}^T + \boldsymbol{P}_{EKF})^{-1} \, , \\ \hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k (\boldsymbol{p}_{EKF} - \boldsymbol{C} \cdot \boldsymbol{x}) \, , \\ \boldsymbol{P}_{k|k} = (\boldsymbol{I}_4 - \boldsymbol{K}_k \cdot \boldsymbol{C}) \cdot \boldsymbol{P}_{k|k-1} \, , \end{cases} \tag{6.16}$$

where a few differences between this update stage and the one from the previous Section 6.1 are noticed. To begin with, the system is completely linear, with the output being calculated simply by multiplying the matrix $\boldsymbol{C}$ with the state vector $\boldsymbol{x}$. The probability matrix $\boldsymbol{P}_{EKF}$ is received from time to time along with the EKF position estimation $\boldsymbol{p}_{EKF}$, to give a certainty of the value that is being calculated in the EKF implemented on the tracker ASVs.

6.2.1 Simulation Results

The CKF implemented on each AUV target is represented by the block diagram in Figure 6.5. The target EKF estimate is broadcasted by an ASV tracker, and received from time to time by the AUV targets.

The simulation results for the CKF are presented in Figure 6.6. As observed in Section 6.1.1, when the EKF Error plot shows values of distances near zero, the estimate is considered to be coinciding with the actual target's position. In light of this, by analysing the CKF Error plot, one can observe how close the target's CKF position prediction is to its actual position.

The variances for the CKF are the defined as

$$\boldsymbol{W} = 10^{-6} \cdot \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} , \boldsymbol{N} = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix} . \tag{6.17}$$

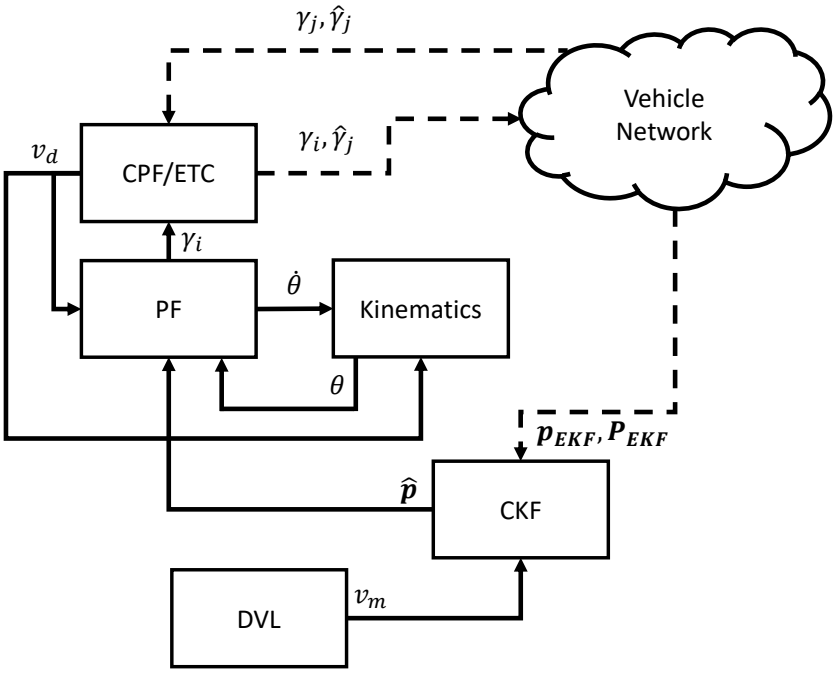


Figure 6.5: CKF included in the overall system represented in block diagram.

Similarly to Section 6.1.1, the probability matrix $\boldsymbol{P}_{k-1|k-1}$ must be initialised. By the same logic, the variance of the process noise and the DVL velocity measurement noise combined, just as stated in Equation 6.13, can be used to initialise the probability matrix $\boldsymbol{P}_{k-1|k-1}$.

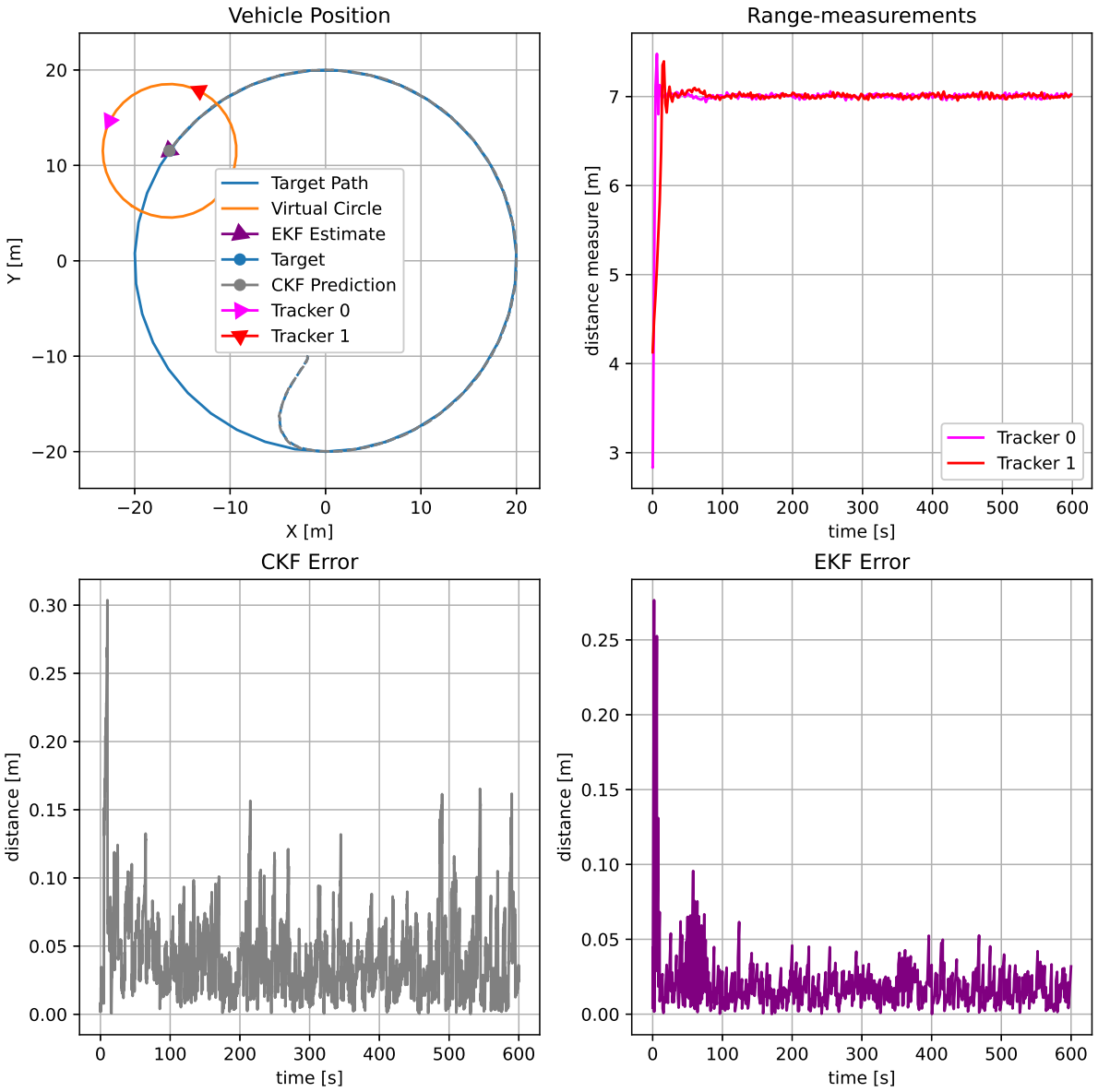


Figure 6.6: Plotting the simulation results of the CKF implemented in the system.

Chapter 7

Cooperative Multiple Formation Control

This section introduces an innovative method of distributed control mechanisms to achieve cooperation between different groups of vehicles, especially applied to the system introduced in Section 1.2. Groundbreaking work is reported in Teixeira [6], where a group of coordinated ASVs follow a moving circle, which in turn follows a target, such as a single or even a group of coordinated AUVs. The moving circle has a velocity control strategy to converge on top of the target. However, in the present thesis a different strategy is proposed, not only regarding faster vehicle convergence to their respective path, but also coordination and synchrony between an elected AUV from the underwater formation and a virtual circle. All things considered, this system guarantees convergence and coordination between all involved vehicles, since there are CPF mechanisms layered and implemented on each of the group of vehicles, so that coordination between vehicles is reached within their own group. The designed system can be observed in Figure 7.1, where a group of vehicles achieve coordination, and the different layers of cooperation are denoted.

It is assumed the virtual circle state and coordination control is implemented on one of the ASV trackers and shared between trackers via WiFi connection, thus being easily accessible to every ASV. The virtual circle moves always on top of its respective path, which might be shared with the elected target AUV or, for instance, the centre of mass of the target AUVs' group, in order to reach convergence between the two points in the 2D frame.

The Cooperative Multiple Formation Control (CMFC) consists on a new CPF layer on top of the previously implemented cooperative layers on both the ASV group and the AUV group, while also relying on ETC, since the communication established is between one of the ASV trackers and the elected AUV target, and therefore it is under the constraints of underwater acoustic communications. Thence and foremost, parallel to regular CPF, a new velocity correction term v_{cf} related to the cooperation between

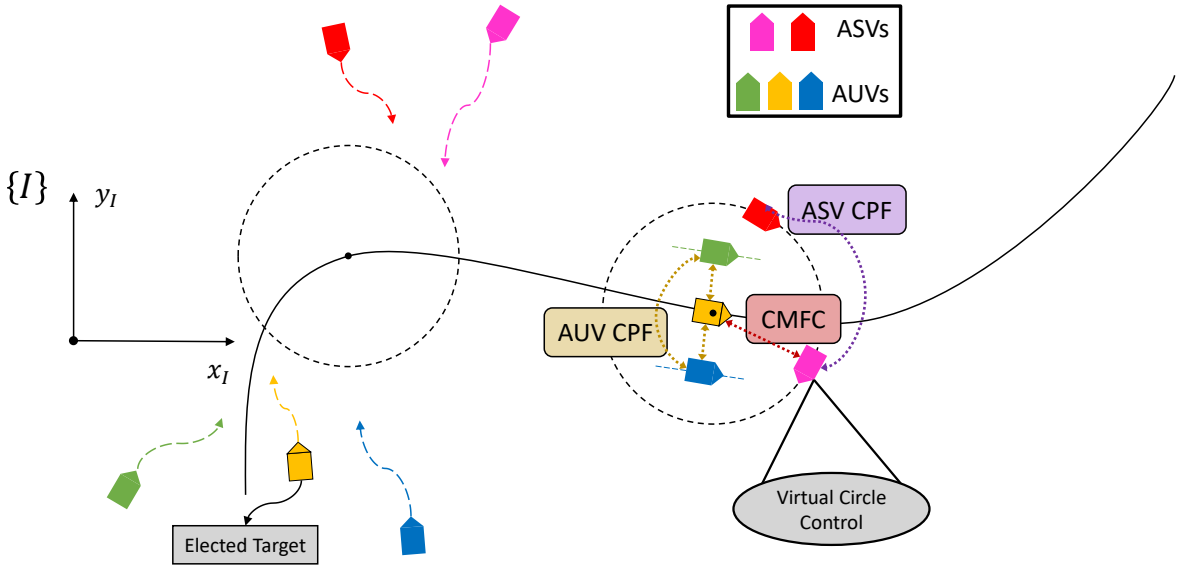


Figure 7.1: Illustration of the CMFC system.

formations arises, for a vehicle i, and is given as

$$v_{cf}^{[i]} = -k_{\xi f}^{[i]} \cdot \tanh\left(\gamma_i^{[i]} - \frac{1}{|\mathcal{N}_i|}\sum_{j\in\mathcal{N}_i}\hat{\gamma}_j^{[i]}\right), \tag{7.1}$$

where $k_{\xi f}^{[i]}$ is a tuning parameter for the convergence of coordination states, determining the minimum and maximum values of the velocity correction of the formations v_{cf} for the vehicle i, and the terms inside the hyperbolic tangent represent the difference between $\gamma_i^{[i]}$, the actual coordination state γ of vehicle i and the average of the estimates of the neighbour coordination states $\hat{\gamma}_j^{[i]}$.

Consider the average distance error of all ASV trackers to their corresponding virtual targets, defined by

$$e_f^{av} = \frac{1}{|\mathcal{N}_{ASV}|}\sum_{m\in\mathcal{N}_{ASV}} e_f^{[m]}, \tag{7.2}$$

where $|\mathcal{N}_{ASV}|$ is the total number of ASV trackers circulating the moving path, and $e_f^{[m]}$ is the distance of the ASV m relative to its virtual target, given by

$$e_f^{[m]} = ||\boldsymbol{p}^{[m]} - \boldsymbol{p}_s^{[m]}||, \tag{7.3}$$

where $\boldsymbol{p}^{[m]}$ is the vehicle's position, and $\boldsymbol{p}_s^{[m]}$ the respective virtual target position on the path, determined by the PF algorithm determined in Section 3.2.

The speed control law for the virtual circle accounts for a delay in the convergence of the ASVs to the circular path, i.e. it only corrects its velocity using the velocity correction term v_{cf}, until the trackers converge to the virtual circle. Once the trackers converge, the desired speed converges to the speed

profile v_p, and the circle begins to move according to

$$v_{\odot} = \underbrace{(1 - \tanh(k_f \cdot e_f^{av}))}_{\text{vehicle distance to moving path control}} \cdot v_p + v_{cf} \,, \tag{7.4}$$

where k_f is a sensitivity tuning gain for the average of the distances from the ASVs to their corresponding virtual targets e_f^{av}. The velocity correction for the formation coordination control v_{cf} is always active and follows the classic CPF logic, similarly to Equation 7.1. Once the virtual circle is coordinated with the elected AUV target, the velocity correction v_{cf} converges to zero.

The velocity law for the elected AUV can be simply formulated as

$$v_{target} = v_p + v_c + v_{cf} \,, \tag{7.5}$$

where the velocity correction v_c is relative to the regular CPF layer of the group of AUVs, and the velocity correction v_{cf} refers to the coordination layer between formations, the CMFC in a nutshell. Once again, when coordination is reached between all the elements in the system, the elected target's velocity v_{target} converges to the assigned velocity profile v_p. The other target AUVs do not suffer any modifications to their velocity laws, since their objective is to achieve coordination solely with the elected AUV target. Global coordination is then achieved as the elected AUV target serves as the bridge between the surface and underwater formations.

To preserve tracker movement before the alignment of both ASV and AUV formations, another velocity law is formulated. The trackers must only move when the virtual circle and elected target are coordinated. To this end, the following error between coordination states is determined

$$\gamma_{error} = \gamma_{\odot} - \hat{\gamma}_{target} \,, \tag{7.6}$$

where $\gamma_{\odot}$ is the coordination state of the virtual circle, and $\hat{\gamma}_{target}$ is the estimated coordination state of the elected AUV target, since the trackers do not have access to the elected target's state at all times. The tracker's velocity law is then given by

$$v_{tracker} = \underbrace{(1 - \tanh(k_g \cdot |\gamma_{error}|))}_{\text{formation discoordination control}} \cdot v_p + v_c \,, \tag{7.7}$$

where k_g is the sensitivity tuning parameter for the coordination state error γ_{error}, and v_c is the usual velocity correction of the CPF layer between the ASVs. The formation discoordination control is the term which gives the ASVs the ability to wait for the vehicle formations to align before performing the encircling MPF manoeuvre.

All these velocity laws have the useful property of being bounded, since all the velocity correction terms are bounded and the implemented control terms, such as the vehicle distance to moving path control and the formation dicoordination control, are within the range of $\varepsilon \in [0, v_p]$.

7.1 Smart Cooperative Path Following

For this section, an approach was devised to provide a more streamlined convergence to the path and then convergence of coordination states. The idea is for the vehicles to perform the PF control without the influence of the CPF coordination state convergence, only including the latter when the vehicles are close enough to the path. The fact that this newly designed strategy focuses on reaching the path with PF techniques before performing the actual CPF does even more justice to the assumption made on the normalisation of the virtual target s to determine the coordination state γ of the vehicle. The desired velocity law v_d for a given vehicle i is then defined as

$$v_d^{[i]} = v_p^{[i]} + \underbrace{(1 - \tanh(k_s^{[i]} \cdot e_f^{[i]}))}_{\text{convergence to path control}} \cdot v_c^{[i]}\,, \tag{7.8}$$

where $v_p^{[i]}$ is the ordinarily assigned speed profile, $v_c^{[i]}$ is the velocity correction associated with the CPF performance, and $k_s^{[i]}$ is the sensitivity tuning parameter for the distance of the vehicle i to its virtual target on the path $e_f^{[i]}$, according to Equation 7.3. Taking all this into consideration, Equation 7.8 hence serves as an alternative to the Equation 4.5, previously determined by the regular CPF in Section 4.1.2.

The stability of the law in Equation 7.8 can be intuitively inferred, since the vehicle performs a PF manoeuvre before executing the CPF coordination technique. In Lapierre et al. [7], stability proof is presented for the PF algorithm employed. In Hung et al. [27], the CPF with ETC mechanism is demonstrated, along with proof of convergence. It is hence within reason to assume there is stability when applying the desired velocity law in Equation 7.8.

The velocity laws determined previously for the CMFC had to be slightly modified to take into account this new feature. The elected target AUV is then governed by

$$v_{target} = v_p + \underbrace{(1 - \tanh(k_s \cdot e_f^{[target]}))}_{\text{convergence to path control}} \cdot (v_c + v_{cf})\,, \tag{7.9}$$

where the convergence to path control, also named as Smart Cooperative Path Following (SCPF), is multiplied by both the CPF between AUV targets velocity correction v_c, and the CMFC velocity correction v_{cf}. Otherwise, the Equation 7.9 does not differ much from the original Equation 7.5. Seeing that this new feature affects every vehicle velocity law, the ASV trackers must consider it as well. The tracker law is now expressed as

$$v_{tracker} = (\underbrace{1 - \tanh(k_g \cdot |\gamma_{error}|)}_{\text{formation discoordination control}} + \underbrace{\tanh(k_s \cdot e_f^{[tracker]}))}_{\text{compensation for SCPF}} \cdot v_p + \underbrace{(1 - \tanh(k_s \cdot e_f^{[tracker]}))}_{\text{SCPF}} \cdot v_c\,, \tag{7.10}$$

where a compensation term for the SCPF is implemented, so that the tracker moves even if it finds itself far from the virtual circle, since now there is no velocity correction while the ASV does not reach the moving path. In this sense, Equation 7.10 is shown as an alternative to the previously denoted Equation 7.7.

7.2 Simulation Results

In the following sections, the algorithms previously explained are tested in simulations, resulting in the final system proposed in this thesis.

7.2.1 Cooperative Multiple Formation Algorithm

The system presented in Figure 7.2 shows the essential mechanism of CMFC and how it blends in with the rest of the system. The CMFC block is implemented on an elected AUV target and the ASV trackers, so that each ASV has access to the velocity law designed in Equation 7.10. However, only an elected ASV tracker runs the velocity law for the circle defined in Equation 7.4. The state of the virtual circle $\gamma_{\odot}$ and the error γ_{error} expressed in Equation 7.6 is constantly shared among the group of ASVs.

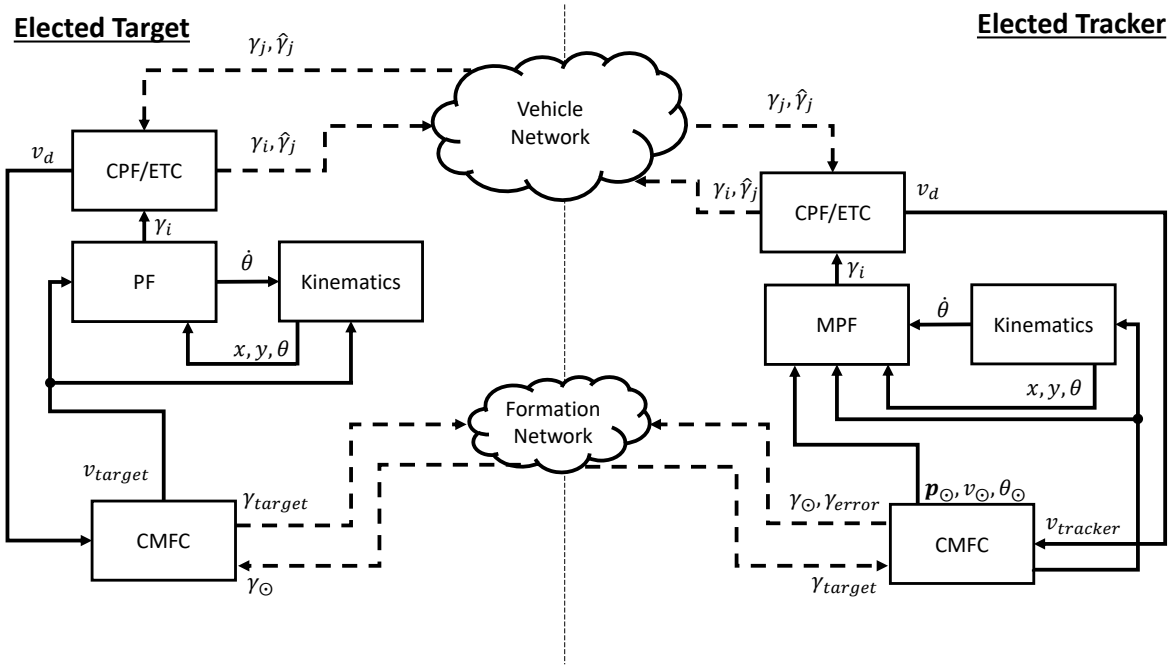


Figure 7.2: CMFC mechanism implemented on the overall system, represented in a block diagram.

The simulation for the CMFC algorithm is shown in Figure 7.3 where a coordination within formations is visible in the Vehicle Path Progression plot, as the lines for the AUV targets converge, as so do the lines for the ASV trackers. The Cooperative Formation Control plot also shows coordination between the two distinct formations, the group of AUVs and the group of ASVs, where the two coordination state lines of the virtual circle (projected within the **Tracker 0**) and the elected AUV target (**Target 1**) converge. The interruption of communications between vehicles, as seen in the ETC Broadcasting plot and the Cooperative Formation Control plot, also indicates achieved coordination between vehicles within the same formation, and also between the two different formations. The Vehicle Velocity plot also shows a constant final velocity for each vehicle, including the virtual circle, giving even more evidence of achieved vehicle coordination. In the Vehicle Position plot, one can observe the motion of each vehicle, adding

altogether to the coherence of the achieved coordination between vehicles.

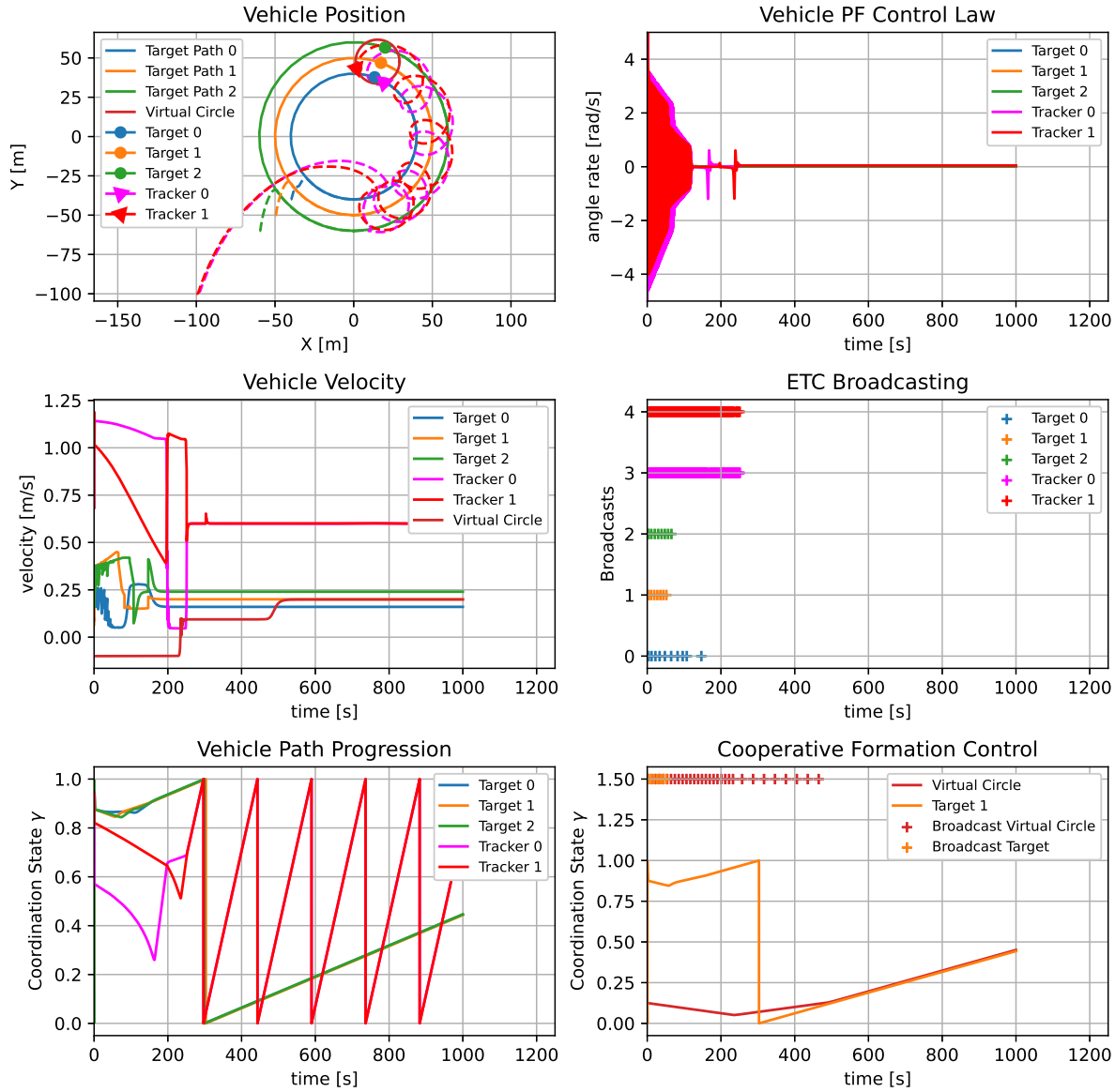


Figure 7.3: Plotting of the simulation for the CMFC system.

7.2.2 Filter Application

The final system is translated in a simulation composed of a formation of three AUV targets $\mathcal{F}_{AUV}$, with a local CPF strategy within the formation, which in turn cooperates with a similar ASV tracker formation $\mathcal{F}_{ASV}$ of two vehicles, since a local CPF strategy is also applied. The ASVs perform an encircling manoeuvre around an elected AUV target from $\mathcal{F}_{AUV}$ using MPF techniques. Moreover, a filter application like the ones seen in 6.1 and 6.2 is implemented on the system. The EKFs are implemented on the elected ASV tracker that performs range measurements to each AUV target in $\mathcal{F}_{AUV}$, and receives similar range measurements performed by the other ASV tracker. Each EKF estimate of the position of the AUV targets is then broadcasted to each of the vehicles in $\mathcal{F}_{AUV}$ so that a mounted CKF can blend that

information with its estimate of its own position, helping to the underwater localisation of the AUVs. Considering all this, the acoustic channel between $\mathcal{F}_{ASV}$ and $\mathcal{F}_{AUV}$ encompass the range measurements, the transmission of the EKF estimate, and the coordination state of both the virtual circle from the CMFC and the elected target AUV. The resulting system is represented in Figure 7.4, showing in a simplified manner how the CMFC works in tandem with the concepts explained in the previous sections.

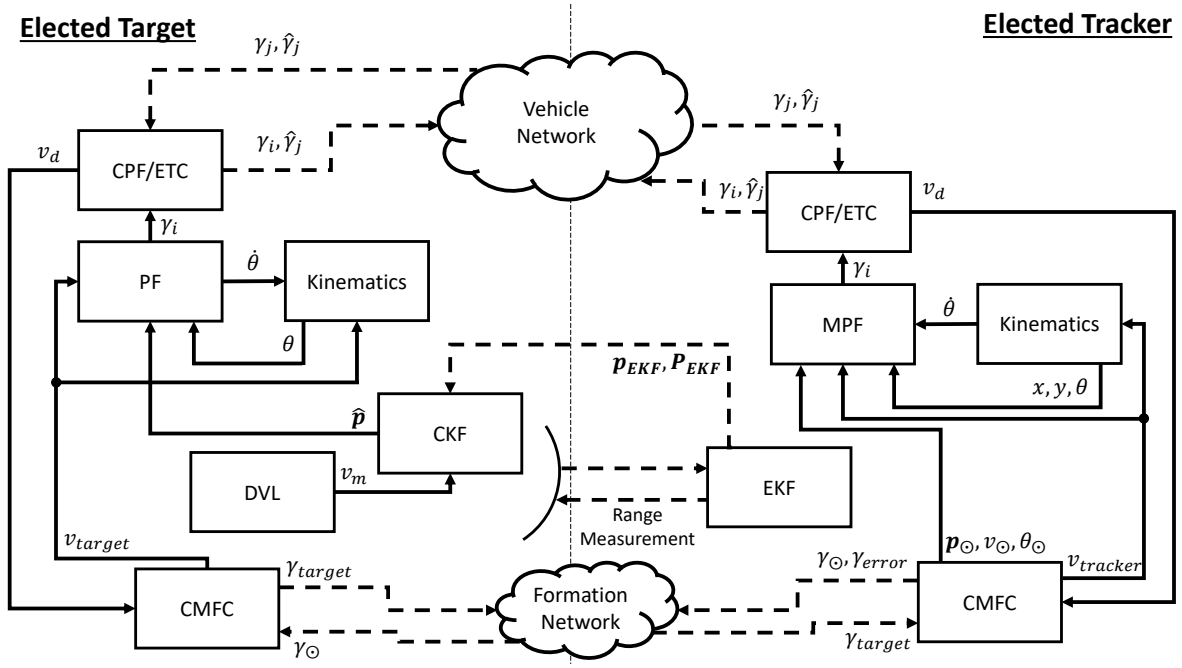


Figure 7.4: CMFC mechanism implemented on the overall system, represented in a block diagram.

The transmission of the EKF estimates starts only after the error between the virtual circle and the elected AUV target is not greater than a certain maximum error threshold t_h. This mechanism shows better performance, since the EKF estimates have time to be better estimated as the vehicles converge to their respective paths. The threshold, for this simulation, was considered as $t_h = 0.05$.

The values for the parameters used during the simulation are displayed on Tables 7.1, 7.2, 7.3, 7.4 and in Equations (7.11) and (7.12).

Table 7.1: Path Following values.

γ	k_1	k_2	k_{delta}	θ_a
1	1	0.3	0.5	0.8

For the EKF, the following variances were used

$$\boldsymbol{Q} = 10^{-6} \cdot \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \boldsymbol{R} = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}. \tag{7.11}$$

Table 7.2: Cooperative Path Following values.

AUV					
k_ξ^0	v_p^0	k_ξ^1	v_p^1	k_ξ^2	v_p^2
0.12	0.16	0.15	0.2	0.18	0.24

ASV			
k_ξ^0	v_p^0	k_ξ^1	v_p^1
0.4	0.5	0.4	0.5

Table 7.3: Event-Triggered Communications values.

c_0	c_1	c_2
0.01	0.5	1

For the CKF, the following variances were used

$$\boldsymbol{W} = 10^{-6} \cdot \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \boldsymbol{N} = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}. \tag{7.12}$$

Table 7.4: Cooperative Multiple Formation Control and Smart Cooperative Path Following values.

$k_{\xi f}^{target}$	$k_{\xi f}^{\odot}$	k_f	k_g	k_s
0.1	0.1	1	0.1	0.01

The simulation shows the designed CMFC technique in action, as well as both filter applications. The Cooperative Formation Control plot presents the convergence between the elected AUV target coordination state γ_{target} and the coordination state $\gamma_{\odot}$ of the virtual circle projected within one the ASV trackers, as the two lines converge. The communications derived from the ETC mechanism are also visible, indicating that the two different formations $\mathcal{F}_{ASV}$ and $\mathcal{F}_{AUV}$ achieve coordination between each other. The Vehicle Position plot not only suggests that, but also demonstrates the desired behaviour of the designed filters. The EKFs implemented on the elected ASV tracker estimate the position and velocity of the AUV targets, showing a desired performance on the EKF Error plot, where the distance between the estimates and the real positions of the AUV targets is minimal. This serves as a great launching point for the CKFs implemented on each AUV target, which performances are represented in the CKF Error plot. As observed, the distance between the CKFs' prediction for each target's position and its actual position is also quite small. Thus, the velocities of all vehicles, including the virtual circle, converge to the velocity profile assigned to each path.

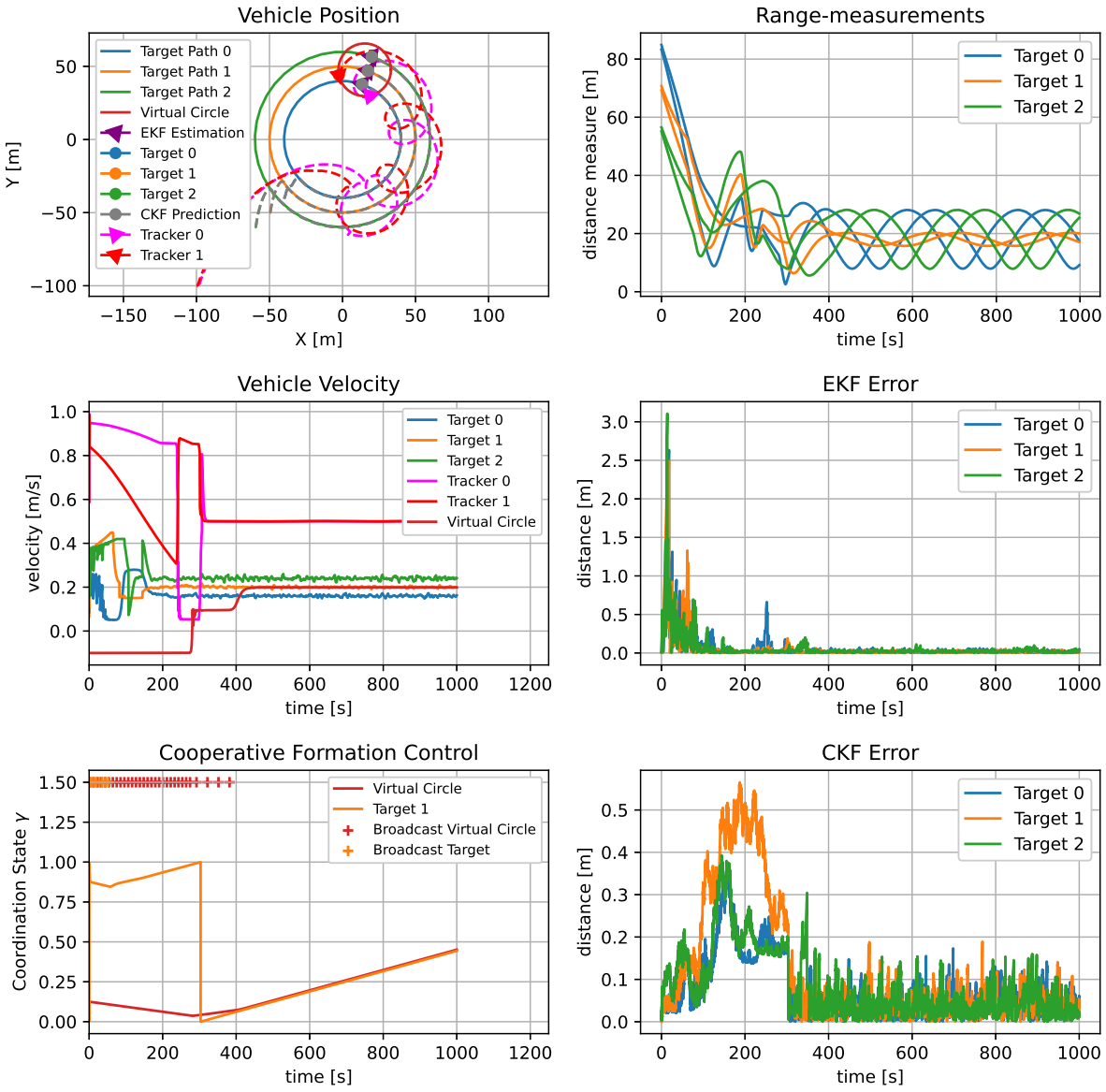


Figure 7.5: Plotting of the simulation for the CMFC system, with filter application.

Chapter 8

Conclusions

In conclusion, a Path Following (PF) centred approach was taken in light of the essence of the problem and objectives defined in Section 1.2. The envisioned system was meant to increase the resolution of seismic surveys with a novel autonomous method of vehicle cooperation and control. With the cooperation between a group of Autonomous Underwater Vehicles (AUVs) and a group of Autonomous Surface Vehicles (ASVs), the cells dividing the ocean floor are illuminated with acoustic signals from different angles. This last described aspect is the reason for the predicted increased resolution on these types of surveys.

The building block of Path Following was crucial throughout the development of this thesis, therefore it was essential to choose the right algorithm to fit the global scope of the project. The method described in Lapierre et al. [7] presented global convergence and a virtual target which could be used as an indicator of the vehicle's progression along a certain path. This came as a great convenience when introducing the next step of vehicle cooperation, paving the way to Cooperative Path Following (CPF).

The cooperation between vehicles led to a connected network where each vehicle communicates with its neighbours to achieve coordination within the whole vehicle group. This is where the virtual target from the PF technique was fundamental, as it directly indicates the coordination state of each vehicle. The fact that CPF is still heavily based on the PF led to a time-independent solution. Along with discrete Event-Triggered Communications (ETC), the group of vehicles is able to reach coordination, thus effectively cooperating amongst them while following a certain path. These features can even take into account temporary vehicle malfunctions and delays in the communication network.

The next contemplated concept was Moving Path Following (MPF), in order to obtain the aforementioned different angles of incidence of the acoustic signals as they reach the ocean floor. In this setup, the ASVs would serve as acoustic sources encircling the AUVs, while these vehicles tow acoustic antennas to pick up the reflected signal. The subsequent MPF law takes into consideration the velocity of the moving path and computes a local PF law, resulting in a desired velocity and angle for the ASVs. This technique guarantees the PF of a virtual moving circular path by the ASV trackers, translating in a circling manoeuvre around the target group of AUVs. The centre of the circular path is laid on top of one of the target or even a whole group AUVs. However, it is assumed the ASV trackers know the position

of the AUVs at all times.

To overcome this naive assumption, an Extended Kalman Filter (EKF) was implemented on an ASV tracker. This filter allows for the estimation of the target's position, thus making the simulation more realistic. Taking it a step further, to atone for underwater localisation, a Complementary Kalman Filter (CKF) was implemented on the AUVs, receiving the EKF estimate from the ASV trackers and blending that information with the Doppler Velocity Log (DVL) sensor installed onboard.

The final step was to establish cooperation between two different groups of vehicles: the ASV trackers and the AUV targets, i.e. Cooperative Multiple Formation Control (CMFC). To this end, a new layer of cooperation was conceived between an elected AUV target from the group of AUVs and one of the ASV trackers, which in turn easily shares over WiFi the information about the underwater formation. This novel system considers the progression of each formation of vehicles, controlling the velocity of the elected AUV target and the virtual circle to where the ASV trackers converge. Furthermore, the virtual moving circle waits for the trackers to arrive at their path, and each tracker only starts the encircling manoeuvre when the elected AUV target and the centre of the virtual circle are aligned. To help the vehicles converge to their respective paths, a Smart Cooperative Path Following (SCPF) was designed, letting the ASVs and AUVs reach their own path with an assigned profile speed before starting to cooperate between each other, thus correcting their velocity.

In the overall view, the system described in Section 1.2 was achieved, with promising simulations using the kinematic model of a vehicle, throughout each section of the this thesis. A new control method for multiple formations arose with improved performance relative to other algorithms proposed in the literature. Coordination between vehicles and formations, with filter applications were employed, and a comprehensive and compact review of the used strategies was accomplished. The dashboard of plots regarding each simulation allowed for a sharp, yet broad understanding of each trial and its relation to the theory.

8.1 Future Work

The intuitive next improvement for this work would be to develop a real world implementation of the accomplished system. All the work developed in the scope of this document is simulated and it only takes into consideration the kinematics of a vehicle. Although the dynamic model of the vehicles were outside of the scope of this thesis, it could serve as a middle point between the developed kinematics-system and the real world vehicle control. An important experiment to be carried out would be the evaluation of this system before the seismic surveying itself, gathering a real batch of seismic data.

Another path which could be explored is the case for multiple AUV target formations. The explanation from Chapter 7 focuses mainly on the simple case of two vehicle formations, a group of AUV targets and a group of ASV trackers. The realisation of a system with a greater number of groups of vehicles involved in the formation control allows for different architectures and approaches to seismic surveying missions.

Yet another interesting feature to implement would be to get a threshold for the distance between the vehicle and its virtual target, on the SCPF. If this distance is greater than the threshold, then the

ETC and CPF mechanism would be turned off, focusing again on the PF. When the distance crosses the threshold and remains lower than the threshold, the CPF and ETC would start working to achieve coordination.

Regarding the EKF, there can be some cases where the estimation may diverge, due to sensor malfunctions, communication losses, among other reasons, and a circular path encircling manoeuvre around a fixed point in space, which could be an educated guess on where the vehicle might be or might have been, may facilitate the re-estimation of the AUV target.

Finally, further proof of stability and convergence for the laws governing the system developed in this thesis is the next step in strengthening the theory behind this work.

These are some demonstrative examples of the work that could be conducted to further develop the issues exposed in this thesis. The future of seismic surveying is deeply connected with autonomous and cooperative systems, where such flexibility and ease of mission and vehicle deployment provide an unfair advantage to conventional methods, such as the large vessels equipped with sparkers, used to tow massive nets of cable and hydrophones, or acoustic antennas.

Bibliography

[1] E. Zereik, M. Bibuli, N. Mišković, P. Ridao, and A. Pascoal. Challenges and future trends in marine robotics. *Annual Reviews in Control*, 46:350–368, 2018. ISSN 1367-5788. doi: https://doi.org/10.1016/j.arcontrol.2018.10.002. URL https://www.sciencedirect.com/science/article/pii/S1367578818300038.

[2] G. Indiveri. Navigation, Guidance and Control of Underwater Vehicles within the Widely scalable Mobile Underwater Sonar Technology (WiMUST) Project: an Overview. ISME, and IST, and INTAL, and Univ. Hertfordshire, and GEO, and CGG, GRAALtech, and EvoLogics, and GeoSurveys. IFAC NGCUV 2015 Workshop, April 2015.

[3] J. Kalwa, A. Pascoal, P. Ridao, A. Birk, T. Glotzbach, L. Brignone, M. Bibuli, J. Alves, and M. Silva. EU project MORPH: Current Status After 3 Years of Cooperation Under and Above Water. *IFAC-PapersOnLine*, 48(2):119–124, 2015. ISSN 2405-8963. doi: https://doi.org/10.1016/j.ifacol.2015.06.019. URL https://www.sciencedirect.com/science/article/pii/S240589631500258X. 4th IFAC Workshop onNavigation, Guidance and Controlof Underwater VehiclesNGCUV 2015.

[4] H. Al-Khatib, G. Antonelli, A. Caffaz, A. Caiti, G. Casalino, I. B. de Jong, H. Duarte, G. Indiveri, S. Jesus, K. Kebkal, A. Pascoal, and D. Polani. The Widely scalable Mobile Underwater Sonar Technology (WiMUST) Project: An Overview. In *OCEANS 2015 - Genova*, pages 1–5, 2015. doi: 10.1109/OCEANS-Genova.2015.7271688.

[5] G. Indiveri. Widely scalable Mobile Underwater Sonar Technology WiMUST Proposal. 2014.

[6] D. Teixeira. Sensor-Based Cooperative Control of Multiple Autonomous Marine Vehicles. Master's thesis, Instituto Superior Técnico, Lisbon, October 2019. MSc in Electrical and Computer Engineering.

[7] L. Lapierre, D. Soetanto, and A. Pascoal. Nonsingular path following control of a unicycle in the presence of parametric modelling uncertainties. *INTERNATIONAL JOURNAL OF ROBUST AND NONLINEAR CONTROL*, pages 485–503, April 2006.

[8] R. Ghabcheloo, A. P. Aguiar, A. Pascoal, C. Silvestre, I. Kaminer, and J. Hespanha. Coordinated path-following in the presence of communication losses and time delays. *SIAM - Journal on Control and Optimization, Vol. 48, No. 1*, pages 234–265, 2009.

[9] F. V. Vanni. Coordinated Motion Control of Multiple Autonomous Underwater Vehicles. Master's thesis, Instituto Superior Técnico, July 2007. MSc in Aerospace Engineering.

[10] R. Rego, N. Hung, and A. Pascoal. Cooperative Motion Control Using Hybrid Acoustic-Optical Communication Networks. *IFAC-PapersOnLine*, 54(16):232–237, 2021. ISSN 2405-8963. doi: https://doi.org/10.1016/j.ifacol.2021.10.098. URL https://www.sciencedirect.com/science/article/pii/S2405896321015007. 13th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles CAMS 2021.

[11] N. T. Hung, F. C. Rego, and A. M. Pascoal. Event-Triggered Communications for the Synchronization of Nonlinear Multi Agent Systems on Weight-Balanced Digraphs. In *2019 18th European Control Conference (ECC)*, pages 2713–2718, 2019. doi: 10.23919/ECC.2019.8796277.

[12] T. Oliveira, A. P. Aguiar, and P. Encarnação. Moving Path Following for Unmanned Aerial Vehicles with Applications to Single and Multiple Target Tracking Problems. *IEEE Transactions on Robotics, Vol. 32, Issue 5*, October 2016.

[13] N. T. Hung, F. F. C. Rego, and A. M. Pascoal. Cooperative Distributed Estimation and Control of Multiple Autonomous Vehicles for Range-Based Underwater Target Localization and Pursuit. *IEEE Transactions on Control Systems Technology*, pages 1–15, 2021. doi: 10.1109/TCST.2021.3107346.

[14] A. Pascoal, I. Kaminer, and P. Oliveira. Navigation system design using time-varying complementary filters. *IEEE Transactions on Aerospace and Electronic Systems*, 36(4):1099–1114, 2000. doi: 10.1109/7.892661.

[15] M. Ribeiro and I. Ribeiro. Kalman and Extended Kalman Filters: Concept, Derivation and Properties. 04 2004.

[16] P. Abreu, H. Morishita, A. Pascoal, J. Ribeiro, and H. Silva. Marine vehicles with streamers for Geotechnical surveys: Modeling, positioning, and Control. LARSyS/ISR/IST (PT) and Dept. Ocean Engineering, Univ. S. Paulo (BR). CAMS2016 Conference, Trondheim., September 2016.

[17] J. Ribeiro, A. P. Aguiar, and A. Pascoal. Motion Control of Single and Multiple Autonomous Marine Vehicles. *Instituto Superior Técnico*, page 10, November 2011. Lisbon, Portugal.

[18] T. I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons Ltd., first edition edition, 2011.

[19] J. Ribeiro. Motion Control of Single and Multiple Autonomous Marine Vehicles. Master's thesis, Instituto Superior Técnico, October 2011. MSc in Electrical and Computer Engineering.

[20] G. Sanches. Sensor-Based Formation Control of Autonomous Marine Robots. Master's thesis, Instituto Superior Técnico, December 2015. MSc in Electrical and Computer Engineering.

[21] F. Branco. Github Thesis Repository: Cooperative Marine Vehicle Navigation and Control with Applications to Geotechnical Surveying. https://github.com/franciscobranco/Thesis.

[22] A. Micaelli and C. Samson. Trajectory tracking for unicycle-type and two-steering-wheels mobile robots. *Programme 4 - Robotique, image et vision, Projet Icare, Rapport de recherche nº2097*, page 36, November 1993. Sophia Antipolis, France.

[23] P. Maurya, A. Aguiar, and A. Pascoal. Marine Vehicle Path Following Using Inner-Outer Loop Control. *Proc. MCMC'09 - 8th Conference on Manoeuvring and Control of Marine Craft*, September 2009. Guarujá (SP), Brazil.

[24] J. M. Lemos. *Controlo no Espaço de Estados*. IST Press, first edition edition, 2019.

[25] H. Liu. Barbalat's Lemma, Nomvember 2009.

[26] K. Kebkal, O. Kebkal, V. Kebkal, A. Pascoal, J. Ribeiro, M. Rufino, L. Sebastião, G. Indiveri, L. Pollini, and E. Simetti. Hydro-Acoustic Communications and Networking in Contemporary Underwater Robotics: Instruments and Case Studies. *Modelling and Control of Autonomous Vehicles, IET, London, UK, Editors Sanjay Sharma and Bidyadhar Subushi*, 2018.

[27] N. Hung, A. Pascoal, and T. 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*, pages 1–27, 2020.