



Doctoral Thesis in Electrical Engineering

Optimizing Ocean Feature Estimation and Tracking through Adaptive Sampling and Formation Control of Autonomous Underwater Vehicles

JOANA FILIPA GOUVEIA FONSECA

Optimizing Ocean Feature Estimation and Tracking through Adaptive Sampling and Formation Control of Autonomous Underwater Vehicles

JOANA FILIPA GOUVEIA FONSECA

Academic Dissertation which, with due permission of the KTH Royal Institute of Technology, is submitted for public defence for the Degree of Doctor of Philosophy on Thursday the 28th of September 2023, at 10:00 a.m. in Kollegiesalen, Brinellvägen 6, KTH Campus, Stockholm.

Doctoral Thesis in Electrical Engineering
KTH Royal Institute of Technology
Stockholm, Sweden 2023

© Joana Filipa Gouveia Fonseca

TRITA-EECS-AVL-2023:57

ISBN 978-91-8040-674-1

Printed by: Universitetservice US-AB, Sweden 2023

Abstract

Increased ocean temperatures caused by climate change are expected to lead to more frequent and severe harmful algal blooms, which deteriorate water quality, cause human illness and fish mortality. Scientific understanding of algal blooms and their dynamics is limited due to the lack of data from such ocean phenomena. State-of-the-art ocean monitoring includes satellite imagery and dedicated research vessels. Mobile sensors based on autonomous underwater vehicles (AUVs) and other robotic technologies are of growing importance for efficient environmental monitoring of the oceans. The overall objective of this thesis is to design a system for ocean feature estimation and tracking based on adaptive sensor sampling using AUVs. The thesis contributions are focused on the following three problems.

The first problem we consider is how to estimate and track circular and non-circular ocean features using a multi-robot system. We propose a circumnavigation control law, proving that it forces the AUVs to converge to a circular formation. Two target estimation algorithms are presented: one is based on a leader-follower approach, while the other is distributed. Both algorithms are shown to successfully estimate and track the mobile target's location.

Secondly, we consider the problem of tracking ocean fronts using a single AUV supported by satellite data. We develop a Gaussian process model for the front estimation and show how it can be updated based on the available sensor and satellite data. Using this model, a control law is developed that guides the AUV to move toward and along the ocean front. The closed-loop system is evaluated through a detailed simulation environment with realistic vehicle and environment models and real algal bloom data.

Finally, we develop an experimental setup based on a real AUV to demonstrate that our method for algal bloom tracking is feasible in practice. We show experimental results from two surveys in the Stockholm archipelago and compare the performance of the real system with simulation studies. The results indicate that the front tracking and gradient estimation algorithms are working well but also suggest important items for further studies.

Sammanfattning

Stigande havstemperaturer på grund av klimatförändringar förväntas leda till fler och allvarigare algblomningar som kommer att försämra vattenkvaliten, men också orsaka sjukdomar bland människor och dödsfall bland fiskar. Samtidigt är den vetenskapliga förståelsen för algblomning och dess utbredningsmekanismer begränsad, och det finns ont om information om detta fenomen. I dagsläget görs havsövervakning med bland annat satellitbilder och dedikerade forskningsfarkoster. Mobila sensorer på autonoma undervattensfarkoster (autonomous underwater vehicles, AUVs) och andra robotikmetoder har därmed börjat spela en större roll i att uppnå en effektiv klimatövervakning i våra vatten. Det övergripande målet med denna avhandling är att designa ett system för detektering och spårning av havsföremål med hjälp av adaptiv sensorprovtagning med AUVs. Avhandlingens bidrag fokuserar på de följande tre problemen.

Det första problemet som behandlas är hur cirkulära och icke-cirkulära havsföremål kan detekteras och spåras av ett system med flera robotar. Vi föreslår en regleralgoritm som bygger på kringsegling, och bevisar hur det resulterar i att alla AUVs konvergerar till en cirkulär formation. Två målestimeringsalgoritmer presenteras: den ena bygger på en ledare-följare-metod och den andra på en distribuerad metod. Vi demonstrerar att båda algoritmerna kan detektera det rörliga målet och spåra dess position.

Det andra problemet som behandlas är spårning av havsfronter med en ensam AUV som har tillgång till satellitinformation. Vi använder en gaussisk process för att modellera fronter och visar att den kan uppdateras med data från sensorer och satelliter. Modellen används följaktligen till att ta fram en regleralgoritm som styr en AUV till fronten och sedan följer den. Det återkopplade systemet utvärderas i simuleringar med realistiska modeller för farkost och vattenmiljö tillsammans med verklig algblomningsdata.

Slutligen utvecklar vi en experimentplattform baserat på en riktig AUV för att demonstera att spårningen av algblomning fungerar i verkligheten. Resultat från två experiment in Stockholms skärgård presenteras, och det verkliga systemets prestanda jämförs med prestandan i simulering. Resultaten indikerar att algoritmerna för frontspårning och gradientestimering fungerar väl, men lyfter även fram frågor som bör besvaras i framtida studier.

Acknowledgments

I'm very thankful to have Professor Karl Henrik Johansson as my supervisor. Thank you for the opportunity to work in this lively team and on such a relevant and exciting research topic. I admire your ability to create a strong, safe, cooperative working environment that allows us to thrive! Thank you for your expertise, your kindness, support, and understanding. I always leave our meetings less anxious than when I get in. Also, thank you for allowing me the freedom to do and attend most of the things I wanted and for the trust in giving me a flexible research topic, schedule, and time frame. If I can, I would also thank the country, Sweden, for receiving me happily and making most of my life here rather easy and comfortable. I would also like to acknowledge the Swedish Research Council Distinguished Professor Grant 2017-01078, and Knut and Alice Wallenberg Foundation Wallenberg Scholar Grant, which partially supported this work.

Thanks also to my co-supervisor Jonas Mårtensson. It was great to work with you on the Nonlinear Control course. I learned a lot from it, and it's encouraging to see all the commitment and energy you put into creating it. I'm also very lucky to have worked with great people such as Tor Arne Johansen from NTNU, Joao Sousa from U.Porto, Jieqiang Wei from Ericsson, Miguel Aguiar from KTH, and Ziwen Yang from SJTU. I also want to express my gratitude to everyone at SMaRC who received my ideas with open arms and worked alongside me to make my goals possible. First, thank you, Ivan Stenius, for allowing me in, being excited to work together, and trusting me to teach a part of your Underwater Technology course. Thanks also to everyone in your group, including Sriharsha, Josefine, Carl, Aldo, and more. Working with you on the demonstrations, presentations, simulations, and field surveys was very exciting.

Regarding this thesis, I would like to thank the remaining people who became part of it. Thank you to Nina Mahmoudian for accepting to be the defense opponent, Maarja Kruusmaa, Baris Fidan, Jana Tumova for participating on the jury committee, and Petter Ögren for being the advanced reviewer and substitute committee. Thank you so much to everyone who helped me proofread this thesis: Robert, Miguel, Pedro, Braghadeesh, Vandana, Rijad, Ines, Dzenan, Frank, Rebecka, and Ehsan. And thanks, Adam, for helping me translate the abstract into Swedish.

I want to particularly acknowledge my office mates for making it the coolest office. Mina, Rodrigo, Yuchao (and Lili), Fei, Stefan, Vahan, Hanxiao, Nana, Braghadeesh. But also, I must extend my gratefulness to this division as a whole,

every one of you, past and present, for making my workplace one I look forward to going to every day. Some of you are Matin, Matias, Vandana, Ines, Robert, Frank, Rijad, Mayank, Rebecka, Dzenan, Elis, Erik, Adam, Jacob, Ting, and more. I give heartfelt thanks to my Master thesis supervisor, Maria Rosario Pinho, for introducing me to academia; to Zhuhuan, for making my Stockholm house feel like home and for all our kitchen conversations; and to Pedro for his partnership and kindness.

Many thanks to my Porto family. Thanks, Dad, for the constant love and support, and thank you, Ana and Marta, for inspiring me to be my best self. Finally, a thank you note, that unfortunately can't be delivered, to my late grandmothers Benilde and Candida for all the love and hard work.

Joana

To my grandmothers Benilde and Candida

Contents

Abstract	ii
Sammanfattning	iv
Acknowledgements	vi
1 Introduction	1
1.1 Motivation	2
1.2 Research questions	8
1.3 Thesis Outline and Contributions	10
2 Background	17
2.1 Marine Sampling Platforms	18
2.2 Multi-agent Control for Marine Sampling	21
2.3 Single-agent Control for Marine Sampling	25
3 Leader-based Adaptive Target Estimation for Multi-agent Systems	33
3.1 Problem Formulation	33
3.2 Adaptive Estimation and Control	37
3.3 Convergence Results	41
3.4 Simulation Results	44
3.5 Summary	50
4 Decentralized Target Estimation for Multi-agent Systems	51
4.1 Problem Formulation	52
4.2 Optimal Circle Estimation and Control Algorithms	56
4.3 Convergence Results	58
4.4 Simulation Results	60

4.5	Extension for Non-circular Shapes	63
4.6	Simulation Results	67
4.7	Summary	72
5	Linear Front Approximation for Single-agent Systems	77
5.1	Problem Formulation	77
5.2	Front Detection Algorithm	80
5.3	Simulation Results	86
5.4	Summary	93
6	Least Squares Front Estimation for Single-agent Systems	95
6.1	Problem Formulation	96
6.2	Front Tracking Algorithm	99
6.3	Simulation Results	103
6.4	Summary	107
7	Gaussian Process Front Estimation for Single-agent Systems	109
7.1	Problem Formulation	110
7.2	Front Tracking Algorithm	110
7.3	Simulation Results	118
7.4	Sensitivity Analysis	125
7.5	Summary	126
8	Marine Experiments	127
8.1	Experimental Setup	128
8.2	Experimental Results in the Baltic Sea	136
8.3	Summary	138
9	Conclusions	143
9.1	Summary	143
9.2	Future work	146
	Appendices	149
A	Citation analysis	151
	Bibliography	156

Chapter 1

Introduction

Algal blooms are common in the oceans and have several negative effects, including lowering the water's oxygen content and creating toxins that can endanger both humans and marine animals. The frequency and severity of algal blooms are raising concerns as the global climate continues to change; there is an urgent need for more detailed and reliable information on algal blooms and other ocean phenomena. Algal blooms are predicted to occur more frequently and with greater intensity as ocean temperatures rise and weather patterns grow more harsh.

Autonomous monitoring methods have been investigated recently to help address the effects of these blooms. A potential strategy is using autonomous underwater vehicles (AUVs) to monitor Research problems and other ocean occurrences. AUVs can be sent into the ocean to gather data without human interaction. With sensors and tools for assessing water temperature, salinity, and dissolved oxygen, AUVs are suitable for detecting harmful algal blooms. Compared to more conventional techniques like ship-based surveys or buoy networks, using AUVs for data collection has several benefits. They can more swiftly and effectively cover broad portions of the ocean than other approaches, especially in isolated or dangerous locations. They can also cover difficult areas such as coastal regions, lakes, and rivers. Additionally, they offer high-resolution data that can be used to create high-resolution maps and models of ocean phenomena, which help scientists better comprehend how algal blooms develop and spread.

In this chapter, we examine how AUVs can be used to track algal blooms in the context of climate change. We highlight the potential of AUVs for data collection on these occurrences and the present status of research on algal blooms and their effects. This thesis aims to contribute to developing more effective

strategies for mitigating the negative impacts of algal blooms and other ocean phenomena on marine ecosystems and coastal communities.

The rest of this chapter is organized as follows. In Section 1.1, we present our motivation for multi- and single-agent systems tracking mobile targets and fronts, considering the phenomena of harmful algal blooms and salinity plumes. In Section 1.2, we formulate the main problem considered in the thesis. The thesis outline and related contributions are presented in Section 1.3.

1.1 Motivation

Researchers worldwide have been using satellites, remote sensing, and buoys to gather information on ocean phenomena and inform forecast models. These methods tend to be expensive, inefficient for spatial coverage, or unable to provide trustworthy data. They always have a human in the loop for decision-making or data post-processing. Having a human in the loop introduces delays and raises the cost. Humans aren't able to handle the vast amount of data generated by satellites, remote sensing, and buoys efficiently. With this thesis, we formulate different scenarios and design and implement appropriate solutions for autonomous coastal surveys using AUVs.

All over the world, the phenomena of harmful algal blooms, as seen in Fig. 1.1, occur frequently and with increasing impact. It has a substantial negative effect on the environment and human health. Therefore, plenty of research has been done regarding the nature of this phenomenon, its causes, and its impact. For example, the Swedish Meteorological and Hydrological Institute (SMHI) has been documenting algal blooms in the Baltic Sea via satellite and monthly missions of a manned research vessel for around 20 years [1]. This thesis suggests novel autonomous approaches to monitor algal blooms and other biological phenomena at sea. Our approaches include a multi- or single-agent marine system, satellite imagery, and numerical models.

Fronts are boundaries between water masses that differ significantly in the value of one or more variables, such as temperature, salinity, or substance concentrations. These fronts shape marine ecosystems as their presence indicates the occurrence of several physical and biological processes of interest, including transition zones, jets, eddies, and phytoplankton blooms [2], [3]. Among these frontal phenomena, harmful algal blooms (HABs) are the motivating scenario of this thesis. Harmful algal blooms are a phenomenon where plankton algae grow rapidly and form very large populations in a short time, which results in



Figure 1.1: Satellite sensor MODIS (Moderate Resolution Imaging Spectroradiometer) provided a satellite picture showing algal blooms in the Baltic Sea in July 2005. (Courtesy of SMHI)

the production of harmful toxins [4]. Algal blooms occur in all types of water: at sea, around the coast, lakes, and streams, both in Sweden and abroad. Usually, algal blooms are mostly near the water surface because the sunlight is strongest there. According to [5], algal blooms cause human illness, large-scale mortality of fish, shellfish, mammals, and birds, and deteriorating water quality. Important questions are why these phenomena occur and why they have been growing over the past years. One of the reasons is climate change, as discussed in [6]. In that study, they infer that climate change will influence marine planktonic systems globally and that it is conceivable that algal blooms may increase in frequency and severity. Higher temperatures and ocean stratification are beneficial for algal bloom species. Also mentioned in [6], agricultural practices and other land usage are important.

These toxins can cause significant harm to marine ecosystems and pose a danger to human activities in the Baltic Sea, such as tourism and aquaculture. Accurate information about algal blooms' location and movement patterns is crucial to monitor and mitigate these detrimental effects. Traditional methods for observation, such as satellite imaging or ship-towed sensors, are generally unable to provide measurements at the spatial and temporal resolutions required

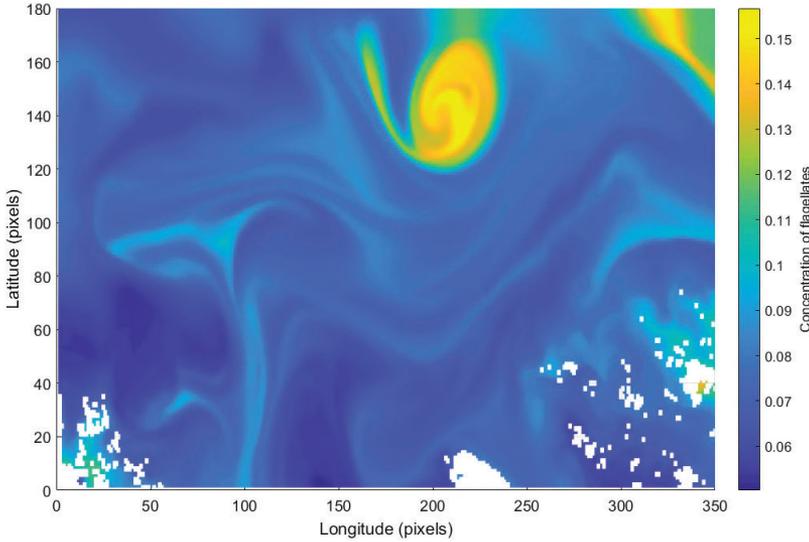


Figure 1.2: SINMOD simulation of concentration of flagellates in the Norwegian sea. (Courtesy of SINTEF)

to understand dynamic ocean phenomena [7]. While remote sensing with satellites can offer a broad overview, such data is weather-dependent and prone to false positives in coastal areas. Thus, there is a significant scientific and societal interest in obtaining *in situ* measurements and developing systems for automated monitoring.

There are many studies on the dynamical modeling of algal blooms. There have been simulation studies of the dynamics of algal blooms, specifically diatoms, and flagellates, two species of algal blooms. Throughout this thesis, we will use SINTEF's numerical ocean model simulation system called SINMOD. Fig. 1.2 shows a snapshot of a SINMOD simulation of flagellates near the Norwegian Sea coast. Each pixel is about 100 meters, so the image is about 35 km longitude and 18 km latitude. Also, an algal bloom is defined as an area with concentrations above a certain threshold, for example, above $0.13\text{mg}/\text{m}^3$. There is an algal bloom shape in the upper center of the figure and a part of another algal bloom shape in the upper right corner. In most results, this simulated abundance and distribution of diatoms and flagellates change remarkably during the highly dynamic spring bloom, and during the summer [8]. In [9], it is stated that future advances in modeling will occur through the junction of models and data,

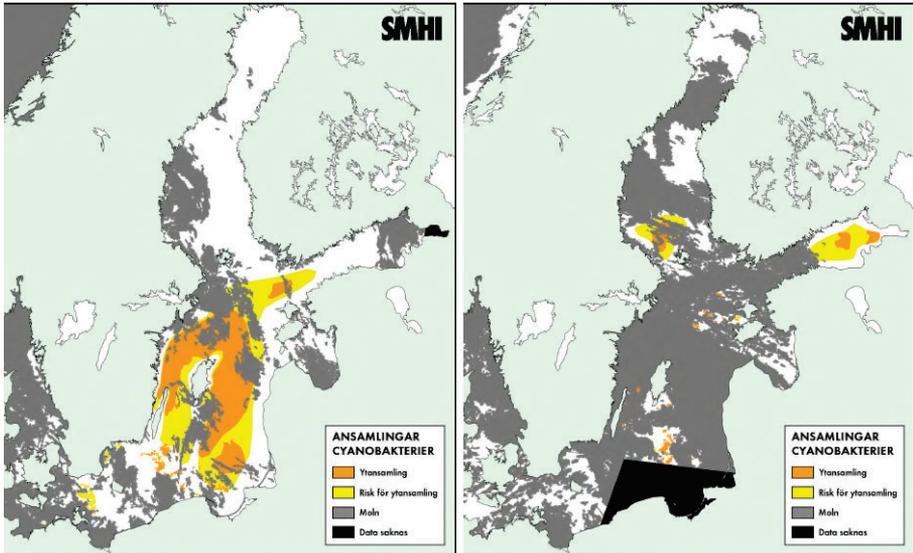


Figure 1.3: Satellite data collected by SMHI. Left: Taken on the 5th of August 2019. Right: Taken on the 6th of August 2019. Legend: Orange for high algae concentration; yellow for risk of high algae concentration; grey for the presence of clouds; and black for missing data. (Courtesy of SMHI)

using data to conceptualize models and using models to understand data. This chapter reviews many available dynamical models and the need for modeling harmful algal blooms.

There are a few approaches to solving the problem of algal bloom data collection and modeling. Two of the most common methods are satellite imagery and monthly missions on a manned research vessel. Using satellite imagery, SMHI has been monitoring the algae situation since 2002 through the Baltic Algae Watch System [10]. This is a satellite-based monitoring system for blue-green (cyanobacteria) algal blooms in the Baltic Sea. Fig. 1.3 represents the data SMHI collected in the summer of 2019, available on their website. Comparing the data from the two consecutive days illustrates how noisy and unpredictable the satellite imagery on algal blooms can be. In the left image, we can see most of the Baltic Sea and thus infer the location of the algal blooms. However, in the right image, one day after, we can barely locate the algal blooms, for instance, off the coast of Stockholm. Clouds cause this difficulty, a common occurrence in this region. In fact, [11] states that turbid coastal and inland waters are a ma-

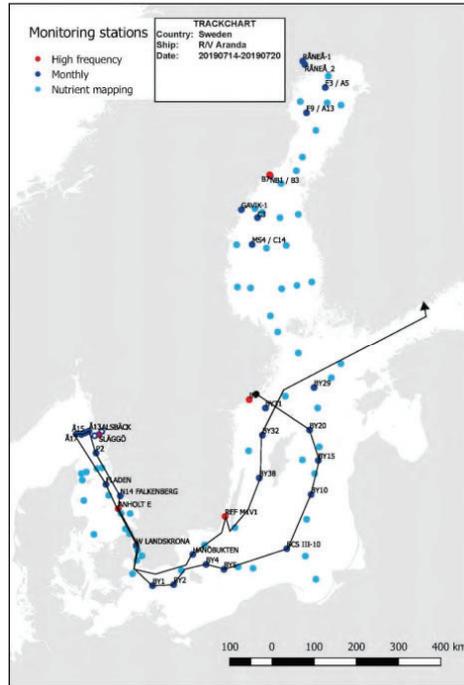


Figure 1.4: SMHI's research vessel mission on the report of July 2019. Legend: Red for a high frequency of data collection; dark blue for monthly data collection; light blue for nutrient mappings; and black line for the vessel's path. (Courtesy of SMHI)

major challenge of atmospheric correction. Local measurements are taken using a long list of sensors and a team of researchers using a research vessel. The procedure and results of each mission are detailed in a report. Fig. 1.4 represents one mission in the report of July 2019 [12]. This figure represents the plan for data collection, including fixed monitoring stations and defined collection points through which the research vessel would pass and collect data.

There is also interest in studying river fronts to understand the dynamics of estuarine circulation and time scales of river flow changes that affect the dispersion of contaminants originating from runoff and wastewater discharges [14]. Typically, the river front has a distinct color or surface temperature. Therefore, river fronts are also studied using numerical models and satellite imagery [15] or by taking water samples or doing Conductivity Temperature and Depth (CTD)



Figure 1.5: Four USVs "duckling" used in [13]

casts from a manned vessel. However, satellite imagery has poor temporal and spatial resolutions, is affected by cloud coverage (as seen in Fig. 1.3), and can only provide a 2D front characterization [16]. AUV-based field studies are now addressing some of these limitations because AUVs can sample the water column continuously and may adapt the sampling strategy to the observations [17].

The solution we provide seeks to substitute this expensive manned mission that occurs once a month with a more affordable, continuous, and autonomous option. For example, with a multi- or single-agent setup using agents such as the Unmanned Surface Vehicle (USV) from KTH, as seen in Fig 1.5, and an algorithm capable of autonomously following and enclosing algal blooms targets or fronts. Multi-agent systems are particularly advantageous for monitoring and detection tasks in ocean environments because they simultaneously enhance coverage of larger spatial areas, thus reducing the time required to survey the ocean. These formations can be tailored to the shape and dynamics of the ocean features, such as circular blooms or ocean fronts. Furthermore, the redundancy provided by multiple agents within a formation offers robustness against sensor noise or communication failures, which is critical in unpredictable environments like the ocean.

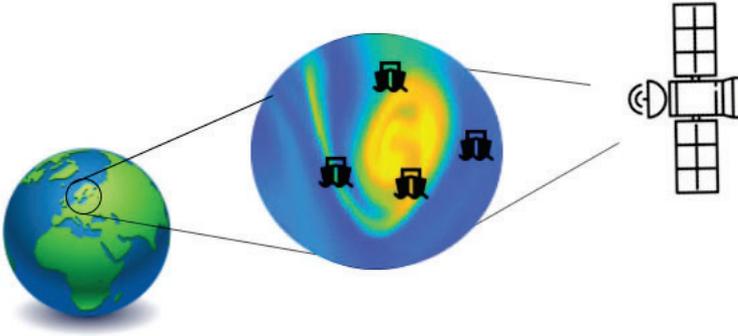


Figure 1.6: Tracking an algal bloom using a multi-vehicle system with local sensors and a satellite

1.2 Research questions

In this thesis, we consider the problem of developing estimation and control algorithms to track ocean features using AUVs. As shown in Fig. 1.6, we consider a system composed of a satellite that provides noisy and cloudy imagery once a day to a system containing one or multiple AUVs. Each of the AUVs in the system is equipped with a GPS receiver for localization and various types of sensors, depending on the features we would like to track and map. One of the most pragmatic examples of such a feature is algal blooms, as illustrated in the figure. The ocean feature to track may be static or dynamic, and it may be a circular shape or a long front. Changes may occur quickly or slowly, according to wind, temperature, ocean currents, etc.

This thesis considers the following research questions:

1. How to control a multi-AUV system to a desired formation?
2. How to guarantee convergence to the feature in an evenly spaced formation?
3. How to follow and map an ocean feature using a single- or multi-AUV system?
4. How to estimate an ocean feature using local AUV measurements and satellite imagery?

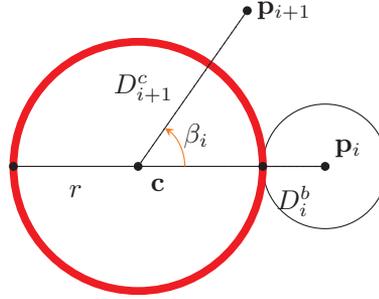


Figure 1.7: Scheme of the estimated c and r as well as the angle β_i measured from c between two agents at positions p_{i+1} and p_i .

5. How to design and execute an experimental test in which an AUV tracks an ocean feature using the developed solutions?

Chapters 3 and 4 answer Questions 1, 2, 3, and 4 with estimation and circumnavigation protocols that guarantee convergence of a multi-AUV system towards a circular target in a regular polygon formation. Chapter 5, 6, and 7 answer Questions 3 and 4 with estimation and control algorithms for tracking ocean feature fronts using a single-agent system. Chapter 8 answers Question 5.

We design estimation and control algorithms for all chapters that match the problem's constraints and objectives. The estimation algorithms in this thesis use two sets of variables: system state variables and measurement variables. The control algorithms use three sets of variables: system state, measurement, and estimated variables (from the estimation algorithm). Fig 1.7 illustrates an estimation and tracking scenario to help us introduce the three types of variables. In this scenario, we consider a multi-agent system estimating features that can be represented by a circular target and maintaining a polygon formation around it. The variables for the state of the system are the position and velocity of each agent defined as p_i and \dot{p}_i , for $i = 1, \dots, n$, respectively, where n is the number of vehicles. The measurement variables concern the distance of a vehicle to the target. We define the distance to the center of the target as D_i^c and the distance to the boundary of the target D_i^b . We also define the angle between agent i and agent $i + 1$ as β_i . After estimating the target's location, size, or curvature, we obtain the estimated variables, defined as the center c and the radius r . The control algorithm is defined as a function of these variables:

$$u_i = f(p_i, p_{i+1}, \hat{c}, \hat{r}, D_i^c, D_i^b)$$

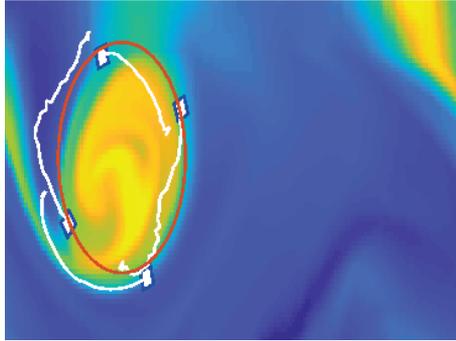


Figure 1.8: Example of a multi-agent system of 4 robots tracking an algal bloom feature.

We also design estimation algorithms which, in this example, corresponds to computing $\hat{\mathbf{c}}$ and \hat{r} based on \mathbf{p}_i , \mathbf{p}_{i+1} , \mathbf{c} , r , D_i^c , and D_i^b .

The problem we consider is how to design the control law and the estimator to ensure the tracking and formation objectives. In this scenario, such objectives would be to 1) guarantee convergence to the boundary of the target and 2) space the AUVs equally along the boundary. Fig. 1.8 shows how four AUVs successfully track an algal bloom target. Notice how the vehicles are very close to the boundary and equally spread across the shape.

We introduced the complex nature of the problem, highlighting the diversity of ocean features, the dynamic environmental factors that influence them, and how current technology isn't enough. We have also clarified the role of state, measurement, and estimated variables in our algorithms, providing a comprehensive framework for understanding our proposed solutions. The research questions presented here guide the subsequent chapters, with each chapter addressing specific aspects of the problem.

1.3 Thesis Outline and Contributions

In this section, we provide an overview of the thesis. We describe each chapter's contents and contributions and indicate the publications upon which they are based.

Chapter 2: Background

In Chapter 2, we provide the background of the thesis by analyzing related literature. It contains state-of-the-art solutions for the previously formulated marine sampling problem, including multi-agent and single-agent systems. We first analyze the tools and technologies available for marine sampling and their usual applications and shortcomings. Within multi-agent systems, we overview the main challenges in cooperation and formation control and the main arrangements such as leader-follower and decentralized coordination. We also evaluate target tracking and estimation as a common application for these systems. We review different control and estimation methods within single-agent systems, focusing mostly on AUVs. Finally, we delve into adaptive sampling methods for front tracking, outlining different fronts.

Chapter 3: Leader-based Adaptive Target Estimation for Multi-agent Systems

Chapter 3 considers the problem of tracking a mobile target using adaptive estimation while circumnavigating it with a system of AUVs. The mobile target considered is an irregular dynamic shape approximated by a circle with a moving center and varying radius. The AUV system is composed of n AUVs, of which one is equipped with an unmanned aerial vehicle (UAV) capable of measuring both the distance to the boundary of the target and its center. The AUV equipped with the UAV uses adaptive estimation to calculate the location and size of the mobile target. The AUV system must circumnavigate the boundary of the target while forming a regular polygon. We design two algorithms: One for the adaptive estimation of the target using the UAVs measurements and another for the control protocol to be applied by all the AUVs in their navigation. The convergence of both algorithms to the desired state is proven up to a limit bound. Two simulated examples are provided to verify the performance of the algorithms designed.

This chapter is based on the following contribution:

J. Fonseca, J. Wei, T. A. Johansen, and K. H. Johansson, "Cooperative circumnavigation for a mobile target using adaptive estimation," *CONTROLO* 2020, Vol. 695, p. 33-48.

Chapter 4: Decentralized Target Estimation for Multi-agent Systems

Chapter 4 proposes a reliable method to track a target using a set of AUVs. A satellite image indicates the existence and initial location of the algal bloom for the deployment of the robot system. The algal bloom area is approximated by a circle with time-varying location and size. This circle is estimated and circumnavigated by the robots, which can sense its boundary locally. A multi-agent control algorithm is proposed to continuously monitor the dynamic evolution of the algal bloom. The algorithm comprises a decentralized least squares estimation of the target and a controller for circumnavigation. We prove the convergence of the robots to the circle in equally spaced positions around it. Simulation results with data provided by the SINMOD ocean model are used to illustrate the theoretical results. We also consider the extension for irregular shapes. We assume that each vehicle measures its distance to the boundary of the target and whether it is inside or outside such target. The convergence of both algorithms to the desired state is proven up to a limit bound. Two simulated examples are provided to verify the performance of the algorithms designed.

This chapter is based on the following contribution:

J. Fonseca, J. Wei, K. H. Johansson, and T. A. Johansen, “Cooperative decentralized circumnavigation with application to algal bloom tracking,” 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), p. 3276-3281.

Chapter 5: Linear Front Approximation for Single-agent Systems

Chapter 5 considers the problem of tracking and mapping a river front with an AUV. Our approach estimates the front direction and adapts the vehicle’s trajectory to retrieve the maximum number of perpendicular transects to the front to improve the mapping performance. The novel aspects of the approach include an exploration phase, in which essential parameters of the front are estimated, the adaptation strategy, encompassing tracking and mapping, and the use of several parameters to enable the user to select and configure the algorithm, as well as to pick the initial deployment locations, according to the environmental conditions. The approach is evaluated in simulation with the help of a high-resolution hydrodynamic model, considering different ocean and meteorological conditions.

This is done with the help of parameterizable motion control algorithms to enable adaptation to these time-varying conditions.

This chapter is based on the following contribution:

D. Teixeira, J. Sousa, R. Mendes, and J. Fonseca, “3D Tracking of a River Plume Front with an AUV,” IEEE OCEANS 2021, San Diego-Porto, p. 1-9.

Chapter 6: Least Squares Front Estimation for Single-agent Systems

In Chapter 6, we propose an algorithm for estimating the gradient of chlorophyll a concentration and a control law for the AUV heading. We run a numerical evaluation of an algal bloom front tracking strategy using Baltic sea data. We define the front as a dynamic curve corresponding to the level set of the chlorophyll a concentration with reference value obtained from satellite data. The AUV records its position and the concentration at each position. The estimation is performed at each timestep in a receding-horizon fashion using the latest data points collected by the AUV. The control input is a heading reference computed as a function of the estimated gradient, such that the AUV moves towards the front when it is far away and along the front when it is close to it. The AUV will remain close to the front after reaching it for the first time.

This chapter is based on the following contribution:

J. Fonseca, M. Aguiar, J. Sousa, and K. H. Johansson, “Algal Bloom Front Tracking Using an Unmanned Surface Vehicle: Numerical Experiments Based on Baltic Sea Data,” IEEE OCEANS 2021, San Diego-Porto, p. 1-7.

Chapter 7: Gaussian Process Front Estimation for Single-agent Systems

Chapter 7 proposes an algorithm containing a control law, a Gaussian Process (GP) estimator, and a sensitivity analysis with an estimation method comparison. We implement a path-planning guidance law with adaptive sampling for controlling the AUV heading. This control law receives information on the position of the AUV, the chlorophyll a concentration measurement it took, and the estimated gradient of the chlorophyll a concentration. We analyze the performance of the control law for different sensor uncertainty levels of chlorophyll a concentration, with full AUV and chlorophyll a concentration sensor model. We propose an algorithm using GP that estimates the gradient of chlorophyll a

concentration. The AUV records its position and chlorophyll a concentration as it moves. The estimation is updated at each timestep by fitting a GP model with the latest data points collected by the AUV. We run a sensitivity analysis in which we vary the standard deviation of the chlorophyll a sensor and measure the impact on the performance of the control law and the estimation method, considering the full AUV and chlorophyll a sensor model. We consider two estimation methods, one based on Gaussian Process regression and the other based on least squares regression.

This chapter is based on the following contribution:

J. Fonseca, A. Rocha, M. Aguiar, and K. H. Johansson, “Adaptive Sampling of Algal Blooms using an Autonomous Underwater Vehicles and Satellite Imagery,” 2023 IEEE Conference on Control Technology and Applications (CCTA), Bridgetown, Barbados.

Chapter 8: Marine Experiments

In Chapter 8, we design an experimental setup and present the results of a realistic simulation and a field experiment in the Baltic sea. We design an experimental setup consisting of a cyber-physical system integrating the AUV software, the AUV hardware, the user interface, and a realistic simulator. The AUV software includes numerous packages that can be divided into the behavior tree, the algal bloom front tracking, the onboard controllers, and the dead-reckoning. The algal bloom front tracking library has been developed for the present work and includes the control law and two implemented estimation methods. The algal bloom front tracking package was also developed for the present work and is responsible for interfacing with the other AUV packages, simulating the chlorophyll a sensor, and plotting. We provide experimental results from two surveys in the Stockholm archipelago in the Baltic Sea. In these experiments, we demonstrate that the proposed algorithm performs well in the real-time real-world scenario and compare them to a simulation under experiment conditions. We also examine the sources of error, namely surface waves that influence the AUV’s movement but also partially occlude the GPS receiver, which introduces noise on the GPS-measured position of the AUV.

This chapter is based on the following contribution:

J. Fonseca, S. Bhat, M. Lock, I. Stenius, and K. Johansson, “Adaptive Sampling of Algal Blooms Using Autonomous Underwater Vehicle and Satellite

Imagery: Experimental Validation in the Baltic Sea,” submitted to IEEE Journal of Oceanic Engineering.

Chapter 9: Conclusions and Future Work

Finally, in Chapter 9, we conclude the present thesis with a summary and discussion of the results and directions for future work, indicating some possible extensions of this thesis and future research directions.

Appendix A: Citation Analysis

Appendix A follows a citation analysis that reflects on the articles cited in this thesis and, more broadly, on the article diversity of the decision and control systems and maritime robotics fields.

Publications not included in the thesis

The following publications are not covered in this thesis but are related to the work presented here.

Z. Yang, J. Fonseca, S. Zhu, C. Chen, and K. H. Johansson, “Adaptive Estimation for Environmental Monitoring using an Autonomous Underwater Vehicle,” 2023 IEEE Conference on Decision and Control (CDC), Singapore.

Z. Yang, J. Fonseca, S. Zhu, C. Chen, and K. H. Johansson, “Distributed Formation Control for Environmental Monitoring: A Gradient Estimation-based Approach,” submitted to IEEE Transactions on Automatic Control.

Chapter 2

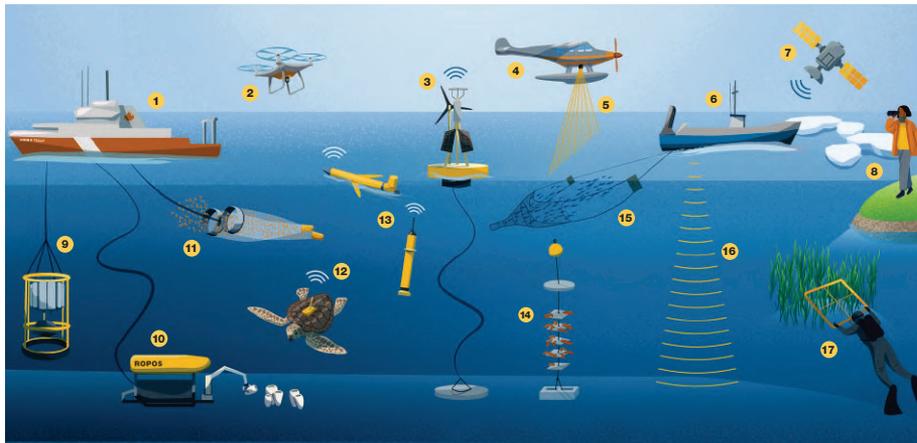
Background

This chapter provides a background for the thesis by analyzing related literature. It contains the state-of-the-art solutions for the problems introduced in Section 1. Most of those solutions consist of autonomous systems for marine sampling applications. These include single-agent systems such as stand-alone AUVs and multi-agent systems such as swarms of AUVs or heterogeneous sampling networks consisting of AUVs, buoys, remote sensing, and others. In the field of multi-agent systems, some relevant challenges lie in the cooperation and formation control between agents, resulting in different arrangements, such as leader-follower or decentralized. This type of system tends to be ideal for target tracking and circumnavigation applications. When looking at single-agent systems, the challenges focus on the performance of the single agent, namely, its ability to estimate an unknown field and adapt its movement with respect to the changes in the field, which is referred as adaptive sampling. Naturally, the challenges regarding single-agent performance are also encountered in a multi-agent system.

The rest of this chapter is organized as follows. In Section 2.1, we introduce the different sampling platforms commonly used for marine sampling. In Section 2.2, we discuss current work on multi-agent systems for target tracking for multiple arrangements regarding cooperation and formation control. Recent developments in single-agent systems follow in Section 2.3, focusing on adaptive sampling for front tracking.

Ocean Sampling Technologies

Technologies—old and new—help to fill gaps in knowledge of Canada’s vast and complex oceans



- 1 Oceanographic missions** support multiple technologies and ocean monitoring
- 2 Small drones** with cameras are used to take aerial photos of habitats and species
- 3 Buoys and other moorings** use sensors to take frequent measurements of ocean conditions at a single location for up to a year
- 4 Aerial surveys** are used to monitor marine mammals, seabirds, and habitats
- 5 LIDAR (light detection and ranging)** is a remote sensing technology used to monitor habitats from the air
- 6 Research vessels** are used to conduct multispecies and acoustic surveys
- 7 Satellites** gather data and images year-round about temperature, sea ice, and phytoplankton
- 8 Observers** document occurrences of seabirds and marine mammals
- 9 Rosettes** collect water samples and carry sensors to measure environmental conditions throughout the water column
- 10 ROVs (remotely operated vehicles)** explore habitats far below the surface
- 11 Plankton nets** collect phytoplankton and zooplankton samples
- 12 Satellite tagging** tracks migratory species such as turtles, fish, sharks, mammals, and seabirds
- 13 Gliders and Argo floats** travel long distances on very little power to gather ocean climate data
- 14 Collector plates** are used to sample invasive tunicate species that attach to them
- 15 Trawl nets** collect samples of invertebrates and fishes
- 16 Sonar** is used to collect data on fishes and zooplankton in the water column and to conduct hydrographic surveys and classify seafloor habitats
- 17 SCUBA divers** collect samples and data in coastal areas

Figure 2.1: An illustrated diagram shows a simplified side view of the ocean and coastline with multiple types of sampling technologies used to collect data about ocean science. Each illustration is linked to a number describing the technology. (Courtesy of Government of Fisheries and Oceans Canada [18])

2.1 Marine Sampling Platforms

There has been significant work in investigating and developing solutions for marine sampling. In Fig. 2.1, several examples of technologies and platforms help fill the marine sampling knowledge gap. Notably, less than one-third of these examples require direct human intervention, and less than one-fifth can be considered autonomous. Regardless of the level of automation and human intervention, we can verify that worldwide, the solution for marine sampling is always the product of different sampling platforms and technologies for varying

applications. Beyond different platforms, a study [19] found that a comprehensive approach to ocean monitoring should connect academic practices and local community and indigenous practices. This Section gives an overview of the most commonly used sampling platforms and technologies for marine sampling.

Some of the most used platforms and technologies for marine sampling include satellites, research vessels, floats, gliders, or AUVs. Satellite imagery offers broad datasets with varying resolutions but falls short for shallow waters when measuring algal blooms. This is because satellite sensing relies on chlorophyll *a* reflectance, which occurs for any underwater plant beyond algal blooms. There is a high density of plants in shallow waters and, thus, many false positives for algal blooms. Despite the shortcomings of satellite imagery for coastal data, alternate solutions for near-shore regions have been the topic of extensive research for the past 50 years in efforts to bridge that gap [20]. For example, in [21], the authors propose a novel wavelet analysis with satellite data from repeating paths for coastal coverage. On the other hand, several solutions for *in situ* data collection, such as research vessels, generally use towed platforms integrating cameras, sensors, acoustic devices, and sidescan data [22]. In some cases, they also use underwater electronic holographic cameras for studying marine organisms such as plankton [23].

Focusing now on autonomous *in situ* sampling, one of the most widely used platforms is the profiling float. This float moves with underwater currents and is sometimes referred to as a Lagrangian float in the literature. Profiling floats are more controllable in regions of more diverse currents, but controlling the horizontal motion of a profiling float remains challenging [24]. Despite their limited controllability, profiling floats such as in the Argo Program are perhaps the greatest international collaborative effort in the history of oceanography and provide researchers with open access to comprehensive data sets [25]. Underwater gliders are more controllable than floats; in [26], the authors who built the glider "Spray" define gliders as *autonomous profiling floats that use a buoyancy engine to move vertically and wings to glide horizontally*. The creators of the glider "Slocum" stated that their glider originated from the idea of adding horizontal propulsion to floats [27]. The "Seaglider" [28] is another example of one of the first gliders to be developed. With the development of gliders, the problem of under-actuated controllability has been a subject of analysis [29]. Gliders fall short in deterministic controllability, usually have a very limited sensor payload, and require a minimum operating depth of approximately 50m. These make them unsuitable for coastal applications in shallow water. On the

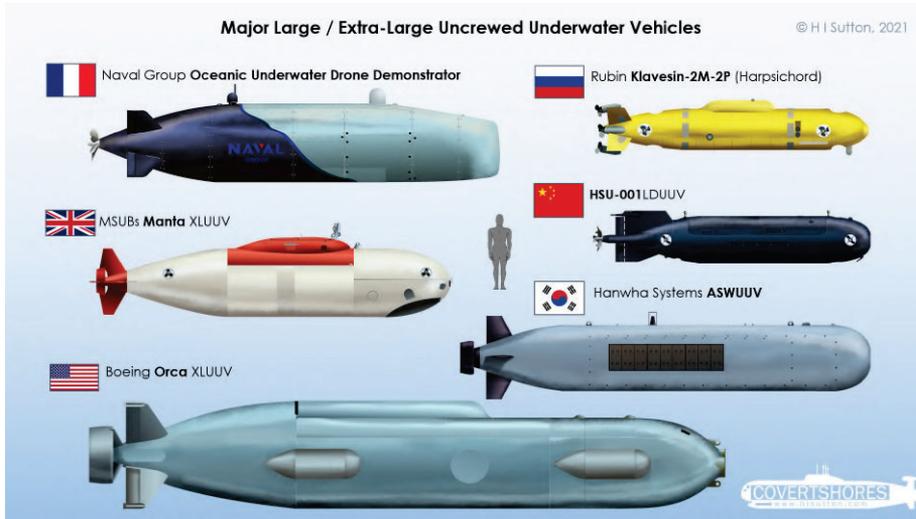


Figure 2.2: Examples of the variety of autonomous underwater vehicles worldwide. (Courtesy of Covert Shores)

other hand, AUVs can carry different payload sensors, have better controllability, and operate at multiple depths.

AUVs are autonomous and highly controllable underwater vehicles that can be shaped and sized with great variety, see Fig. 2.2, and have been considered for solving the environmental sampling problem. One of the most common solutions in the literature is the open-loop scenario with a fixed sampling pattern. The most widely used sampling pattern is, undoubtedly, the lawn-mower [30], which has been used for both single-agent [31] and multi-agent scenarios [32]. However, other relevant methods, such as the spiral and circular patterns in [33], aim to improve efficiency and robustness. Or the oval spiral coverage strategy to plan coverage paths that better suit oval-shaped areas of interest [34]. While these open-loop strategies enable and even guarantee coverage of survey areas, they are inherently not designed to react or respond to changes in the observed features. In such cases, there is great motivation to close the loop.

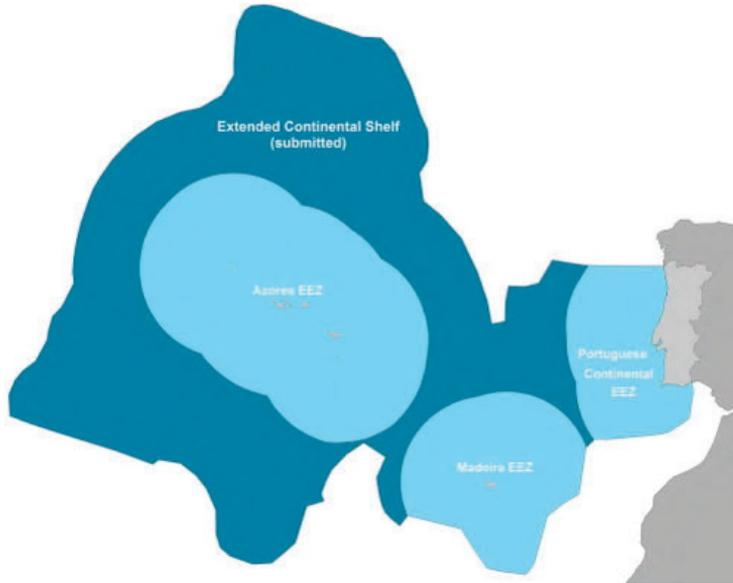


Figure 2.3: Pictographic representation of the Portuguese ocean and land territory.

2.2 Multi-agent Control for Marine Sampling

In this Section, we overview the challenges and developments of multi-agent systems for the application of marine sampling. Marine sampling is a field that has been growing over the past years as the need for ocean monitoring has increased. Unmanned vehicles have been recognized to allow higher precision and cost efficiency levels in many research expeditions [35]. Moreover, using a multi-agent system introduces more robustness by eliminating the single point of failure. As a result, control of multi-agent systems presents itself as an essential component of the problem of marine sampling.

Marine sampling is one of this decade's key challenges and prominent investments. There is a need for a sustained, persistent, and affordable presence in the oceans. Oceans cover 96% of the Earth, thus making ocean observation a problem on a truly planetary scale. This problem is particularly important to countries with plenty of ocean territory, such as Portugal, as depicted in Fig. 2.3. In a book on the future of the Portuguese ocean, Sousa et al. [36] describe that constant ocean monitoring is necessary, albeit not an easy task. They claim that

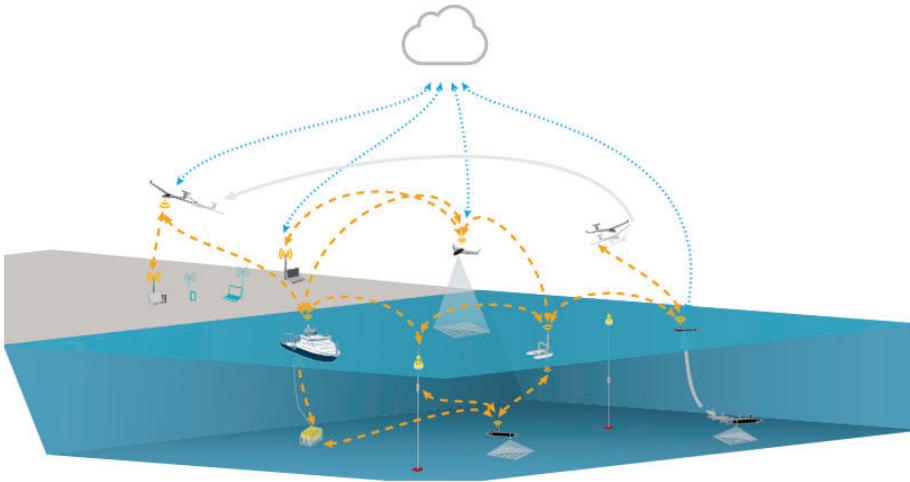


Figure 2.4: Design, construction, and operation of unmanned underwater, surface, and air vehicles development of tools and technologies for the deployment of networked vehicle systems. (Courtesy of LSTS [37])

some key applications are understanding and monitoring climate change, ocean acidification, unsustainable fishing, pollution, waste, loss of habitats, biodiversity, shipping, security, and mining. They further claim that such goals can only be achieved by *an incremental and multi-dimensional approach* including two steps: First, an increase in the number of systems in operation in the oceans with new fleets of robotic vehicles of unprecedented spatial and temporal resolution. Second, networking existing systems and new robotic vehicle systems for coordinated adaptation to observational needs. An illustration of such a system is represented in Fig. 2.4.

Over the past few decades, a vast body of research on multi-agent systems has been conducted. Compared to a more complex single agent, their durability, affordability, and efficiency are frequently the causes of such interest. A multi-agent system is composed of interconnected agents coordinated to complete a designated task. Each agent has its dynamics and communicates with a subset of the other agents, possibly influencing each other's decisions. We can identify various types of multi-agent systems depending on two factors: each agent's sensing capability and the interaction topology of the agents [38]. Multi-agent systems are sometimes inspired by multi-organism partnerships, for example,

ants building an underground home, birds flying energy efficiently, or even humans working together on a project. Multi-agent systems, for example, systems of AUVs, can be applied to monitor harmful algal blooms in the Baltic and Norwegian seas. These algal blooms threaten the environment and human health; therefore, there is a growing need to study their evolution in real-time.

Unmanned vehicles are especially useful in difficult or dangerous environments and when real-time data exchange is necessary [39]. In [40], it is stated that autonomous systems are becoming more powerful and utilize the capabilities of several types of devices, such as UAVs, USVs, and AUVs.

2.2.1 Cooperative Control of Multi-agent Systems

Multi-agent systems present numerous control challenges. The benefits that cooperative multi-agent systems offer have inspired extensive research efforts. Murray [41] classifies multi-agent systems challenges into three main categories: (i) the uncertainty caused by inter-agent communications and distributed operation, (ii) system complexity due to the problem size and coupling between tasks, and (iii) scalability to a potentially large collection of agents. Recently, Cao et al. [42] defined four main directions of research: consensus, formation control, optimization, and estimation. Cooperative circumnavigation for target tracking is a particular problem within cooperative multi-agent control. The literature on this topic is, in fact, quite extensive and spans over 20 years of research. Some examples are formation control or cooperative circumnavigation of a known target, formation, and estimation for tracking a moving target defined as a unit point, and finally, more recently, estimation protocols for moving targets of particular shapes and sizes.

Many algorithms within multi-agent formation for target tracking are related to formation control to observe a known target and, therefore, do not require an estimation or circumnavigation. This type of work focuses, for instance, on fast and energy-efficient convergence of each agent to a desired position while sometimes optimizing communication costs. In Fig. 2.5, we see a classical example of formation control without estimation or circumnavigation. One of the earliest results proposes a path-following algorithm for formation control of a multi-agent system [43]. The authors prove that if the tracking errors are bounded, their method stabilizes the formation error. However, it is assumed that there is perfect information about the path to follow. In [44], a control protocol is designed for avoiding obstacles and inter-agent collisions while converging to a specified target position and forming an equilateral triangular formation around

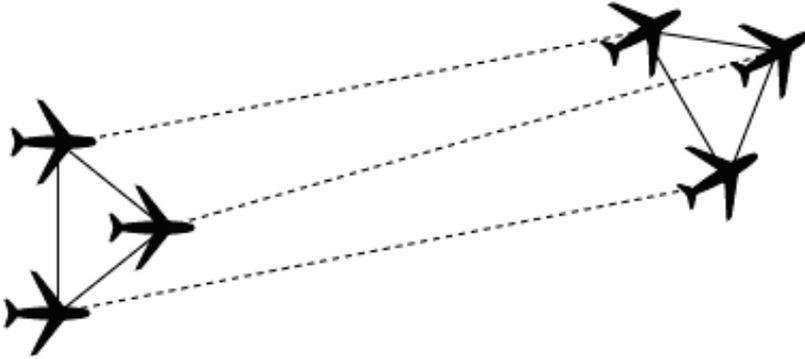


Figure 2.5: Three planes maintain triangular formation while following a path.

the target. Also, in [45], [46], [47], and in Section 6.3.1 of [48], formation protocols are proposed where the robots are capable of converging towards a desired pattern by acquiring their distances between each other. Additionally, in [49] and [50], controllers are synthesized for a swarm of agents to generate a desired two-dimensional geometric pattern specified by a simple, closed planar curve. It is assumed that the shape is given to the swarm and is not estimated in real-time. Finally, an example of optimal circle circumnavigation is presented in [51], where the objective is area scanning. Note that the literature above does not cover target estimation.

2.2.2 Target Tracking and Circumnavigation

There is extensive work on circumnavigation algorithms that integrate formation control with target estimation. A target is generally defined as a moving unit point, and the agents measure and estimate its location. The algorithms tend to be either distance-based, bearing-based, or both. One of the first works on distance-based algorithms deals with agents moving around the target while forming an optimal geometry [52]. In [53], there is only one agent and, therefore, no formation control. This agent can measure its distance to the unit point target and converge to it using sliding mode control. A closely related work [54] proposes an adaptive protocol to circumnavigate around a moving point. The authors employ adaptive estimation for point tracking at a known distance. In [55], a distance-based algorithm for pattern formation is proposed, which guarantees convergence while tracking the target. The agents detect their distance to

other agents and the moving target and follow it while circumnavigating. [56] devises an algorithm such that one robot can circumnavigate a circular target from a prescribed radius using bearing measurements. Several other works propose similar algorithms considering bearing measurements and focusing on other challenges. Such as the challenge of measurement noise amplification, considered in [57], by proposing an estimator that avoids using explicit derivatives of measurements. Also, the challenge of collision avoidance and experimental verification is the focus in [58]. Related results [59] and [60] use either bearing or distance measurements to the target while using a network of autonomous agents to circumnavigate. Circumnavigation is done with a predefined distance to the target, as in [61], where a localization and circumnavigation algorithm of a slowly drifting target is proposed. Here, the authors analyze distance-based, bearing-based measurements and various communication protocols. The target is the biggest distinction between these works and those we develop in this thesis. In the above articles, the target is assumed to be a unit point, and the agents must circumnavigate it at a predetermined relative distance. In contrast, our problem deals with a dynamic irregular shape.

A multi-agent system for tracking oil spill plumes was successfully simulated in a realistic test environment by Fahad *et al.* [62]. This is an estimator-controller system based on the model of a front with a single-point source that causes fluid propagation through advection and diffusion. Li *et al.* [63] developed a tracking approach that relies on specific properties of the oil front. Thus, it is unsuitable for other marine features such as river fronts. There is also work on multi-agent formation control and target tracking when the target is not a unit point. For example, [64] proposes a protocol for target tracking in 3D with guaranteed collision avoidance. The difference is that in [64], it is assumed that the target is a fixed object that may move and rotate but never change its shape, which is different from this thesis' objective. In the literature above, the authors did not account for a shape-shifting target that requires constant measuring and estimation while performing formation control for target circumnavigation.

2.3 Single-agent Control for Marine Sampling

In this Section, we consider systems consisting of a single agent for marine sampling. Similar to multi-agent systems, single-agent systems are a cost-efficient marine monitoring solution for severely undersampled territories and for understanding ocean phenomena. Single-agent systems are deemed particularly

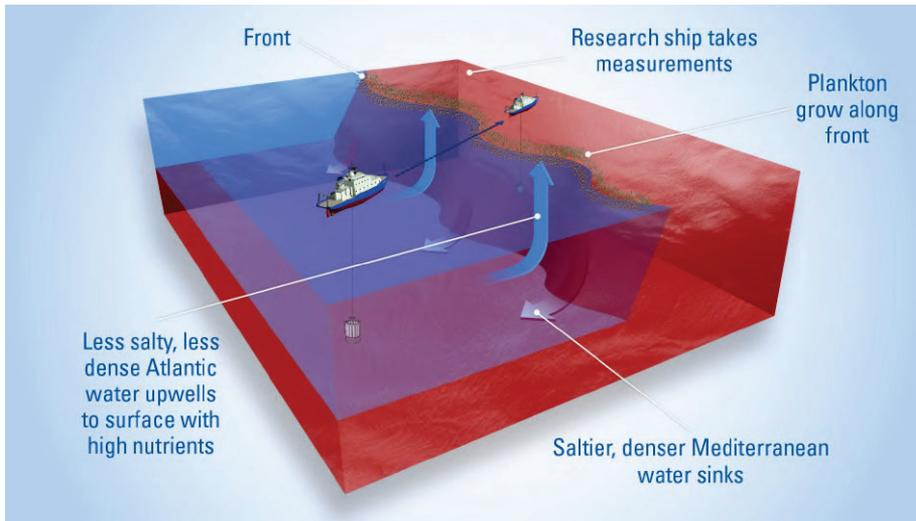


Figure 2.6: Plankton grows more prolifically at fronts where different water masses meet. In this case, less-salty, less-dense water from the Atlantic Ocean enters the Mediterranean Sea, where it meets saltier, denser Mediterranean water. The denser water sinks beneath the lighter water. (Courtesy of WHOI [66])

useful for specific applications while maintaining the possibility for extension to a more complex multi-agent system that may offer more coverage and robustness. For example, in a recent survey [65], the authors consider environmental sensing one of the three main impact areas of AUV research. In this thesis, we consider the application of tracking marine fronts. Adaptive sampling is also a relevant tool for in-situ monitoring tasks aided by external data and numerical models.

2.3.1 Front Tracking

Front tracking is a unique yet widely researched topic for single-agent marine sampling missions. Fronts are unique yet researched extensively because most marine life converges toward them. Less dense water rises along fronts and brings nutrients from the deep into the sunlit surface ocean, as illustrated in Fig. 2.6. Phytoplankton flourishes in these nutritious areas, and so do zooplankton populations. Fish then seek out these areas to feed off of Zooplankton [66]. We can further observe that fishing boats seek these fronts for squid fishing at

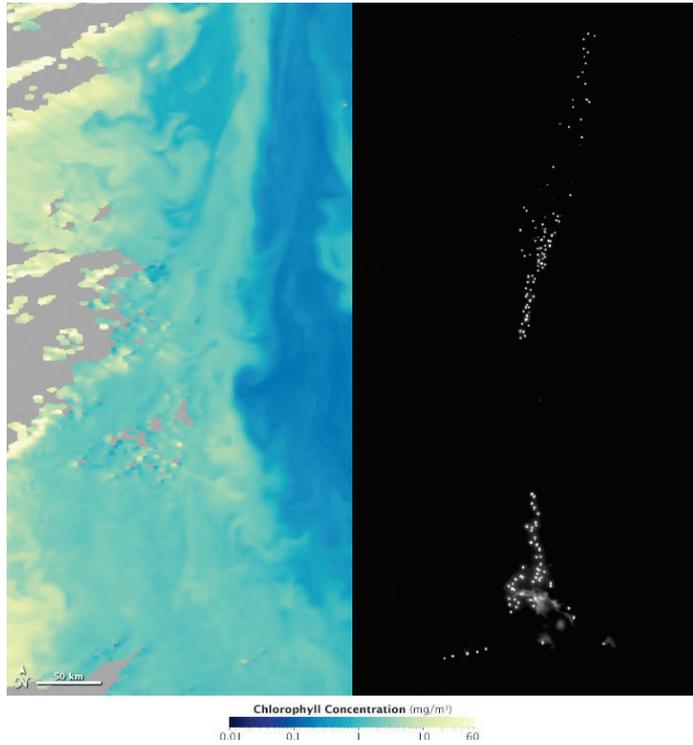


Figure 2.7: On April 18 and 19, 2012, daytime chlorophyll concentrations (left) and nighttime ocean lighting (right) line up in the South Atlantic. (Map by NASA Earth Observatory, using data from the NASA Ocean Color Group [68] and the VIIRS day-night band on Suomi NPP.)

night [67], where algal bloom presence directly correlates to fishing boat distribution, as seen in Fig. 2.7.

As a result, front tracking has received increasing attention in recent years. Ocean fronts are characterized by strong variations of some phenomena or variables such as algal blooms [69], salinity [70], temperature [71], Rhodamine dye [72], oil propagation [73], or other water properties. Front tracking can be divided into front estimation and agent control algorithms. Regarding front estimation, various methods have been considered in the literature, such as model-based estimation for front propagation [62] and local estimation of front direction [74]. Regarding agent control algorithms, we can find both multi-agent [75] and single-agent strategies [76], as well as adaptive algorithms for tracking depth

[77] and non-adaptive zig-zag algorithms for tracking upwelling fronts [78], or velocity fronts [79]. Despite the large differences between these approaches, there are a few common threads, namely the use of zig-zag tracking algorithms for the agents tracking the front, since in most applications, there is a need to collect data in and around the front; the use of either USVs or AUVs with the application-specific sensor for local data collection. An AUV-mounted sonar, capable of detecting hydrocarbon clouds, is at the core of an oil front tracking algorithm that detects the front without crossing it [80]. Zhang *et al.* [81] proposed an approach to detect a horizontally oriented subsurface hydrocarbon front located between depths of 1100 and 1200 m caused by the 2010 Deepwater Horizon oil spill in the Gulf of Mexico.

Marine robots have successfully tracked ocean isolines of salinity or temperature. A segment of the open-ocean Northern Pacific Subtropical Front was autonomously tracked with an AUV running a novel front core tracking algorithm [82]. The parameters of the salinity front were first estimated to inform the tracking algorithm. The proposed algorithm uses a novel model for the salinity signature of the front. Previous algorithms model the salinity signature with a linear function connecting the boundaries of the front. Outside of these boundaries, salinity is considered to be constant. Other algorithms model this signature as a smooth function, with the front located at the peak of the salinity gradient. Belkin *et al.* [82] uses the front edges instead of the front axis to guide the front core tracking algorithm. The vehicle follows the front axis by remaining inside the front. This strategy has two main advantages: the distance traveled by the vehicle is minimized because the front is not fully crossed like in a typical zigzag approach, and the progression along the front progression is faster. Furthermore, the risk of losing the front is small because the AUV is kept inside the front.

An upwelling front-tracking algorithm was field-tested with an AUV in Monterey Bay, CA [83]. The algorithm uses a yoyo vertical trajectory to measure the temperature at different depths while crossing the front at different locations. The multiple front detection points are used to estimate the progression of the front. A similar yoyo trajectory will be used in our implementation to measure salinity at different depths. Analogous to the described system, multiple crossings of the front are then used to estimate the front's progression. A front-tracking algorithm for a thermal front caused by a nuclear power plant outflow was implemented on a USV and successfully deployed in the field [84]. The algorithm proposed an adaptive zigzag trajectory based on measurements. A front indication function then uses these measurements to signal if the vehi-

cle is inside or outside the front. The authors conclude that the dependence of this function on six different variables increases the robustness of the tracking system. A similar zigzag-like trajectory is used in this thesis.

Cannell and Stilwell studied two approaches for static front tracking and mapping [85]. In the first, an AUV travels a fixed path in the study area and then returns a probabilistic distribution of the front based on the measurements and path of the agent. The second approach is a non-parametric boundary tracking algorithm that first runs a classification phase that identifies the two different regions and proposes a probability distribution of the front's boundary that is then used to guide the agent along the front. These approaches do not apply to the dynamic front of the Douro River. Zig-zag trajectories for tracking marine fronts were experimentally validated with AUVs in [79]. The tracking system generates an initial path for the vehicle based on previous characterization of the front and using observations from different remote sensing agents.

There are also chemical fronts, such as a front of Rhodamine dye [86] or an oil front propagation [62]. On the latter, the authors propose a model-based estimator-controller for autonomous robots to track front propagation. This is based on a model of advection-diffusion equation that describes point-source pollution propagation in marine environments [63]. Unfortunately, this algorithm does not apply to other ocean phenomena because it is built on the unique properties of propagation of the oil front.

Coastal upwelling is also an example of an ocean phenomena front. It is a wind-driven ocean process that brings cooler, saltier, and usually nutrient-rich deep water upward. Upwelling fronts support enriched phytoplankton and zooplankton populations [87], thus having great influences on ocean ecosystems. A method for an AUV to autonomously detect and track an upwelling front was developed in [76]. Here, the objective was to transect the front multiple times at the same location to detect the front location. In [78], the method was extended to implement the commonly used zig-zag through the front. While relevant, this objective and application differ from our present thesis.

Oceanic thermal fronts are another example of biological importance and dynamic evolution. In [71], a method was developed for a Wave Glider to detect and track a thermal front autonomously. Similarly to the coastal upwelling scenario, the protocol transects the front multiple times. Also, in [88], a 2D adaptive front tracking algorithm was developed and applied to thermal fronts. However, compared to zig-zag protocols, one weakness is that the start location of the front-tracking mission affects the performance.

2.3.2 Adaptive Sampling

Adaptive sampling is a type of intelligent behavior that allows an agent to make decisions autonomously during a mission in response to the environment and agent state changes. Having a closed-loop control architecture, an AUV can perceive the environment, interpret the data, and take follow-up measures. Over the past two decades, there has been significant effort in closing the loop using adaptive sampling strategies. Adaptive sampling is a closed-loop control architecture in which an agent autonomously makes decisions during a mission in response to environmental and agent state changes. As reviewed in [17], adaptive sampling can be divided into three objectives: source localization, front determination, and tracking and mapping. These objectives can be realized for different targets: thermoclines, algal blooms, oils spills, etc., using different vehicle configurations: single-agent, multi-agent with leader-follower, cooperative multi-agent, etc. In [89], the focus is on covering multiple thermoclines as they evolve in time and space in a dynamic water column. In [90] and [91], the problem of source localization using a multi-agent system is approached as a gradient climb with optimal formation to minimize the gradient estimate error.

An example of front determination is [92], where the authors use a single AUV to find and track a salinity and temperature front while zig-zagging around it. Considering the mapping problem, in [93], the authors evaluate how to find the optimal path to maximize the accuracy of the field estimates for single and multi-agent scenarios. Also, in [94], the front determination problem is considered with a single AUV zig-zagging the front, but for bathymetric contours. In [74], a deep Chlorophyll maximum layer is tracked and mapped using three agents moving adaptively - one on the surface, one tracking the layer, and one mapping the area around the layer. Other examples include an adaptive sampling algorithm that augments a standard Gaussian process (GP) with a nearest neighbor prior [95]. Unlike our approach, this paper does not use external data to aid the vehicle's decisions, while being similar to our method of building a GP model from measurements. Also, in [96], the authors propose a GP model trained with forecasted data and updated in real-time with measurements. This algorithm relies on a squared exponential kernel, which will be considered and discussed further in the present thesis.

In this field, one challenge is integrating external data (satellite imagery, numerical models and predictions, etc.) with local measurements (taken by autonomous agents such as AUVs, buoys, gliders, or others). Consequently, this extends to the problem of creating and maintaining databases with local marine

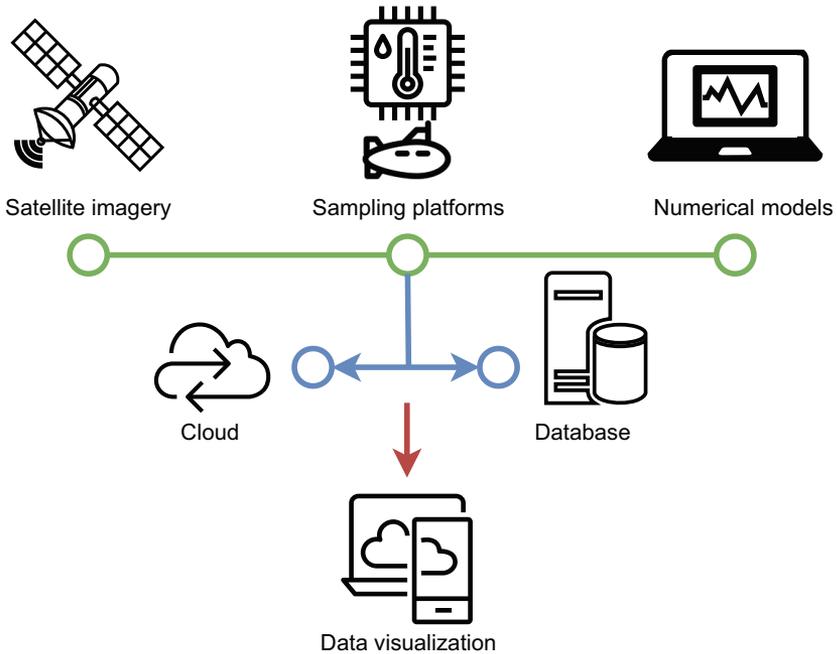


Figure 2.8: Information flow from the different data sources based on ocean observations, data management with cloud storage and local databases, and visualization of complete datasets. (This infographic is inspired by a similar one presented in [97])

data sets to allow for data visualization. This problem is synthesized in Fig. 2.8. In this thesis, we delve into the problem of how to aid ocean sampling missions using external data. An early example uses a predictive ocean model to assist in motion planning for steering an AUV to a high-valued location [98]. Here, it is assumed that there is a predicted model for the day of the mission, which is not available in this thesis’s scenario. Other related results include using knowledge from previous missions to create a model [99] and using hydrodynamic and biological model systems as prior information [100]. Such adaptive sampling is closely linked to data assimilation. Among the first works in data assimilation for oceanic applications are [101], [102]. In [103], the data collected by the AUV is augmented with remote sensing, buoys, a ship, particle imaging systems, and discrete water samples. In [104], the authors use a genetic algorithm to opti-

mize the deployment, measurements, and information gain of a team of AUVs and mooring platforms. Such measurements are then included in an estimation framework to estimate and forecast environmental parameters given a dynamic ocean model. In [105], the authors propose using generic environmental models that are updated with data collected by a team of AUVs to update the map used by these vessels and, with it, perform mission-specific goals. Also, in [106], a sparse and variational Gaussian process is trained with datasets of different seafloor textures and then used for seafloor texture classification. Our approach, in contrast, uses satellite data as prior in a Gaussian Process regression framework that is similarly updated with local measurements from AUVs to create high-resolution estimates of the environment around the autonomous agents.

Ocean ecosystems are greatly influenced by the structure and dynamics of fronts [107]. Detection and tracking ocean fronts are important for investigating the formation, evolution, and interaction of ocean water masses [108]. Knowing the boundary between these water masses enables targeted sampling of the waters. As noted in the previous Section, an ocean front delineates the boundary between water masses distinguished by physical, chemical, or biological characteristics. Some examples of ocean fronts are algae, salinity, and temperature fronts. A lack of efficient observations has hampered progress in understanding the dynamics of fronts. For instance, global satellite measurements of ocean-surface velocities and air-sea fluxes are only available at resolutions of a few hundred kilometers [109]. Therefore, the present thesis suggests approaching data collection using new unmanned sampling strategies for AUVs. AUVs can perform measurement runs over a long time at sea [110], making them a frequent choice for oceanographic data collection [111].

Chapter 3

Leader-based Adaptive Target Estimation for Multi-agent Systems

In this chapter, we discuss the problem of multi-vehicle target tracking. This target is an irregular dynamic shape approximated by a circle with a moving center and time-varying radius. We will use adaptive estimation while circumnavigating the target with a system of AUVs. The multiple AUV system is composed of n AUVs of which one is equipped with a UAV capable of measuring both the distance to the boundary of the target and its center. This AUV equipped with the UAV uses adaptive estimation to calculate the location and size of the mobile target. The AUV system must circumnavigate the boundary of the target while forming a regular polygon.

This chapter is organized as follows. In Section 3.1, we describe the system mathematically and formulate the problem to be solved in the following section. In Section 3.2, we design two algorithms: One for the adaptive estimation of the target using the UAV's measurements and the other for the control protocol to be applied by all AUVs in their navigation. In Section 3.3, the convergence of both algorithms to the desired state is proved up to a limit bound. Finally, in Section 3.4, two simulated examples are provided to verify the performance of the algorithms designed. Concluding remarks come in Section 3.5.

3.1 Problem Formulation

We consider the problem of tracking a shape using a multi-AUV system and a UAV. This target shape may be very irregular and with time-varying parameters.

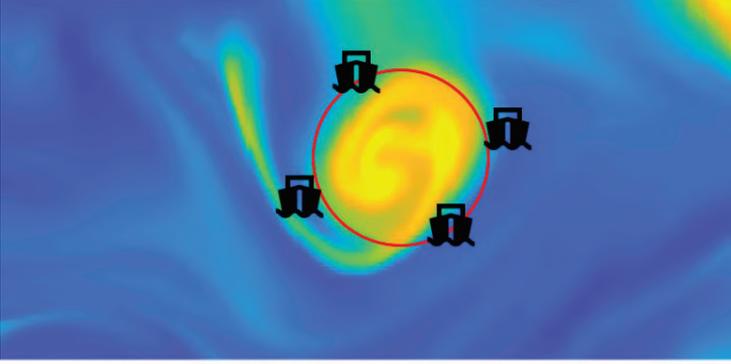


Figure 3.1: 4 AUVs circumnavigating a circular algal bloom

We assume the shape is close to a circle, as seen in Fig. 3.1. The UAV provides an initial image of the target, which confirms such an assumption and helps us deploy the AUVs.

3.1.1 System description

We define this circle as

$$(\mathbf{c}, r) \in \mathbb{R}^3, \quad (3.1)$$

where $\mathbf{c} = (x, y)$ and r are the circle's center and radius, respectively. We denote $(\hat{\mathbf{c}}, \hat{r}) \in \mathbb{R}^3$ as the estimates of the circle. Then the UAV would provide initial estimates $\hat{\mathbf{c}}(0) = (\hat{x}(0), \hat{y}(0))$ and $\hat{r}(0)$.

This UAV obtains data from the target and shares it with the AUVs so they can move toward the target. The UAV constantly measures its distance from the target, calculates its target estimates, and shares it with all AUVs. The measurements consist of its distance to the center and to its boundary. Each AUV has access to its GPS position and the AUV's position in front of it, counterclockwise.

The multi-AUV system will jointly circumnavigate the target and provide real-time information on different fronts. We define n AUVs and, using the UAV information, they are initialized at positions $\mathbf{p}_i(0)$, $i \in [1, \dots, n]$, which are outside of the shape and form a counterclockwise directed ring on the surface. The kinematic of the AUVs is of the form

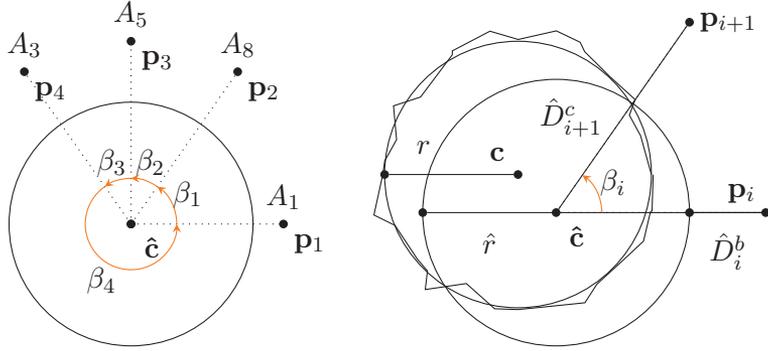


Figure 3.2: (Left) System with vehicles A_1, A_3, A_5, A_8 at positions $\mathbf{p}_1, \mathbf{p}_4, \mathbf{p}_3, \mathbf{p}_2$, respectively. (Right) Estimated $\hat{\mathbf{c}}, \hat{r}$, real \mathbf{c}, r , and angle β_i between two vehicles at \mathbf{p}_{i+1} and \mathbf{p}_i .

$$\dot{\mathbf{p}}_i = \mathbf{u}_i, \quad i \in [1, \dots, n], \quad (3.2)$$

where \mathbf{p}_i is a vector that contains the position $\mathbf{p}_i = [x_i, y_i]^\top \in \mathbb{R}^2$ and $\mathbf{u}_i \in \mathbb{R}^2$ is the control input.

To avoid the AUVs concentrating in a region, in which case they may lose information on other fronts, we would like to space them equally along the defined circle. Therefore, we define the counterclockwise angle between the vector $\mathbf{p}_i - \hat{\mathbf{c}}$ and $\mathbf{p}_{i+1} - \hat{\mathbf{c}}$ as β_i for $i = 1, \dots, n-1$, and the angle between $\mathbf{p}_n - \hat{\mathbf{c}}$ and $\mathbf{p}_1 - \hat{\mathbf{c}}$ as β_n ,

$$\begin{aligned} \beta_i &= \angle(\mathbf{p}_{i+1} - \hat{\mathbf{c}}, \mathbf{p}_i - \hat{\mathbf{c}}), & i = 1, \dots, n-1 \\ \beta_n &= \angle(\mathbf{p}_1 - \hat{\mathbf{c}}, \mathbf{p}_n - \hat{\mathbf{c}}). \end{aligned} \quad (3.3)$$

Then, it holds that

$$\beta_i(0) \geq 0, \quad \text{and} \quad \sum_{i=1}^n \beta_i(0) = 2\pi. \quad (3.4)$$

This is represented in the left scheme of Fig. 3.2.

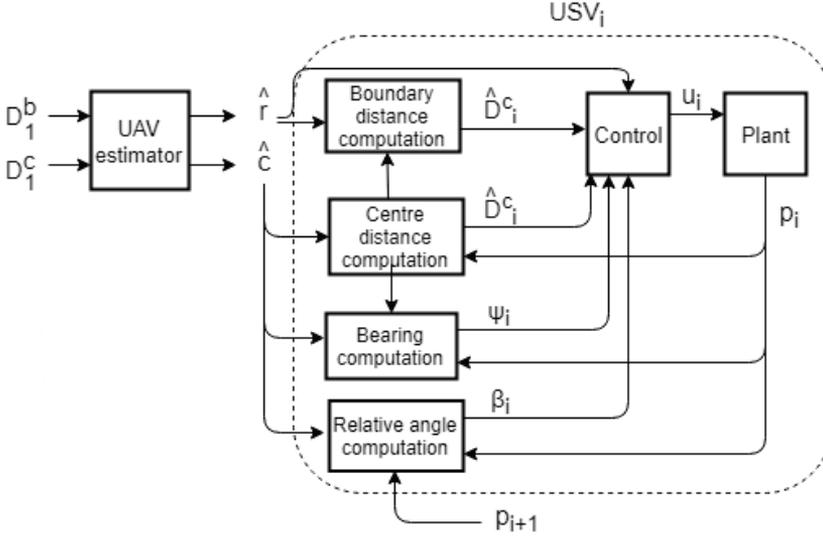


Figure 3.3: The UAV estimates the center and radius of the target based on its distance measurements and shares it with all AUVs. Each AUV i calculates its control protocol.

Note that the ℓ_2 -norm is denoted simply as $\|\cdot\|$ without a subscript. Now, we can define the distance from the UAV to the center and the boundary of the target circle as

$$\begin{aligned} D_1^c &= \|\mathbf{c} - \mathbf{p}_1\| \\ D_1^b &= |r - D_1^c|, \end{aligned} \quad (3.5)$$

respectively. Note that the UAV senses the distances to the target and then calculates the target estimates. This UAV operation is represented in the left part of Fig. 3.3.

After obtaining the target estimates, each AUV i would be able to calculate its own distances \hat{D}_i^c and \hat{D}_i^b

$$\begin{aligned} \hat{D}_i^c &= \|\hat{\mathbf{c}} - \mathbf{p}_i\| \\ \hat{D}_i^b &= |\hat{r} - \hat{D}_i^c|, \end{aligned} \quad (3.6)$$

as represented in the right scheme of Fig. 3.2. We summarise each AUVs' computation scheme in the right part of Fig. 3.3.

3.1.2 Problem statement

Definition 3.1 (Circumnavigation) *When the target is stationary, i.e., \mathbf{c} and r are constant, circumnavigation is achieved if the AUVs*

1. *move in a counterclockwise direction on the boundary of the target, and*
2. *are equally distributed along the circle, i.e., $\beta_i = \frac{2\pi}{n}$.*

More specifically, we say that the circumnavigation is achieved asymptotically if the previous criteria are satisfied for $t \rightarrow \infty$.

For the case with time-varying target, we assume that $\|\dot{\mathbf{c}}\| \leq \varepsilon_1$ and $|\dot{r}| \leq \varepsilon_2$ for some positive constants ε_1 and ε_2 .

Now, we are ready to pose the problem of interest that will be solved in the following sections.

Problem 1. Design a UAV estimator for $\mathbf{c}(t)$ and $r(t)$ when distance measures (3.6) are available to the UAV, and design the control inputs \mathbf{u}_i for the AUVs such that for some positive $\varepsilon_1, \varepsilon_2$,

$$\|\dot{\mathbf{c}}\| \leq \varepsilon_1, \quad (3.7)$$

$$|\dot{r}| \leq \varepsilon_2, \quad (3.8)$$

there exist positive K_1, K_2 , and K_3 satisfying

$$\limsup_{t \rightarrow \infty} \|\hat{\mathbf{c}}(t) - \mathbf{c}(t)\| \leq K_1 \varepsilon_1, \quad (3.9)$$

$$\limsup_{t \rightarrow \infty} |\hat{r}(t) - r(t)| \leq K_2 \varepsilon_2, \quad (3.10)$$

$$\limsup_{t \rightarrow \infty} |\hat{D}_i^c(t) - \hat{r}(t)| \leq K_3 \varepsilon_2, \quad (3.11)$$

$$\lim_{t \rightarrow \infty} \beta_i(t) = \frac{2\pi}{n}. \quad (3.12)$$

3.2 Adaptive Estimation and Control

This section proposes an estimation and control mechanism for Problem 1. We consider n AUVs at positions \mathbf{p}_i and one UAV capable of measuring its distance D_i^b to the target boundary and its distance D_i^c to the target center.

Then, it should estimate (\mathbf{c}, r) from its distance measures, i.e., D_i^b and D_i^c , and share the information with the AUVs. Each AUV calculates its desired velocity, taking into account its angle β_i to the next AUV and its distance to the target center and boundary, obtained with the estimates of the target.

3.2.1 Adaptive estimation

This subsection relates to the protocol followed by the UAV for estimation. Recalling Fig. 3.3, we will construct the UAV estimator block. Motivated by [54], we propose the following adaptive estimation of the radius r of the target using the UAV A_1 in position \mathbf{p}_1 . Observe that

$$\frac{d}{dt}(D_1^b)^2 = 2(\dot{r} - \dot{D}_1^c)(r - D_1^c). \quad (3.13)$$

Assume the estimate of r is denoted as \hat{r} , we have

$$\frac{1}{2}\left(\frac{d}{dt}(D_1^b)^2 - \frac{d}{dt}(D_1^c)^2\right) + \dot{D}_1^c \hat{r} = \dot{D}_1^c(\hat{r} - r) + \dot{r}(r - D_1^c). \quad (3.14)$$

Then for some positive constant γ , the dynamic

$$\dot{\hat{r}} = -\gamma \dot{D}_1^c \left[\frac{1}{2}\left(\frac{d}{dt}(D_1^b)^2 - \frac{d}{dt}(D_1^c)^2\right) + \dot{D}_1^c \hat{r} \right] \quad (3.15)$$

can estimate the variable r under the persistent excitation condition on \dot{D}_1^c . Persistent excitation is key in establishing parameter convergence in adaptive identification [112], [113].

Definition 3.2 (*Continuous time persistent excitation condition*) [113] *The function $f \in \mathcal{L}_e^2(\mathbb{R}^n)$ is said to be persistently exciting (p.e.) if there exist positive constants ε_1, T such that for all $\tau > 0$,*

$$\int_{\tau}^{T+\tau} f(t)f(t)^\top dt > \varepsilon_1 I_n.$$

T will be termed an excitation period of f .

Then, in this case

$$\frac{d}{dt}(\hat{r} - r) = -\gamma(\dot{D}_1^c)^2(\hat{r} - r) - \vartheta_{\hat{r}}, \quad (3.16)$$

where $\vartheta_{\dot{r}} = \dot{r}(\gamma\dot{D}_1^c(r - D_1^c) + 1)$ is bounded by $M_1\varepsilon_2$. Indeed, all its elements are bounded by $M_1 > 0$ and recall that $|\dot{r}| \leq \varepsilon_2$. Note that $r - D_1^c$ is bounded because r and D_1^c are also bounded. Furthermore, as it will be clear soon, $\vartheta_{\dot{r}}$ can be replaced by $\vartheta_{\dot{r}} = \dot{r}(\gamma V(r - D_1^c) + 1)$ using equations (3.21) and (3.22), where V is the bounded estimate of D_1^c .

However, the implementation of (3.15) needs the derivative of D_1^b and D_1^c , which is not desired. It would require explicit differentiation of measured signals with accompanying noise amplification. Thus, for some positive constant α , we adopt the state variable filtering and then design the estimator as follows

$$\dot{z}_1(t) = -\alpha z_1(t) + \frac{1}{2}(D_1^b(t))^2 \quad (3.17)$$

$$\eta(t) = \dot{z}_1(t) \quad (3.18)$$

$$\dot{z}_2(t) = -\alpha z_2(t) + \frac{1}{2}(D_1^c(t))^2 \quad (3.19)$$

$$m(t) = \dot{z}_2(t) \quad (3.20)$$

$$\dot{z}_3(t) = -\alpha z_3(t) + D_1^c(t) \quad (3.21)$$

$$V(t) = \dot{z}_3(t) \quad (3.22)$$

with initial conditions $z_1(0) = z_2(0) = z_3(0) = 0$. Now, together with the above dynamics, the estimator for r is given as

$$\dot{\hat{r}} = -\gamma V[\eta - m + V\hat{r}]. \quad (3.23)$$

We are interested in obtaining \mathbf{c} from the measurements D_1^c and D_1^b . Thus, we must again use adaptive estimation for the target's center \mathbf{c} .

Observe that

$$\frac{d}{dt}(D_1^c)^2 = 2(\dot{\mathbf{p}}_1 - \dot{\mathbf{c}})^\top(\mathbf{p}_1 - \mathbf{c}). \quad (3.24)$$

Assume the estimation of \mathbf{c} is denoted as $\hat{\mathbf{c}}$, we have

$$\frac{1}{2}\left(\frac{d}{dt}(D_1^c)^2 - \frac{d}{dt}\|\mathbf{p}_1\|^2\right) + \dot{\mathbf{p}}_1^\top \hat{\mathbf{c}} = \dot{\mathbf{p}}_1^\top(\hat{\mathbf{c}} - \mathbf{c}) + \hat{\mathbf{c}}^\top(\mathbf{c} - \mathbf{p}_1). \quad (3.25)$$

Then the dynamic

$$\dot{\hat{\mathbf{c}}} = -\gamma\dot{\mathbf{p}}_1\left[\frac{1}{2}\left(\frac{d}{dt}(D_1^c)^2 - \frac{d}{dt}\|\mathbf{p}_1\|^2\right) + \dot{\mathbf{p}}_1^\top \hat{\mathbf{c}}\right] \quad (3.26)$$

can estimate the parameter \mathbf{c} under some persistent excitation condition on $\dot{\mathbf{p}}_1$. Indeed, in this case

$$\frac{d}{dt}(\hat{\mathbf{c}} - \mathbf{c}) = -\gamma\|\dot{\mathbf{p}}_1\|^2(\hat{\mathbf{c}} - \mathbf{c}) - \vartheta_{\hat{\mathbf{c}}}, \quad (3.27)$$

where $\vartheta_{\hat{\mathbf{c}}} = \gamma\dot{\mathbf{c}}^\top \dot{\mathbf{p}}_1(\mathbf{c} - \mathbf{p}_1) + \dot{\mathbf{c}}$ is bounded by $M_2\varepsilon_1$. Indeed all its elements are bounded by $M_2 > 0$ and recall that $|\dot{\mathbf{c}}| \leq \varepsilon_1$. Note that $\mathbf{c} - \mathbf{p}_1$ is bounded because \mathbf{c} and \mathbf{p}_1 are within a finite map. Furthermore, as it will be clear soon, $\vartheta_{\hat{\mathbf{c}}}$ can be replaced by $\vartheta_{\hat{\mathbf{c}}} = \gamma\dot{\mathbf{c}}^\top V_2(\mathbf{c} - \mathbf{p}_1) + \dot{\mathbf{c}}$ using equations (3.30)-(3.31), where V_2 is the estimate of \mathbf{p}_1 and it is bounded.

However, the implementation of (3.26) needs the derivative of \mathbf{p}_1 and D_1^c , which is not desired. Therefore we use the previously defined equation (3.20) for D_1^c and redefine it as $\eta_2(t) = \dot{z}_2(t)$ and add the following filter

$$\dot{z}_4(t) = -\alpha z_4(t) + \frac{1}{2}\mathbf{p}_1(t)\mathbf{p}_1^\top(t) \quad (3.28)$$

$$m_2(t) = \dot{z}_4(t) \quad (3.29)$$

$$\dot{z}_5(t) = -\alpha z_5(t) + \mathbf{p}_1(t) \quad (3.30)$$

$$V_2(t) = \dot{z}_5(t) \quad (3.31)$$

with initial conditions $z_4(0) = z_5(0) = 0$. After updating (3.26) with the above dynamics, the estimator for \mathbf{c} is given as

$$\dot{\hat{\mathbf{c}}} = -\gamma V_2[\eta_2 - m_2 + V_2^\top \hat{\mathbf{c}}]. \quad (3.32)$$

3.2.2 Control Algorithm

This subsection relates to the protocol followed by the AUVs for control. Recalling Fig. 3.3, we will construct the AUV control block. Therefore, we want to obtain the desired control input \mathbf{u}_i using the previously measured and estimated variables.

The total velocity of each AUV comprises two sub-tasks: approaching the target and circumnavigating it. Therefore we define the direction of each AUV towards the estimated center of the target as the bearing ψ_i ,

$$\psi_i = \frac{\hat{\mathbf{c}} - \mathbf{p}_i}{\hat{D}_i^c} = \frac{\hat{\mathbf{c}} - \mathbf{p}_i}{\|\hat{\mathbf{c}} - \mathbf{p}_i\|}. \quad (3.33)$$

The first sub-task is related to the bearing ψ_i , and the second is related to its perpendicular, $E\psi_i$. We define a rotation matrix E as

$$E = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}. \quad (3.34)$$

Then, let us first consider the control law \mathbf{u}_i where δ is a parameter to be defined.

$$\mathbf{u}_i = \dot{\hat{\mathbf{c}}} + ((\hat{D}_i^c - \hat{r}) - \frac{1}{\delta}\dot{\hat{r}})\psi_i + \beta_i \hat{D}_i^c E\psi_i. \quad (3.35)$$

The control actuation of an AUV is limited; therefore, we have to make sure that the implemented control is within the actuation bounds, and so we introduce

$$\mathbf{U}_i = \delta \mathbf{u}_i \quad (3.36)$$

where δ is the same as before. For a specific \mathbf{u}_i , it is possible to have \mathbf{U}_i within some specified bounds.

3.3 Convergence Results

In this section, we prove that the estimator and control algorithm proposed in the previous section converge to the desired behavior.

Theorem 3.1 *The initial condition satisfies $\hat{D}_i^c(0) > \hat{r}(0) > 0$. Suppose $\dot{\mathbf{p}}_1(t)$ and $\dot{D}_1^c(t)$ are p.e., $\|\dot{\hat{\mathbf{c}}}\| \leq \varepsilon_1$, and $|\dot{\hat{r}}| \leq \varepsilon_2$. Consider the system (3.35) with the control protocol (3.36), and the initialization satisfying $\|\mathbf{p}_i(0) - \hat{\mathbf{c}}(0)\| > 0$, then there exists K_1 , K_2 , and K_3 such that circumnavigation of the moving circle with equally spaced AUVs can be achieved asymptotically up to a bounded error, i.e.*

$$\limsup_{t \rightarrow \infty} \|\hat{\mathbf{c}}(t) - \mathbf{c}(t)\| \leq K_1 \varepsilon_1, \quad (3.37)$$

$$\limsup_{t \rightarrow \infty} |\hat{r}(t) - r(t)| \leq K_2 \varepsilon_2, \quad (3.38)$$

$$\limsup_{t \rightarrow \infty} |\hat{D}_i^c(t) - \hat{r}(t)| \leq K_3 \varepsilon_2, \quad (3.39)$$

$$\lim_{t \rightarrow \infty} \beta_i(t) = \frac{2\pi}{n}. \quad (3.40)$$

Proof 3.1 *The proof is divided into four parts. In the first part, we prove that (3.37) and (3.38) hold. In the second part, we prove that the estimated distance \hat{D}_i^c converges to the estimated radius \hat{r} , or in other words, that (3.39) holds. In the third part, we prove that the singularity of the bearing ψ_i is avoided. In the last part, we show that the angle between the AUVs will converge to the average consensus for n AUVs, $\beta_i = \frac{2\pi}{n}$, meaning (3.40) holds.*

1. *Firstly, we prove that (3.37) and (3.38) hold. The proof for boundedness of the center (3.37) can be found on [54], Proposition 7.1. However, the proof for the boundedness of the ratio needs to be derived in this chapter. Then, we have that*

$$\begin{aligned}
 \dot{\tilde{r}} &= \dot{\hat{r}} = -\gamma V[\eta - m + V\hat{r}] \\
 &= -\gamma V[\eta - m + V(\tilde{r} + r)] \\
 &= -\gamma V^2\tilde{r} - \gamma V[\eta - m + Vr] \\
 &= -\gamma V^2\tilde{r} + G
 \end{aligned} \tag{3.41}$$

where $G = -\gamma V[\eta - m + Vr]$. We know that $|G| \leq k_1\epsilon_2$ for some $k_1, \epsilon_2 \geq 0$ because V is bounded and that $|\eta - m + Vr| < k_2$ we can prove that for a Lyapunov function $W_r = \frac{1}{2}\tilde{r}^2$ we get

$$\begin{aligned}
 \dot{W}_r &= \tilde{r}\dot{\tilde{r}} = \tilde{r}(-\gamma V^2\tilde{r} + G) \\
 &= -\gamma V^2\tilde{r}^2 + \tilde{r}G \\
 &\leq -\gamma V^2\tilde{r}^2 + k_1\epsilon_2\tilde{r}
 \end{aligned} \tag{3.42}$$

then we get that for $\dot{W}_r \leq 0$ to hold, $-\gamma V^2\tilde{r}^2 + k_1\epsilon_2\tilde{r} \leq 0$ must hold. So, we have that when $\tilde{r} \geq \frac{k_1\epsilon_2}{\gamma V^2}$ or $\tilde{r} \leq -\frac{k_1\epsilon_2}{\gamma V^2}$, $\dot{W}_r \leq 0$ so that $|\tilde{r}|$ is within $\pm \frac{k_1\epsilon_2}{\gamma V^2}$. This error \tilde{r} is then proved to converge asymptotically to a ball since \dot{D}_1^c is p.e..

2. *We prove that all AUVs reach the estimate of the boundary of the moving circles asymptotically, i.e., $\lim_{t \rightarrow \infty} \|\mathbf{p}_i(t) - \hat{\mathbf{c}}(t)\| = \lim_{t \rightarrow \infty} \hat{D}_i^c(t) = \hat{r}(t)$, so (3.39) holds.*

Consider the function $W_i(t) := \hat{D}_i^c(t) - \hat{r}(t)$ whose time derivative for $t \in [0, +\infty)$ is given as

$$\dot{W}_i = \frac{(\hat{\mathbf{c}} - \mathbf{p}_i)^\top (\dot{\hat{\mathbf{c}}} - \dot{\mathbf{p}}_i)}{\hat{D}_i^c} - \dot{\hat{r}}$$

$$\begin{aligned}
&= -\frac{(\hat{\mathbf{c}} - \mathbf{p}_i)^\top}{\hat{D}_i^c} \delta((\hat{D}_i^c - \hat{r} - \dot{\hat{r}})\psi_i + \beta_i \hat{D}_i^c E \psi_i) - \dot{\hat{r}} \\
&= -\frac{(\hat{\mathbf{c}} - \mathbf{p}_i)^\top}{\hat{D}_i^c} \psi_i \delta(\hat{D}_i^c - \hat{r} - \dot{\hat{r}}) - \frac{(\mathbf{c} - \mathbf{p}_i)^\top}{\hat{D}_i^c} E \psi_i \delta \beta_i \hat{D}_i^c - \dot{\hat{r}} \\
&= -\delta(\hat{D}_i^c - \hat{r} - \dot{\hat{r}}) - \dot{\hat{r}} \\
&= -\delta W_i.
\end{aligned}$$

Hence for $t \in [0, +\infty)$, we have $\hat{D}_i^c(t) = \delta W_i(0)e^{-t} + \hat{r}(t)$ which implies W_i is converging to zero exponentially.

3. Finally, we show that the angle between the AUVs will converge to the average consensus for n AUVs, $\beta_i = \frac{2\pi}{n}$, so (3.40) holds.

Firstly, note that we can write an angle between two vectors $\beta_i = \angle(v_2, v_1)$ as

$$\beta_i = 2 \operatorname{atan2}((v_1 \times v_2) \cdot z, \|v_1\| \|v_2\| + v_1 \cdot v_2) \quad (3.43)$$

and its derivative as

$$\dot{\beta}_i = \frac{\hat{v}_1 \times z}{\|v_1\|} \dot{v}_1 - \frac{\hat{v}_2 \times z}{\|v_2\|} \dot{v}_2 \quad (3.44)$$

where $z = \frac{v_1 \times v_2}{\|v_1 \times v_2\|}$, $\hat{v}_i = \frac{v_i}{\|v_i\|}$, $i = 1, 2$.

Then, for $v_1 = \mathbf{p}_i - \hat{\mathbf{c}}$ and $v_2 = \mathbf{p}_{i+1} - \hat{\mathbf{c}}$ we get

$$\begin{aligned}
\dot{\beta}_i &= \frac{\hat{v}_1 \times z}{\|v_1\|} \dot{v}_1 - \frac{\hat{v}_2 \times z}{\|v_2\|} \dot{v}_2 \\
&= \frac{\hat{v}_1 \times z}{\|v_1\|} \delta((\hat{D}_i^c - \hat{r} - \dot{\hat{r}})\psi_i + \beta_i \hat{D}_i^c E \psi_i) \\
&\quad - \frac{\hat{v}_2 \times z}{\|v_2\|} \delta((\hat{D}_{i+1}^c - \hat{r} - \dot{\hat{r}})\psi_{i+1} + \beta_{i+1} \hat{D}_{i+1}^c E \psi_{i+1}) \\
&= -\frac{1}{\|v_1\|} \beta_i + \frac{1}{\|v_2\|} \beta_{i+1} \\
&= \delta(-\beta_i + \beta_{i+1}), \quad i = 1, \dots, n-1 \\
\dot{\beta}_n &= \delta(-\beta_n + \beta_1).
\end{aligned}$$

which can be written in a compact form as following

$$\dot{\beta} = -\delta B^\top \beta \quad (3.45)$$

where B is the incidence matrix of the directed ring graph from v_1 to v_n .

First, we note that the system (3.45) is positive (see e.g., [114]), i.e., $\beta_i(t) \geq 0$ if $\beta_i(0) \geq 0$ for all $t \geq 0$ and $i \in \mathcal{I}$. This proves the positions of the AUVs are not interchangeable.

Second, noticing that B^\top is the (in-degree) Laplacian of the directed ring graph which is strongly connected, then by Theorem 6 in [115], β converges to consensus $\frac{2\pi}{n} \mathbf{1}$.

Note how the AUV A_i will necessarily maintain its relative position \mathbf{p}_i throughout the circumnavigation mission. This proves that AUV A_i is always in position \mathbf{p}_i . We proved both convergences of the angle to the average consensus for n AUVs and the convergence of these vehicles towards the boundary of the target up to a given bound. Therefore, we guarantee collision avoidance.

Recall Definition 1 on persistent excitation. This means that for the persistently exciting condition to apply, the AUV must move in a trajectory not confined to a straight line in the 2D space. As referred in [54], the AUV cannot simply head straight toward the target but must execute a richer class of motion.

Note that the p.e. condition is assumed for Theorem 1. and not proved. However, in the results section, we will verify if the p.e. assumptions are valid for our simulations within the simulation time.

3.4 Simulation Results

In this section, we present simulations for the protocol designed in section 3.3. We use the derived method for estimation of the target (3.23) and (3.32) and the controlling protocol for the AUVs (3.36). For this section, we discretize the whole algorithm to be able to use it computationally. The first subsection considers the persistent excitation condition, and the second subsection analyses what happens when this condition is not verified.

3.4.1 Simulations with p.e. guarantees

In this subsection, we simulate a moving target with initial position $(x[0], y[0]) = (25, 25)$, radius $r[0] = 10$, and dynamic according to

$$\begin{aligned} x[t+1] &= x[t] + \alpha_1[t] + 0.5 \\ y[t+1] &= y[t] + \alpha_2[t] + 0.5 \\ r[t+1] &= r[t] + \alpha_3[t] \end{aligned} \tag{3.46}$$

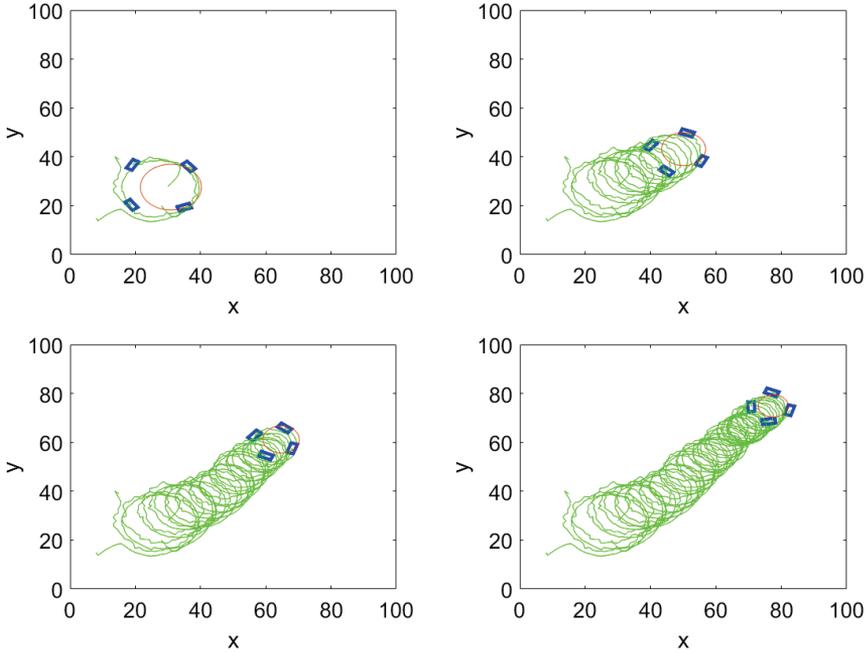


Figure 3.4: Time-lapse of four AUVs (blue rectangles) circumnavigating a moving target (red) with a representation of their paths (green)

However, we simulate that the UAV will provide an initial noisy estimate of $(\hat{x}[0], \hat{y}[0]) = (25, 25)$, radius $\hat{r}[0] = 20$. Note that at time $t = 0$, the radius estimate is double the actual radius. Here, $\alpha_i[t]$ is a random scalar drawn from the uniform distribution within the $[-0.5, 0.5]$ interval for $i = 1, 2, 3$. For this generated target, we got the following results. The AUVs circumnavigate the moving target in Fig. 3.4. This gives us a more practical idea of how the AUVs behave in their target-tracking mission.

Fig. 3.5 shows various plots that analyze the system's behavior. We compare the real and estimated targets in the first and second rows. Note that the estimate of the center $\hat{c}(\hat{x}, \hat{y})$ has an estimation error of up to 2 units. Also, note that the radius estimate \hat{r} comprises two instances. First, the initial estimate provided by the UAV was very noisy, so we can see the estimate converging rapidly to a more accurate estimation. We see an estimation error of up to 2 units in the second. On the third row left column, we can see the distance D_i^b of each target to the boundary of the target - the perfect tracking would result in a distance D_i^b of 0

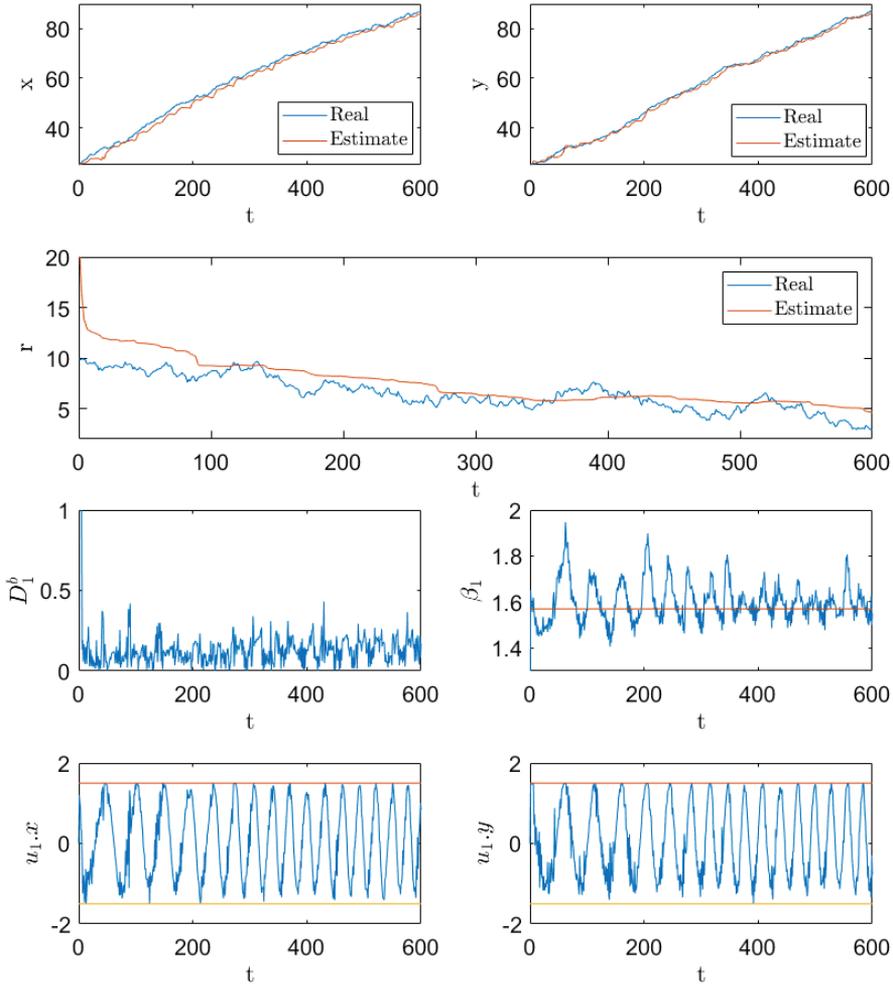


Figure 3.5: First and second row: real and estimated target's center $\mathbf{c} : x, y$ and radius r . Third row: tracking error of AUV A_1 , D_1^b , and angle β_1 . Fourth row: control input of AUV A_1 , $\mathbf{u}_1 : x, y$

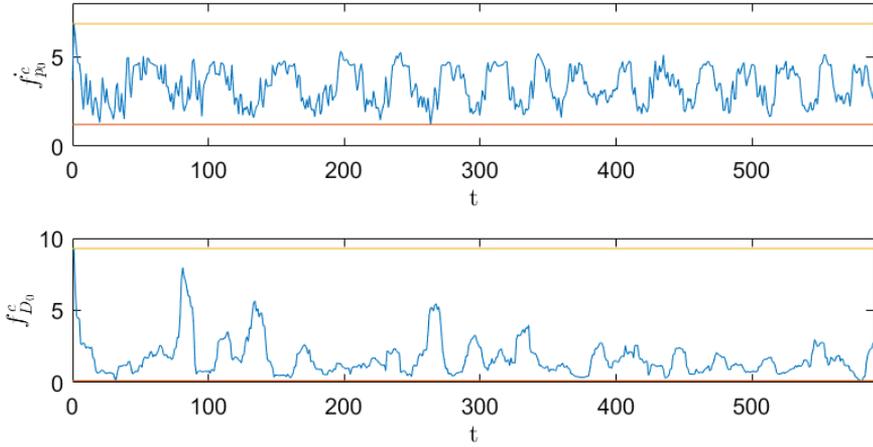


Figure 3.6: First row: $f_{\dot{\mathbf{p}}_1}[t]$ is bounded by strictly positive bounds. Second row: $f_{\dot{D}_1^c}[t]$ is bounded by strictly positive bounds.

for all AUVs for every time step. Here, we have an error of up to 0.5 units, except for the beginning, where the error can reach 10 units. This is merely because the AUVs are initially far from the target.

On the third row, right column, we have the angle between AUV A_1 and A_2 , β_1 . Having 4 AUVs, the perfect tracking would result in $2\pi/4 = \pi/2 \approx 1.57$ for all AUVs for every time step. We can see this reference as the red line in the plot, so we see that, for AUV A_1 , the error is up to 0.2 radians. Finally, on the fourth row, we have the control input \mathbf{u}_1 of AUV A_1 , with parameters x and y . Recall Remark. 2, where we stated that, for a practical implementation, there should be a maximum velocity u_{max} . For this case study, we defined that $u_{max} = 1.5$, and we plotted this limit in red. Note how the control input stays within the limit values 1.5 and -1.5.

Since we considered as an assumption that $\dot{\mathbf{p}}_1[t]$ and $\dot{D}_1^c[t]$ are p.e., we now evaluate whether this is the case for this simulation example. According to [116],

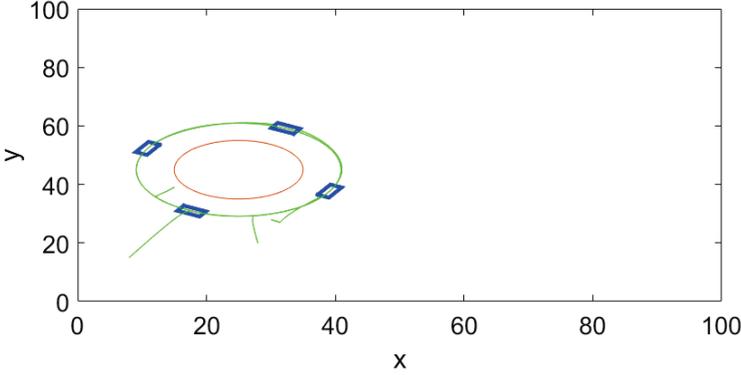


Figure 3.7: Four AUVs (blue rectangles) circumnavigating a moving target (red) with a representation of their paths (green).

we can adapt Definition 3.2. to the discrete-time case so we obtain the functions

$$\begin{aligned}
 f_{\dot{\mathbf{p}}_1}[t] &= \sum_{k=t}^{t+m} \dot{\mathbf{p}}_1^\top[k] \dot{\mathbf{p}}_1[k], \\
 f_{\dot{D}_1^c}[t] &= \sum_{k=t}^{t+m} \dot{D}_1^c[k]^2,
 \end{aligned} \tag{3.47}$$

which must fulfil $\rho_2 > f_{\dot{\mathbf{p}}_1}[t] > \rho_1$ and $\rho_4 > f_{\dot{D}_1^c}[t] > \rho_3$ for positive ρ_i .

As seen in Fig. 3.6., these conditions are fulfilled for $\rho_1 = 1.1026$, $\rho_2 = 6.8371$, $\rho_3 = 0.2443$, and $\rho_4 = 8.8497$. Then, for these results in this simulation time, the p.e. conditions hold.

3.4.2 Simulations without p.e. guarantees

In this subsection, we simulate a static target with position $(x[0], y[0]) = (25, 25)$ and radius $r[0] = 10$ for all time t . As in the previous subsection, we simulate that the UAV provides an estimate of $(\hat{x}[0], \hat{y}[0]) = (25, 25)$ and radius $\hat{r}[0] = 20$. This means that the estimates for the center will not have any initial error, and the estimate for the radius will have an error of $e_r[0] = \hat{r}[0] - r[0] = 20 - 10 = 10$.

As seen in Fig. 3.7, the position estimation seems correct, but the radius estimation appears wrong.

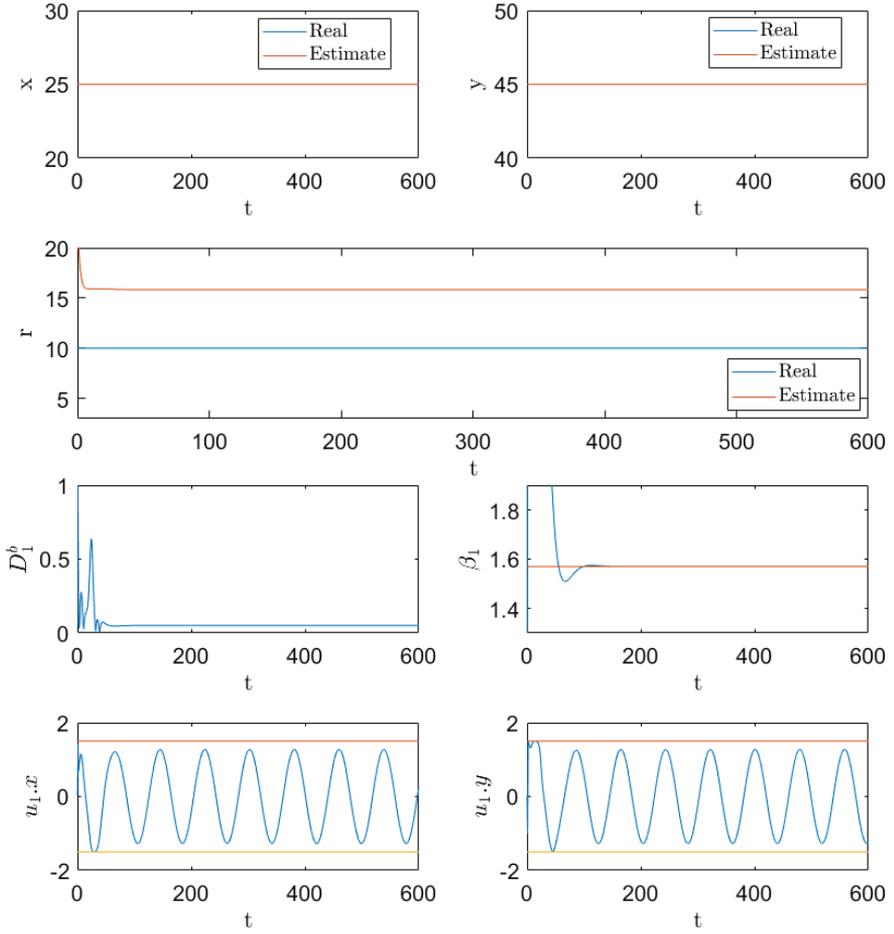


Figure 3.8: First and second row: real and estimated target's center $\mathbf{c} : x, y$ and radius r . Third row: tracking error of AUV A_1 , D_1^b , and angle β_1 . Fourth row: control input of AUV A_1 , $\mathbf{u}_1 : x, y$

From the first row, Fig. 3.8, we can see how the estimates for the center $\mathbf{c}(x, y)$ are correct for all the simulation time. However, from the second row, we can see a steady-state error for the estimation of r . Recall that the estimators derived in 3.3 for \mathbf{c} and r rely on the p.e. conditions for $\dot{\mathbf{p}}_1$ and \dot{D}_1^c , respectively. Then, it seems that the p.e. condition on \dot{D}_1^c does not hold; therefore, the estimation of r does not converge to the actual r .

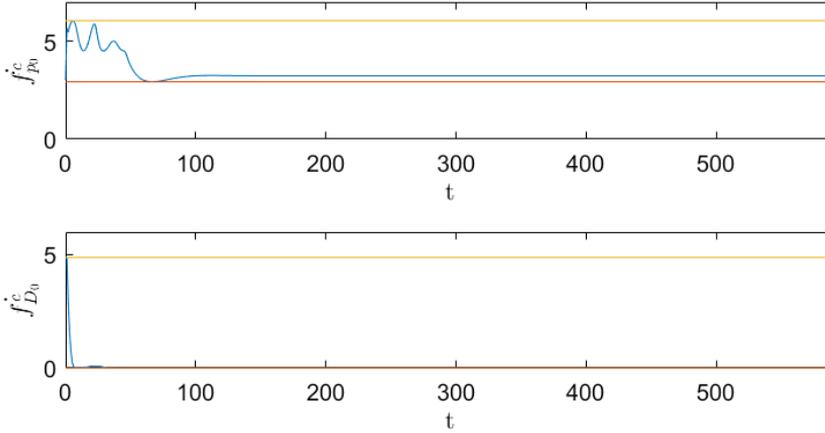


Figure 3.9: First row: $f_{\hat{p}_1}[t]$ is bounded by strictly positive bounds. Second row: $f_{D_1^c}[t]$ is bounded by a strictly positive bound and zero.

From Fig. 3.9, we can conclude that, for this simulation time, even though the p.e. condition is verified for \hat{p}_1 , it is not verified for D_1^c since for some time t the minimum bound is not strictly positive.

3.5 Summary

In this chapter, we considered the problem of multi-vehicle target tracking. We assumed the target was an irregular dynamic shape approximated by a circle with a moving center and varying radius. We defined the problem mathematically by introducing relevant variables and equations that relate them. We also defined the measurements available to each agent and the estimation and circumnavigation objectives. We created an adaptive estimation and control algorithm and proved their mathematical convergence to a bound according to our objectives. We presented two simulation results: one to analyze convergence performance and the other to represent the need for the persistence of excitation when applying the developed algorithms.

Chapter 4

Decentralized Target Estimation for Multi-agent Systems

In this chapter, we define a different setup than in Chapter 3 and, therefore, a different algorithm to solve the multi-vehicle target tracking problem. We try to decentralize the measurement step by having all vehicles capable of collecting and sharing data to achieve target estimation. As in the previous chapter, we consider that the target is an irregular dynamic shape approximated by a circle with a moving center and varying radius. In this chapter, the AUV system is composed of n AUVs, all measuring their distance to the boundary of the target. The AUV system must circumnavigate the boundary of the target while forming a regular polygon. We also present an extension of the results to a more general scenario with irregular, non-circular moving shapes. Each vehicle is equipped with a sensor that indicates its distance to the target's boundary, including whether it is inside or outside of the target. The first step is the estimation of the parameters of the algal bloom curvature in each region of the shape, that is, its center and radius for every time instance and every vehicle. The second step is to design a control law for the vehicles to circumnavigate the shape. We perform a numerical study of the convergence of vehicles to the target's boundary.

This chapter is organized as follows. In Section 4.1, we describe the system mathematically and formulate the problem to be solved in the following section. In Section 4.2, we design two algorithms: one for finding the optimal circle target using the UAV's measurements and another for the control protocol to be applied by all AUVs in their navigation. In Section 4.3, the convergence of both algorithms to the desired state is proved up to a limit bound. A simulated

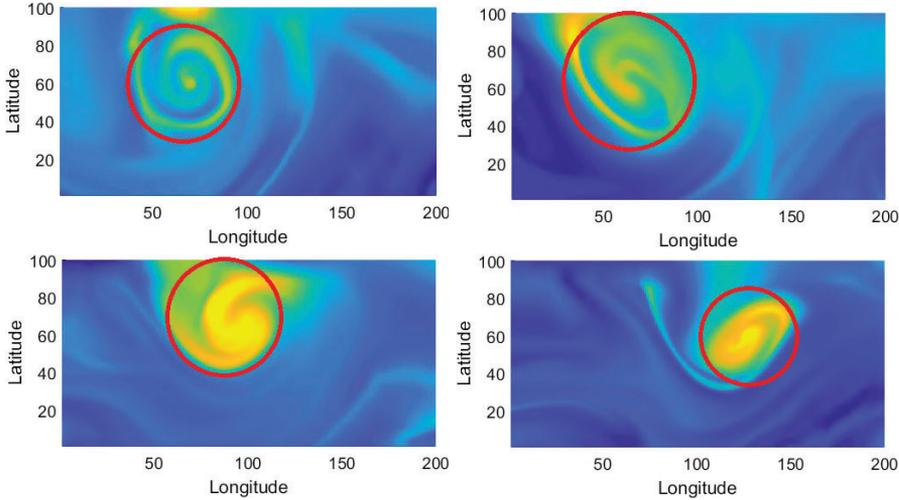


Figure 4.1: Time-lapse of the algal bloom progression. There is approximately half a day between each image. Warm colors (yellow, orange, green) indicate a high density of algal blooms, and cold colors (blues) indicate a low density of algal blooms.

example using SINMOD data is provided to verify the performance of the algorithms designed follows in Section 4.4. In Section 4.5, we extend the protocol from the previous chapter to arc estimation. Two sets of simulations illustrating the performance of the extended algorithm are given in Section 4.6. Concluding remarks come in Section 4.7.

4.1 Problem Formulation

We consider the problem of tracking a circular shape using a multi-robot system and a satellite. This shape may be very irregular and unstable over time. We assume a circle can approximate the shape. An initial image of the algal bloom confirms such an assumption, as seen in Fig. 4.1, and then we can decide to use our algorithm to deploy the agents.

We define this circle as

$$(\mathbf{c}, r) \in \mathbb{R}^3, \quad (4.1)$$

where $\mathbf{c} = (x, y)$ and r are the circle's center and radius, respectively. After confirming the algal bloom is close enough to a circle, we can estimate it by

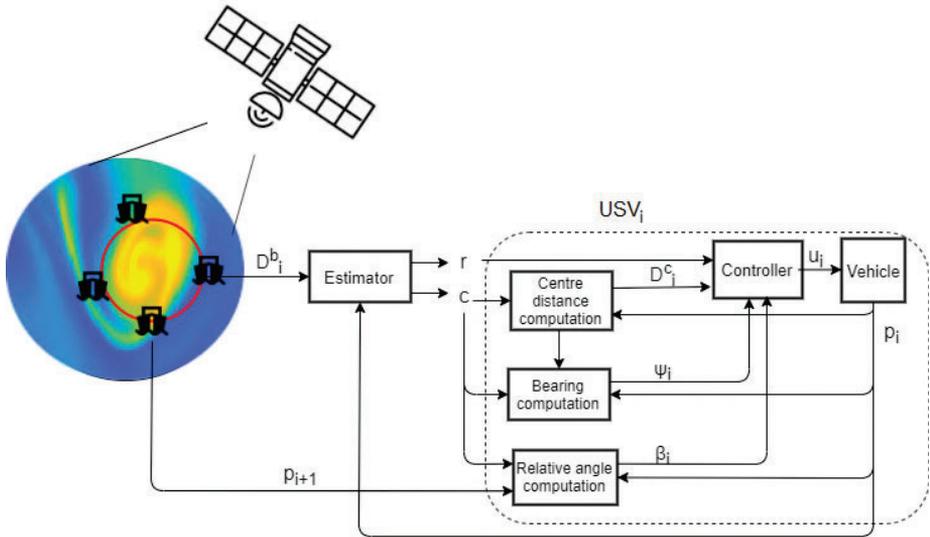


Figure 4.2: Each AUV estimates the center and radius of the target based on its distance measurements and shares it with all AUVs. Each AUV i calculates its control protocol.

our robot’s measurements. This estimate is represented as $(\hat{c}, \hat{r}) \in \mathbb{R}^3$. Note that using a circle shape does not compromise the generality of the algorithm. Instead, it guarantees a smooth circumnavigation for any irregular shape close to a circle. A similar algorithm can be done for shapes approximated by ellipsoids, but we present a simpler case with circle shapes for notation simplicity.

To solve this tracking problem, we use two types of tools: a satellite and a system of robots. Depending on the weather, the satellite obtains data from the target as an image. Then, using image processing, it calculates the possible initial center and radius of such a circle and shares it with the robots so they can move toward the target and initiate circumnavigation. Thus, the satellite would provide initial estimates $\hat{c}(0) = (\hat{x}(0), \hat{y}(0))$ and $\hat{r}(0)$. The robots constantly measure their distances to the target’s boundary and whether they are inside or outside the target and share it with the other robots. Each robot has access to its GPS position and the robot’s position in front of it. This communication scheme is represented in Fig. 4.2. Values such as β_i , p_i , and D_i^b will be soon properly defined.

The system of robots will jointly circumnavigate the target and provide real-time information on different fronts. We define we have n agents and, using the satellite information, they are initialized at positions $\mathbf{p}_i(0)$, $i \in [1, \dots, n]$, which are outside of the shape and form a counterclockwise directed ring on the surface. The kinematic of the agents is of the form

$$\dot{\mathbf{p}}_i = \mathbf{u}_i, \quad i \in [1, \dots, n], \quad (4.2)$$

where \mathbf{p}_i is a vector that contains the position $\mathbf{p}_i = [x_i, y_i]^\top \in \mathbb{R}^2$ and $\mathbf{u}_i \in \mathbb{R}^2$ is the control input.

To avoid the agents concentrating in a region, in which case they may lose information on other fronts, we would like to distribute the agents equally along the defined circle. Therefore, we introduce two more parameters. The counterclockwise angle between the vector $\mathbf{p}_i - \hat{\mathbf{c}}$ and $\mathbf{p}_{i+1} - \hat{\mathbf{c}}$ is denoted as β_i for $i = 1, \dots, n - 1$, and the angle between $\mathbf{p}_n - \hat{\mathbf{c}}$ and $\mathbf{p}_1 - \hat{\mathbf{c}}$ is denoted as β_n , i.e.,

$$\begin{aligned} \beta_i &= \angle(\mathbf{p}_{i+1} - \hat{\mathbf{c}}, \mathbf{p}_i - \hat{\mathbf{c}}), & i = 1, \dots, n - 1 \\ \beta_n &= \angle(\mathbf{p}_1 - \hat{\mathbf{c}}, \mathbf{p}_n - \hat{\mathbf{c}}). \end{aligned} \quad (4.3)$$

Notice that in this case,

$$\beta_i(0) \geq 0, \quad \text{and} \quad \sum_{i=1}^n \beta_i(0) = 2\pi. \quad (4.4)$$

This is represented in figure Fig. 4.3.

We can define the distance of each agent i to the center as $D_i^c = \|\mathbf{c} - \mathbf{p}_i\|$. Since we do not have access to the center \mathbf{c} , the distance to the estimated center is represented as $\hat{D}_i^c = \|\hat{\mathbf{c}} - \mathbf{p}_i\|$. Then, knowing that each robot has access to its distance to the boundary, we can define it as

$$D_i^b = D_i^c - r. \quad (4.5)$$

Each agent constantly measures this value, as in Fig. 4.3 and Fig. 4.4. Note that D_i^b is positive if the agent is outside the algal bloom area or negative if it is inside the algal bloom area. For example, if an agent i is inside the circle about 5 meters, then $D_i^b = -5$, and if this agent is outside the circle about 5 meters, then $D_i^b = 5$.

Now, we are ready to pose the problem of interest that will be solved in the following sections.

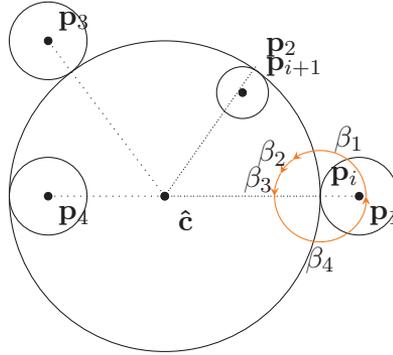


Figure 4.3: Example scheme of the system with four agents at positions p_1, p_4, p_3, p_2 . Note how each has access to the distance to the boundary, represented by a circumference.

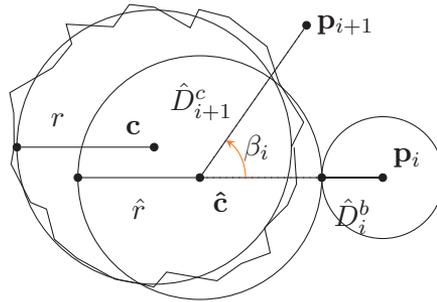


Figure 4.4: Scheme of the estimated \hat{c} , \hat{r} and the real target c , r as well as the angle β_i between two agents at p_{i+1} and p_i

Problem 2. Design estimators for $c(t)$ and $r(t)$ when both the distance measures (4.5) and GPS positions are available to each agent. Design the control input u_i for all the agents such that for some positive $\varepsilon_1, \varepsilon_2$,

$$\|\dot{\hat{c}}\| \leq \varepsilon_1 \tag{4.6}$$

$$|\dot{\hat{r}}| \leq \varepsilon_2, \tag{4.7}$$

there exist positive K_1, K_2 , and K_3 satisfying

$$\limsup_{t \rightarrow \infty} \|\hat{c}(t) - c(t)\| \leq K_1 \varepsilon_1, \tag{4.8}$$

$$\limsup_{t \rightarrow \infty} |\hat{r}(t) - r(t)| \leq K_2 \varepsilon_2, \tag{4.9}$$

$$\limsup_{t \rightarrow \infty} |D_i^b(t)| \leq K_3 \varepsilon_2, \quad (4.10)$$

$$\lim_{t \rightarrow \infty} \beta_i(t) = \frac{2\pi}{n}. \quad (4.11)$$

4.2 Optimal Circle Estimation and Control Algorithms

Here follows our solution for Problem 2. We consider n agents with positions \mathbf{p}_i , and we assume all of them are capable of measuring their distances D_i^b to the target boundary, including whether they're inside (D_i^b is negative) or outside (D_i^b is positive) of it. Then, they should estimate (\mathbf{c}, r) from their shared measurements. For robustness, they update their estimates by taking the average of the estimated variables by the n agents. Also, if one or more agents suffered faulty measurements due to bad conditions or failure, the system is ready to support that situation by using the remaining agent's estimates. Each agent calculates its desired velocity, considering its angle β_i to the next agent and its distance to the boundary. The scheme in Fig. 4.5 summarises this algorithm loop.

The first step is the estimation of the circle. Having all the agents constantly measuring D_i^b , we can fit a unique circle as in Fig. 4.3, given that the target shape is a circle. Mathematically, such a circle can be obtained through triangulation; therefore, we only need three agents to obtain a unique solution. However, more than three agents are considered for better coverage of all the fronts and robustness. Note that, in the result section, we used four agents. So, we apply the least squares method to obtain the approximated circle as in (4.12).

$$\min_{\hat{\mathbf{c}}, \hat{r}} \sum_i^n \left(\|\mathbf{p}_i - \hat{\mathbf{c}}\| - (\hat{r} + D_i^b) \right)^2. \quad (4.12)$$

s.t. $\hat{r} > 0$.

We want to obtain the desired control input \mathbf{u}_i using the previously measured and estimated variables. The total velocity of each agent comprises two sub-tasks: approaching the target and circumnavigating it. Therefore, we define the direction of each agent towards the center of the target as the bearing $\psi_i(t)$,

$$\psi_i = \frac{\hat{\mathbf{c}} - \mathbf{p}_i}{\hat{D}_i^c} = \frac{\hat{\mathbf{c}} - \mathbf{p}_i}{\|\hat{\mathbf{c}} - \mathbf{p}_i\|}. \quad (4.13)$$

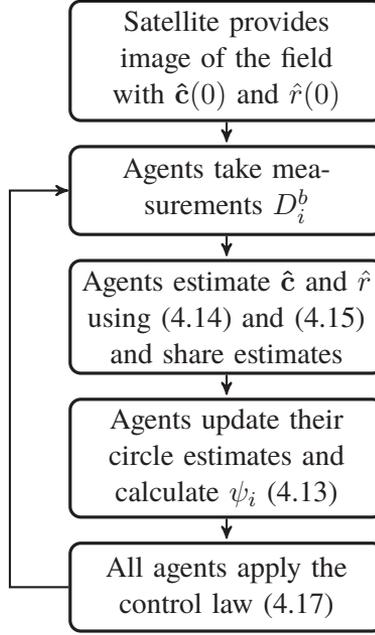


Figure 4.5: Scheme of the algorithm run on the system

Note that ψ_i in (4.13) is not well-defined when $\hat{D}_i^c = 0$, thus we need to prove that this singularity is avoided for all $t \geq 0$ in Theorem 3.1.

To build the control, we need to define $\dot{\hat{\mathbf{c}}}$ and $\dot{\hat{r}}$. Even though $\mathbf{c}(t)$ and $r(t)$ are continuous functions, our estimates $\hat{\mathbf{c}}$ and \hat{r} are, inevitably, a discrete function. Therefore, for each time interval Δ_T , we approximate $\dot{\hat{\mathbf{c}}}(t)$ and $\dot{\hat{r}}(t)$ as

$$\dot{\hat{\mathbf{c}}}[t] = \frac{\hat{\mathbf{c}}[t + \Delta_T] - \hat{\mathbf{c}}[t]}{\Delta_T} \quad (4.14)$$

$$\dot{\hat{r}}[t] = \frac{\hat{r}[t + \Delta_T] - \hat{r}[t]}{\Delta_T} \quad (4.15)$$

The first sub-task is related to the bearing ψ_i , and the second is related to its perpendicular, $E\psi_i$. Here, E is as defined in 3.34. Therefore, let us first consider the control law \mathbf{u}_i where δ is a parameter to be defined.

$$\mathbf{u}_i = \dot{\hat{\mathbf{c}}} + ((\hat{D}_i^c - \hat{r}) - \frac{1}{\delta} \dot{\hat{r}}) \psi_i + \beta_i \hat{D}_i^c E \psi_i \quad (4.16)$$

The control actuation of an AUV is limited; therefore, we have to make sure

that the implemented control is within the actuation bounds, and so we introduce

$$\mathbf{U}_i = \delta \mathbf{u}_i, \quad (4.17)$$

where δ is the same as before. For a specific \mathbf{u}_i , it is possible to have \mathbf{U}_i within some specified bounds.

4.3 Convergence Results

Theorem 4.1 *Consider the system (4.2) with the control protocol (4.17), and $\|\dot{\mathbf{c}}\| \leq \varepsilon_1$, $|\dot{r}| \leq \varepsilon_2$, then there exists K_1 , K_2 , and K_3 such that circumnavigation of the moving circle with equally spaced agents can be achieved asymptotically up to a bounded error, i.e.*

$$\limsup_{t \rightarrow \infty} \|\hat{\mathbf{c}}(t) - \mathbf{c}(t)\| \leq K_1 \varepsilon_1, \quad (4.18)$$

$$\limsup_{t \rightarrow \infty} |\hat{r}(t) - r(t)| \leq K_2 \varepsilon_2, \quad (4.19)$$

$$\limsup_{t \rightarrow \infty} |D_i^b(t)| \leq K_3 \varepsilon_2, \quad (4.20)$$

$$\lim_{t \rightarrow \infty} \beta_i(t) = \frac{2\pi}{n}. \quad (4.21)$$

Proof 4.1 *The proof is divided into three parts. In the first part, we prove that the estimated distance \hat{D}_i^c converges to the estimated radius \hat{r} , or in other words, that (4.20) holds. In the second part, we prove that the singularity of the bearing ψ_i is avoided. In the last part, we show that the angle between the agents will converge to the average consensus for n agents, $\beta_i = \frac{2\pi}{n}$, meaning (4.21) holds.*

1. *We prove that all agents reach the estimate of the boundary of the moving circles asymptotically, i.e., $\lim_{t \rightarrow \infty} \hat{D}_i^c(t) = \hat{r}(t)$, so (4.20) holds.*

Consider the function $W_i(t) := \hat{D}_i^c(t) - \hat{r}(t)$ whose time derivative for $t \in [0, \tau_{\max})$ is given as

$$\begin{aligned} \dot{W}_i &= \frac{(\hat{\mathbf{c}} - \mathbf{p}_i)^\top (\dot{\hat{\mathbf{c}}} - \dot{\mathbf{p}}_i)}{\hat{D}_i^c} - \dot{\hat{r}} \\ &= -\frac{(\hat{\mathbf{c}} - \mathbf{p}_i)^\top}{\hat{D}_i^c} \psi_i \delta (\hat{D}_i^c - \hat{r} - \dot{\hat{r}}) - \frac{(\mathbf{c} - \mathbf{p}_i)^\top}{\hat{D}_i^c} E \psi_i \delta \beta_i \hat{D}_i^c - \dot{\hat{r}} \\ &= -\delta (\hat{D}_i^c - \hat{r} - \dot{\hat{r}}) - \dot{\hat{r}} \end{aligned}$$

$$= -\delta W_i.$$

Hence for $t \in [0, +\infty)$, we have $\hat{D}_i^c(t) = \delta W_i(0)e^{-t} + \hat{r}(t)$ which implies W_i is converging to zero exponentially.

2. Now, we prove that ψ_i in (4.13) is well-defined, or in other words, that its singularity is avoided for all time $t \geq 0$, $\hat{D}_i^c \neq 0 \forall t$.

Having $\hat{D}_i^c(t) = \delta W_i(0)e^{-t} + \hat{r}(t)$ from the previous proof and knowing that $W_i(0)$ is always positive and that it converges to zero exponentially, we have that if $\hat{r}(t) > 0$ then $\hat{D}_i^c(t) > 0, \forall t$.

So we would have to prove that $\hat{r}(t) > 0 \forall t$. Given that we use the least squares method to obtain the estimate of the radius, we can see how one of the constraints guarantees that $\hat{r}(t) > 0 \forall t$. Then we conclude that $\hat{D}_i^c \neq 0 \forall t$ and that the bearing $\psi_i(t)$ is well defined $\forall t$.

3. Finally, we show that the angle between the agents will converge to the average consensus for n agents, $\beta_i = \frac{2\pi}{n}$, so (4.21) holds.

Firstly, note that we can write an angle between two vectors $\beta_i = \angle(v_2, v_1)$ as

$$\beta_i = 2 \operatorname{atan2}((v_1 \times v_2) \cdot z, \|v_1\| \|v_2\| + v_1 \cdot v_2) \quad (4.22)$$

and its derivative as

$$\dot{\beta}_i = \frac{\hat{v}_1 \times z}{\|v_1\|} \dot{v}_1 - \frac{\hat{v}_2 \times z}{\|v_2\|} \dot{v}_2 \quad (4.23)$$

where $z = \frac{v_1 \times v_2}{\|v_1 \times v_2\|}$, $\hat{v}_i = \frac{v_i}{\|v_i\|}$, $i = 1, 2$.

Then, for $v_1 = \mathbf{p}_i - \hat{\mathbf{c}}$ and $v_2 = \mathbf{p}_{i+1} - \hat{\mathbf{c}}$ we get

$$\begin{aligned} \dot{\beta}_i &= \frac{\hat{v}_1 \times z}{\|v_1\|} \dot{v}_1 - \frac{\hat{v}_2 \times z}{\|v_2\|} \dot{v}_2 \\ &= \frac{\hat{v}_1 \times z}{\|v_1\|} \delta((\hat{D}_i^c - \hat{r} - \dot{\hat{r}})\psi_i + \beta_i \hat{D}_i^c E \psi_i) \\ &\quad - \frac{\hat{v}_2 \times z}{\|v_2\|} \delta((\hat{D}_{i+1}^c - \hat{r} - \dot{\hat{r}})\psi_{i+1} + \beta_{i+1} \hat{D}_{i+1}^c E \psi_{i+1}) \\ &= -\frac{1}{\|v_1\|} \beta_i + \frac{1}{\|v_2\|} \beta_{i+1} \\ &= \delta(-\beta_i + \beta_{i+1}), \quad i = 1, \dots, n-1 \\ \dot{\beta}_n &= \delta(-\beta_n + \beta_1). \end{aligned}$$

which can be written in a compact form as following

$$\dot{\beta} = -\delta B^\top \beta \quad (4.24)$$

where B is the incidence matrix of the directed ring graph from v_1 to v_n . First, we note that the system (4.24) is positive (see e.g., [114]), i.e., $\beta_i(t) \geq 0$ if $\beta_i(0) \geq 0$ for all $t \geq 0$ and $i \in \mathcal{I}$. This proves the positions of the agents are not interchangeable. Second, noticing that B^\top is the (in-degree) Laplacian of the directed ring graph which is strongly connected, then by Theorem 6 in [115], β converges to consensus $\frac{2\pi}{n} \mathbf{1}$.

Remark 4.1 Note how the agent A_i will maintain its relative position p_i throughout the circumnavigation mission. We can prove that agent A_i is always in position p_i .

Remark 4.2 We proved both convergence of the angle to the average consensus for n agents and convergence of these agents towards the boundary of the target up to a given bound. Therefore, we guarantee collision avoidance.

4.4 Simulation Results

In this section, we present simulations for the protocol designed in section 4.2. We use the derived method to estimate the target (4.12) and the controlling protocol for the agents (4.17). For this section, we discretize the whole algorithm to be able to use it computationally.

We use the target in the images provided by SINMOD simulations [117]. The present simulation corresponds to approximately four days of data, and the target we obtained is approximately 1-3km in radius.

In Fig. 4.6, we can see the robot system circumnavigating the algal bloom target in a time-lapse. This specific algal bloom target is challenging as it shape-shifts quite abruptly. Note that the agents were deployed in positions in the boundary, so their initial error $D_i^b(0)$ is zero. Note also how, in some instances of the mission, the target moves fast to such an extent that results in a delayed motion for the robots. This effect is foreseen and explained in Theorem 4.1.

Analyzing the simulations, we observe each variable in Fig. 4.7. Firstly, we can see the comparison between the real position of the target and the estimates our algorithm provided. We can observe that the estimation closely follows the real value with a very small error. Secondly, we analyze the distance of agent 1 to

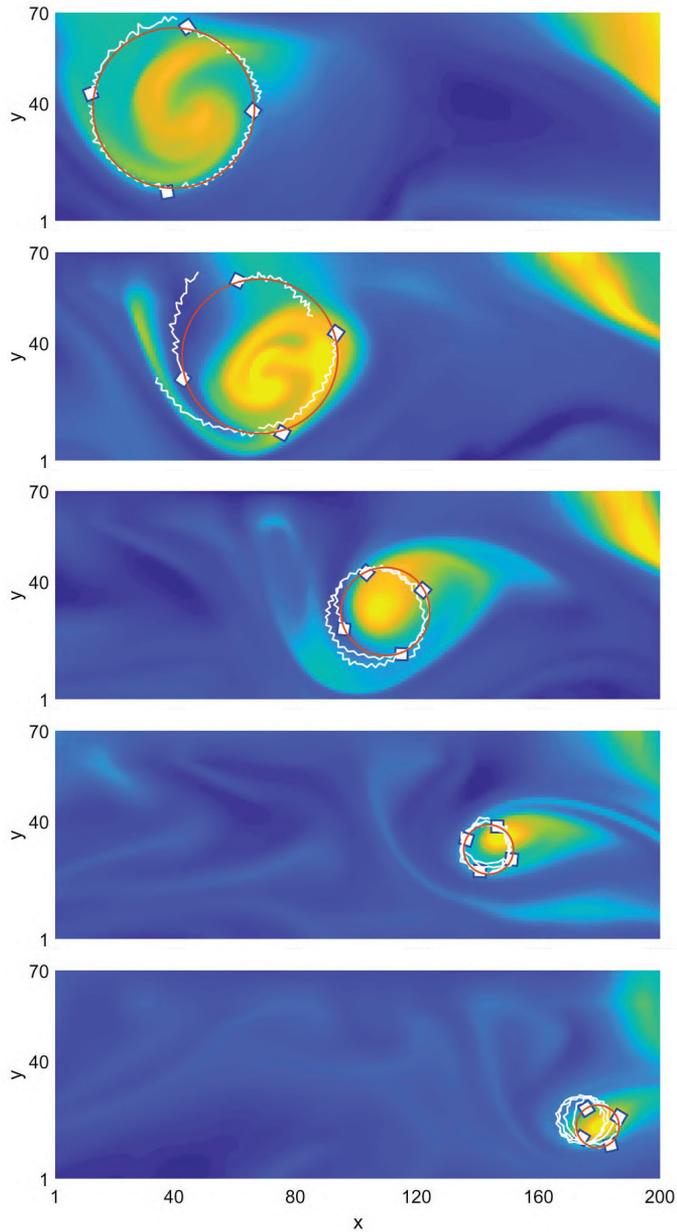


Figure 4.6: Time-lapse of four agents circumnavigating a moving target (red) with a representation of their paths (white). Each plot is approximately half a day after the previous one.

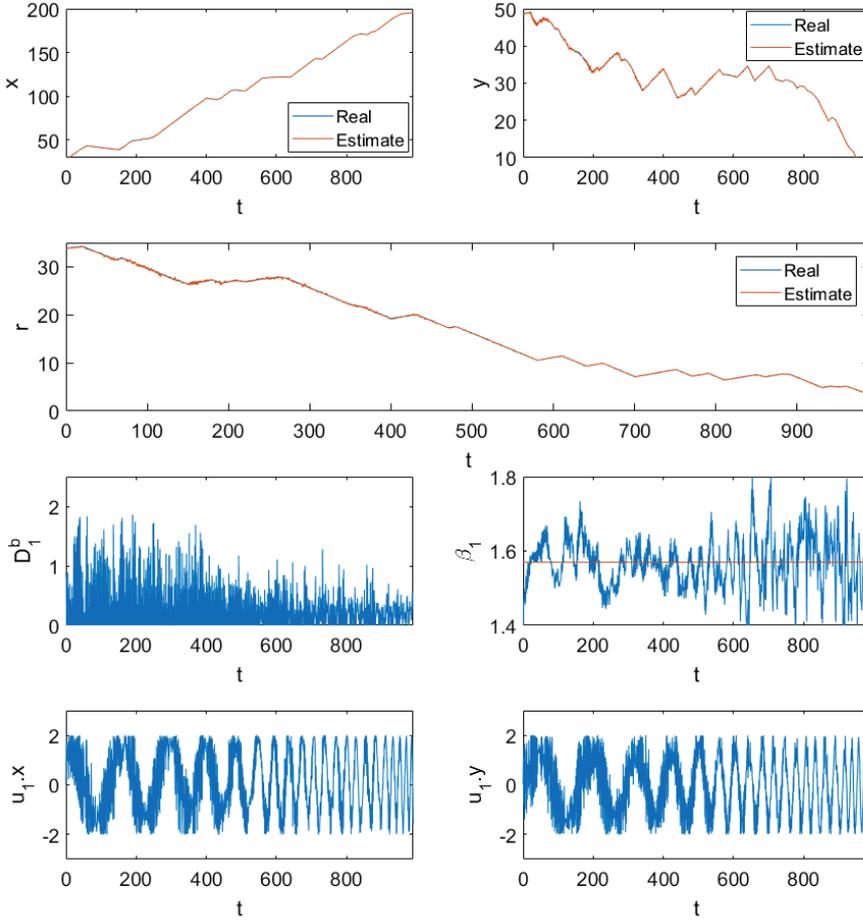


Figure 4.7: First and second row: real and estimated target's center $\mathbf{c} : x, y$ and radius r . Third row: tracking error of agent 1, D_1^b and angle β_1 . Fourth row: control input of agent 1, $\mathbf{u}_1 : x, y$

the boundary D_1^b and the angle between agent 1 and 2, β_1 . We can see the error is within the expected boundaries according to Theorem 1. Regarding the distance to the boundary, the error never exceeds 2 units (200 meters) and is usually up to 1 unit (100 meters). Note that each x and y coordinate unit corresponds to about 100 meters. Also, each time iteration unit corresponds to 6min. As for the angle between agents, the maximum error is 0.2 radians, corresponding to

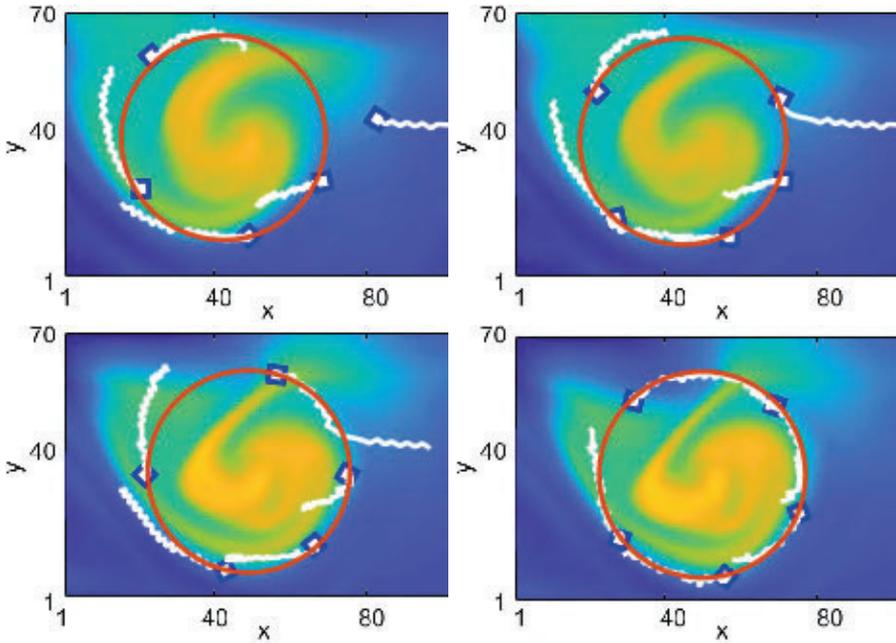


Figure 4.8: Time-lapse of five AUVs (blue rectangles) circumnavigating a moving target (red) with a representation of their paths (white)

a maximum angle error of 11 degrees. If we look at the plots for the control input of our agents, namely, for agent 1, we can see how the control was applied up to a maximum value. We defined the maximum speed of the agent for each coordinate to be 2 y units per 1 x unit, corresponding to 2km/h in each Cartesian direction ($200\text{m} / 6\text{min} = 2\text{km/h}$).

Having the same protocols and data set, we simulated a different scenario where a fifth vehicle enters the system, as seen in Fig. 4.8. We can see how the vehicles adjust the angles between each other to make space for the new vehicle. Notice how, in the last figure, the vehicles seem to converge to a regular polygon formation. This scenario represents the scaling possibility of our protocol.

4.5 Extension for Non-circular Shapes

In this section, we extend the results from Section 4.2 to a more general scenario. We consider circumnavigating irregular, non-circular moving shapes. Our goal

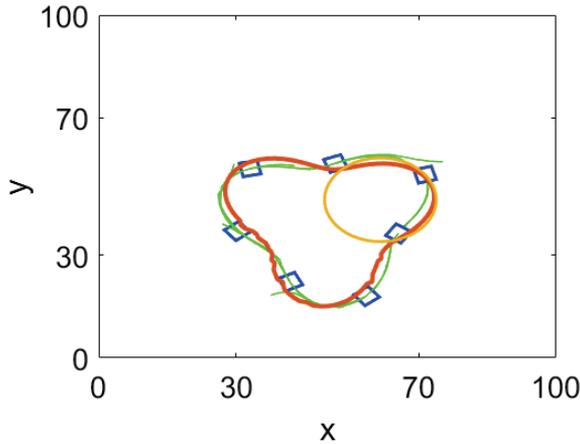


Figure 4.9: Example of an irregular shape with seven vehicles circumnavigating it.

is to circumnavigate an irregular algal bloom target using a group of vehicles. Each vehicle is equipped with a sensor that indicates its distance to the target's boundary, including whether it is inside or outside of the target. The first step is the estimation of the parameters of the algal bloom curvature in each region of the shape, that is, its center and radius for every time instance and every vehicle. The second step is to design a control law for the vehicles to circumnavigate the shape. We perform a numerical study of the convergence of vehicles to the target's boundary. This section considers the problem of tracking an irregular, moving, and time-varying shape using a multi-vehicle system and a satellite. We see an example of such an irregular shape in Fig. 4.9.

An initial image of the algal bloom will begin the mission. The vehicles measure their distance to the target's boundary and whether they are inside or outside the target. Each vehicle shares this information with its two neighboring vehicles. We assume that all vehicles have a common sense of direction concerning the target, and we assume that every vehicle has one vehicle on the left and one on the right. Each vehicle has access to its GPS positions, its distance to the boundary of the target, and its two neighbors' distances to the target.

The system of vehicles will circumnavigate the target and provide real-time information on its boundary. We define n vehicles at positions $\mathbf{p}_i(0), i \in [1, \dots, n]$. They start outside the target and form a counterclockwise, undirected ring on

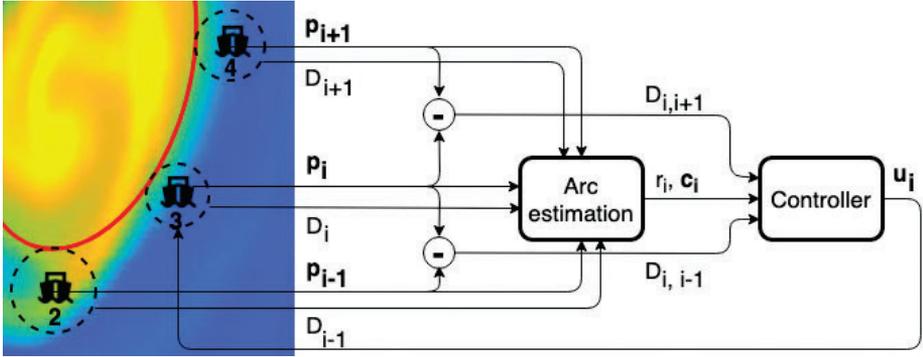


Figure 4.10: Control architecture for vehicle i communicating with its two neighbours $i - 1$ and $i + 1$.

the surface. The kinematics of the vehicles is the same as in previous chapters: $\dot{\mathbf{p}}_i = \mathbf{u}_i$, $i \in [1, \dots, n]$ where \mathbf{p}_i is a vector that contains the position $\mathbf{p}_i = [x_i, y_i]^\top \in \mathbb{R}^2$ and $\mathbf{u}_i \in \mathbb{R}^2$ is the control input.

Each vehicle should define a local curvature through a circle as in Fig.4.9. We define this circle as

$$(\mathbf{c}_i, r_i) \in \mathbb{R}^3, \quad (4.25)$$

where $\mathbf{c}_i = (x_i, y_i)$ and r_i are the circle's center and radius corresponding to vehicle i .

The distance of each vehicle to the boundary of the target is defined as the smallest measured distance to the boundary, D_i . It is measured by each vehicle. Note that D_i is positive if the vehicle is outside the target and negative if it is inside. We want to space the vehicles equally along the shape. We can do this by making the distances between vehicles approximately equal. For each vehicle i , its distance to the neighbour in the right $D_{i,i+1}$ and to the neighbour in the left $D_{i,i-1}$:

$$\begin{aligned} D_{i,i+1} &= \|\mathbf{p}_i - \mathbf{p}_{i+1}\|, & i &= 1, \dots, n-1 \\ D_{i,i-1} &= \|\mathbf{p}_i - \mathbf{p}_{i-1}\|, & i &= 2, \dots, n. \end{aligned} \quad (4.26)$$

$$D_{n,1} = D_{1,n} = \|\mathbf{p}_n - \mathbf{p}_1\|.$$

4.5.1 Control strategy and arc estimation

The control architecture is summarised in Fig. 4.10. We consider n vehicles at positions \mathbf{p}_i , and we assume all of them are capable of measuring their distances

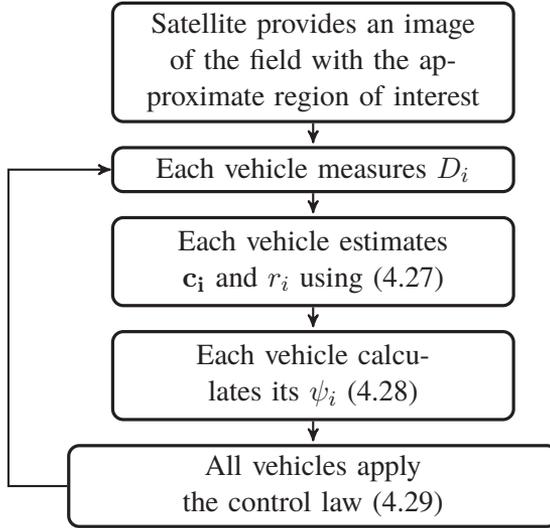


Figure 4.11: Algorithm for positioning the vehicles on the target boundary

D_i to the target boundary, including whether they are inside (D_i is negative) or outside (D_i is positive) of it. Then, each vehicle estimates (\mathbf{c}_i, r_i) from its two neighbors' shared measurements and positions. Each vehicle calculates its desired velocity considering its distance D_{i+1} and D_{i-1} to its neighbors and its distance to the boundary. The scheme in Fig. 4.11 summarises the algorithm.

The first step is the definition of the local circle for each vehicle. Having all the vehicles constantly measuring D_i , we fit some shape as in the left image of Fig. 4.12. But, since the target shape is not regular like a circle, we define curvatures for every time instance that each vehicle should follow. The curvature for each vehicle is defined in a distributed fashion using the information of its own and its two neighbors by creating a circle (\mathbf{c}_i, r_i) . For example, in the middle image of Fig. 4.12 we can see that vehicle 2 defines a circle using the information of vehicles 1, 2, and 3 while vehicle 3 defines a circle as in the right image of Fig. 4.12 using the information of vehicles 2, 3, and 4. Each vehicle estimates its circle from

$$\min_{\mathbf{c}_i, r_i} \sum_{k=i-1}^{i+1} (\|\mathbf{p}_k - \mathbf{c}_i\| - (r_i + D_k))^2, \quad i = 1, \dots, n. \quad (4.27)$$

s.t. $r > 0$.

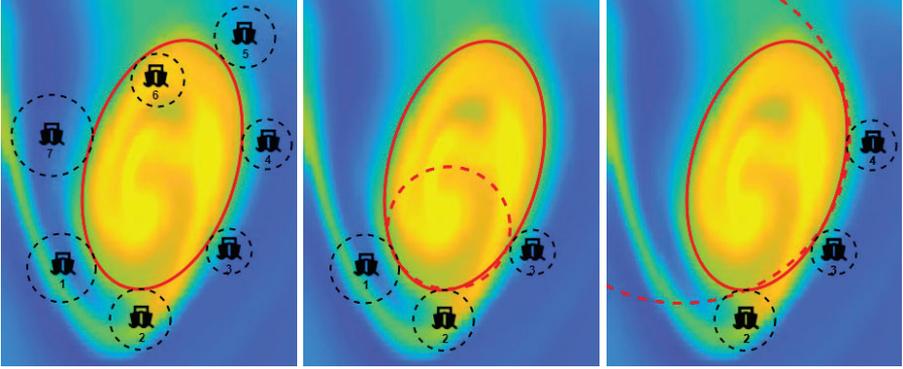


Figure 4.12: Left: 7 vehicles circumnavigate an algal bloom shape (represented by the red oval shape) while measuring their distances to its boundary (dashed black circles). Middle: vehicle 2 communicates with vehicles 1 and 3 to define its ideal curvature (red dashed line) for time instance t . Right: vehicle 3 communicates with vehicles 2 and 4 to define its ideal curvature (red dashed line) for time instance t .

Using the measured and estimated variables, we want to obtain the desired control input \mathbf{u}_i . The total velocity of each vehicle comprises two parts: approaching the target and circumnavigating it. Therefore, we define the direction of each vehicle towards the center of the target as the bearing,

$$\psi_i = \frac{\mathbf{c}_i - \mathbf{p}_i}{\|\mathbf{c}_i - \mathbf{p}_i\|}. \quad (4.28)$$

The control law for each vehicle i is

$$\mathbf{u}_i = D_i \psi_i + \frac{D_{i,i+1}}{D_{i,i-1}} E \psi_i \quad (4.29)$$

4.6 Simulation Results

This section presents simulations for the protocol designed in Section 4.5.1. In the first subsection, we apply our protocol to a slowly drifting and shape-shifting ellipsoid, and in the second subsection, we apply it to a static yet irregular shape that looks like a three-leafed clover.

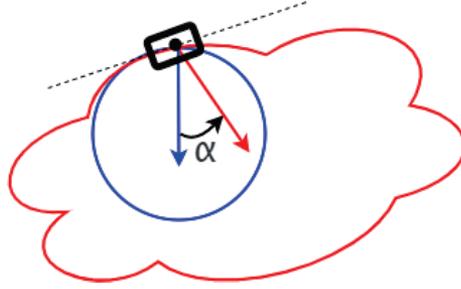


Figure 4.13: Representation of the angle α between the normal vector to the circle (blue) and the normal vector to the target (red) at the vehicle's location (black square and dot).

We define the angle α between the normal vector to the circle and the normal vector to the target at the vehicle's location as in Fig. 4.13. Note that, as seen in this figure, a perfect estimate corresponds to $\alpha = 0$.

4.6.1 Oval shape with five vehicles

We first simulate a moving target with initial position $(x[0], y[0]) = (50, 40)$, horizontal radius $r_h[0] = 25$, vertical radius $r_v[0] = 15$. The shape evolves with the following dynamics

$$\begin{aligned}
 x[t + 1] &= x[t] + \gamma_1[t] + 0.2 \\
 y[t + 1] &= y[t] + \gamma_2[t] + 0.2 \\
 r_h[t + 1] &= r_h[t] + \gamma_3[t] + 0.2 \\
 r_v[t + 1] &= r_v[t] + \gamma_4[t] + 0.2.
 \end{aligned} \tag{4.30}$$

We assume that each vehicle has access to a common initial noisy estimate of $(\hat{x}[0], \hat{y}[0]) = (50, 40)$, radius $\hat{r}[0] = 25$. Note that at time $t = 0$, the radius estimate equals the ellipsoid's largest radius. Here, $\gamma_i[t]$ is a random scalar drawn from the uniform distribution within the interval $[-0.5, 0.5]$ for $i = 1, \dots, 4$.

Fig. 4.14 shows the multi-vehicle system converges towards the moving target. The red ellipsoid is the target shape, the blue squares are the vehicles, and the green lines are the path each vehicle took. Fig. 4.15 illustrates how the AUVs estimate and track the target. Here the yellow circle is the circle estimate of one of the vehicles. As shown, the estimated circle of vehicle i partially coincides

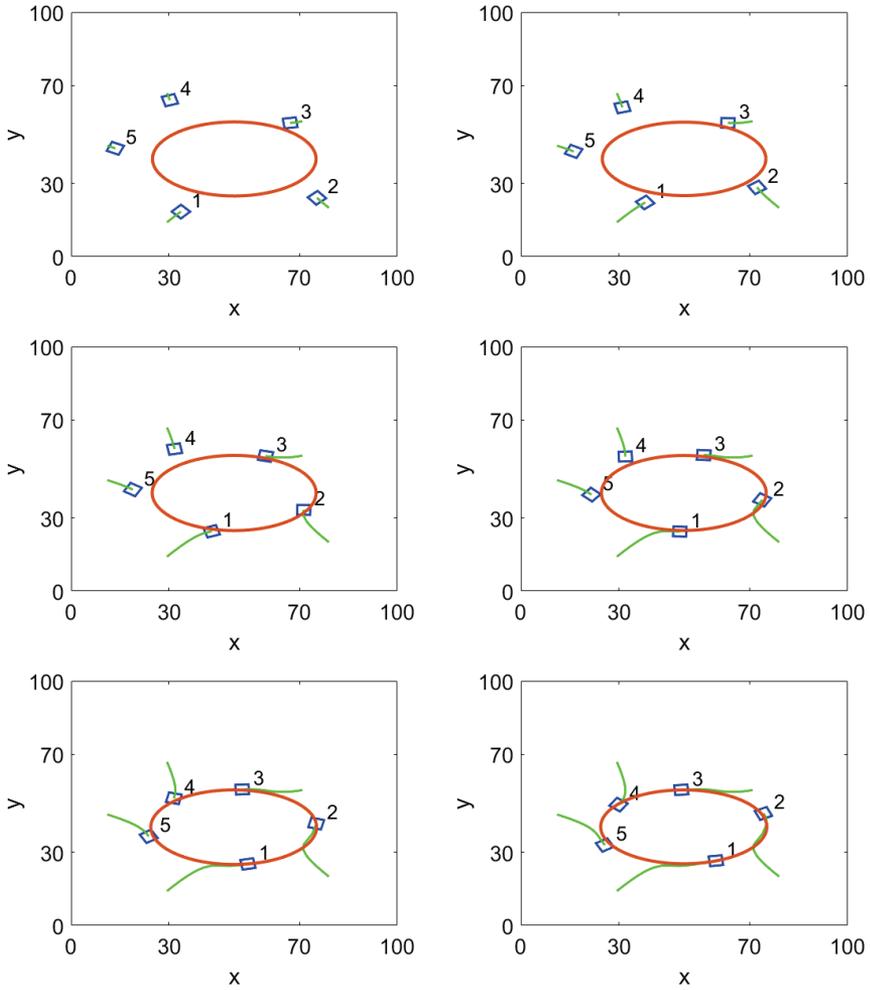


Figure 4.14: Time-lapse of five AUVs (blue rectangles) converging and starting to circumnavigate a moving target (red) with a representation of their paths (green).

with the target in the neighboring region of vehicle i . We can see that the trajectories closely match the target shape.

Fig. 4.16 shows a detailed analysis of this case study. The top two figures compare the ellipse target to the estimated circle for vehicle 1. This comparison

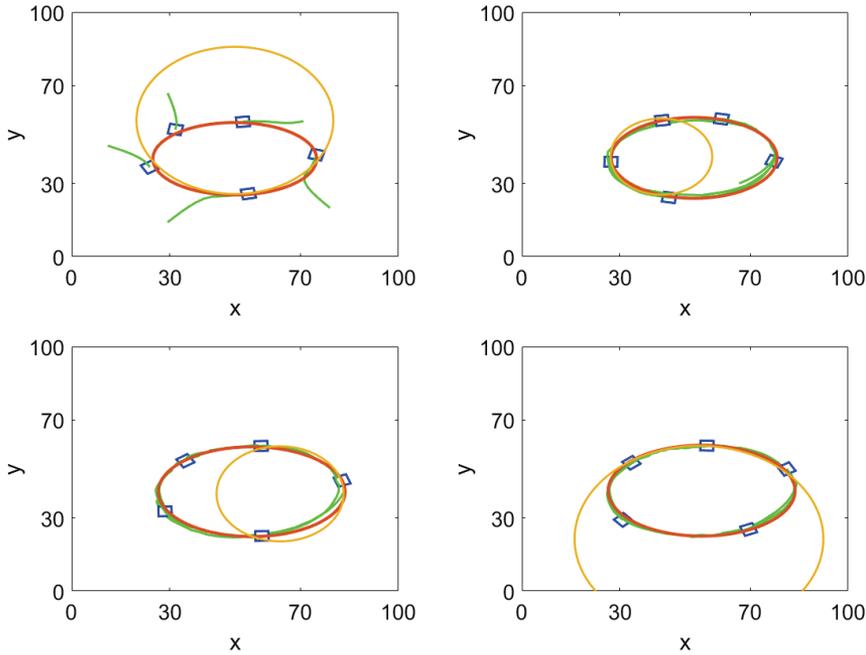


Figure 4.15: Time-lapse of five AUVs (blue rectangles) circumnavigating a moving target (red) with the estimated circle of one of them (yellow) and a representation of their paths (green)

comprises the center with x and y parameters. We can see that the center is, on average, close to the ellipsoid's center. The second row shows the time evolution of $\cos(\alpha)$, the angle between the normal vectors to the surface of both the ellipse and the circle. Note that the normal vector to the surface of the circle ψ_1 is used by each vehicle. If the estimation protocol is perfect, it should be equal to the normal vector to the target. This corresponds to $\alpha = 0$. The figure shows that the estimated normal vector is close to the target's normal vector.

The left plot in the third row shows the distance of vehicle 1 to the boundary of the target, D_1 . The vehicle starts far from the target and quickly converges toward its boundary. As the target moves and changes size and shape, the vehicle will constantly adjust its trajectory toward the boundary. This creates the observed ripple in the plot.

The third row right plot shows the ratio of two distances: the distance of vehicle 1 to the neighboring vehicles n and 2. As expected, this ratio is close

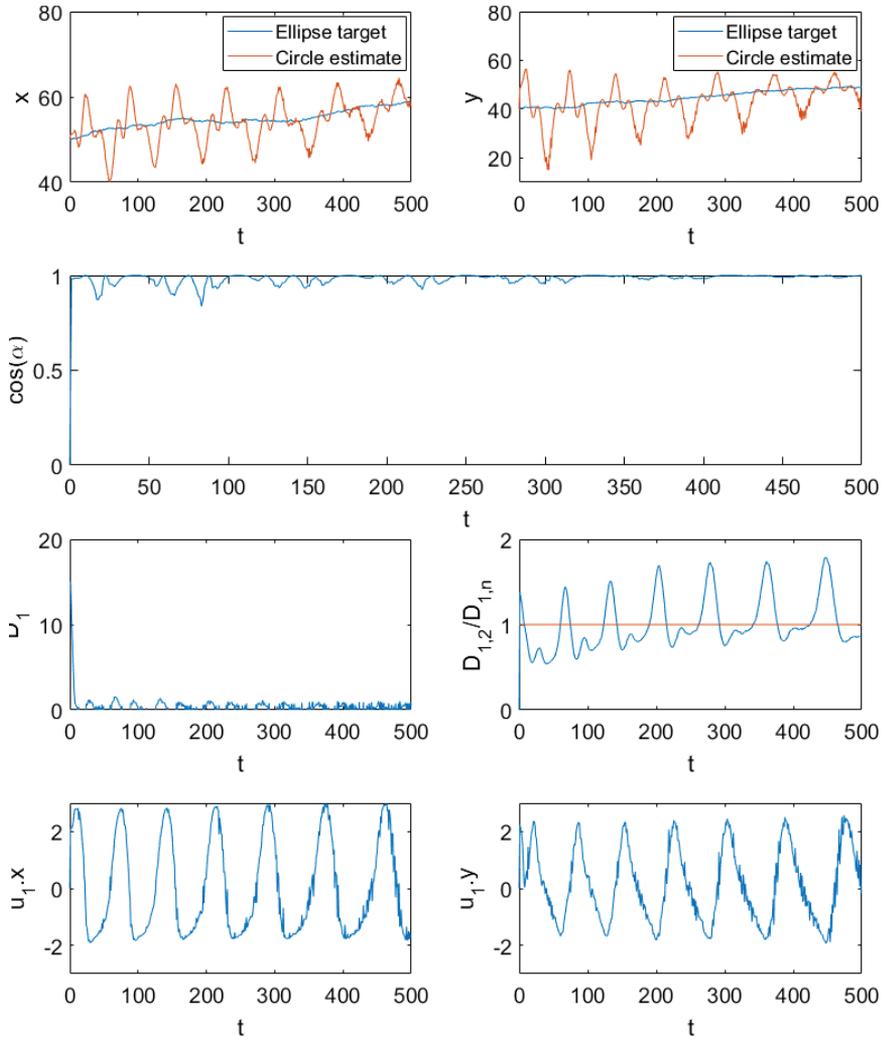


Figure 4.16: First row: target and estimated circle's center $\mathbf{c} : x, y$. Second row: Cosine of the angle between the normals to the target and circle $\cos(\alpha)$. Third row: tracking error of AUV A_1 , D_1 and ratio between vehicle's distances $D_{1,2}/D_{1,n}$. Fourth row: control input of AUV A_1 , $\mathbf{u}_1 : x, y$

to one. On the last row, we have the control input for vehicle 1. Note that the oscillation corresponds to a turn around the target.

4.6.2 Irregular shape with seven vehicles

We first simulate a static target shape with static position $(x, y) = (50, 40)$ and radius $r(\theta) = 5 \sin(0.06\pi\theta) + 20$. We assume that each vehicle has access to a common initial noisy estimate of $(\hat{x}[0], \hat{y}[0]) = (50, 40)$, radius $\hat{r}[0] = 20$. Note that at time $t = 0$, the radius estimate equals the shape's average radius. We use seven vehicles to track this target.

Fig. 4.17 presents the convergence towards the target. Fig. 4.18 shows, similarly to Fig. 4.15, how a local circle is used to compute the control. The plots indicate that the protocols perform well and that there are some oscillations due to the target shape. A more detailed analysis is given in Fig. 4.19. The top two plots show that the center is, on average, close to the target's center and oscillating around it.

4.7 Summary

In this chapter, we considered the problem of multi-vehicle target tracking for circular and non-circular shapes. The target was assumed to be an irregular dynamic shape approximated by a circle with a moving center and varying radius. The difference from the previous chapter is the setup and the protocols designed. We proposed a decentralized estimation protocol in which all agents measure their distance to the boundary and, by sharing this information, determine the optimal circumnavigation circle. This problem was mathematically defined by introducing relevant variables and equations that relate them. We also defined the measurements available to each agent and the new estimation and circumnavigation objectives. An optimal algorithm was proposed for estimating the target and a control algorithm, and we proved their mathematical convergence up to a bound according to our objectives. Two simulation results were presented: one to analyze convergence performance and the other to represent the possibility of scalability regarding introducing new vehicles of the system while applying the developed algorithms.

The proposed algorithm was extended to a more generic scenario for irregular dynamic shapes, maintaining the decentralized estimation protocol in which all vehicles measured their distance to the boundary. It was assumed that the vehicles share this information with their two neighbors, and each vehicle determines a circle that best approximates the curvature at the vehicle's area. We presented two numerical simulation results using a dynamic oval and non-circular

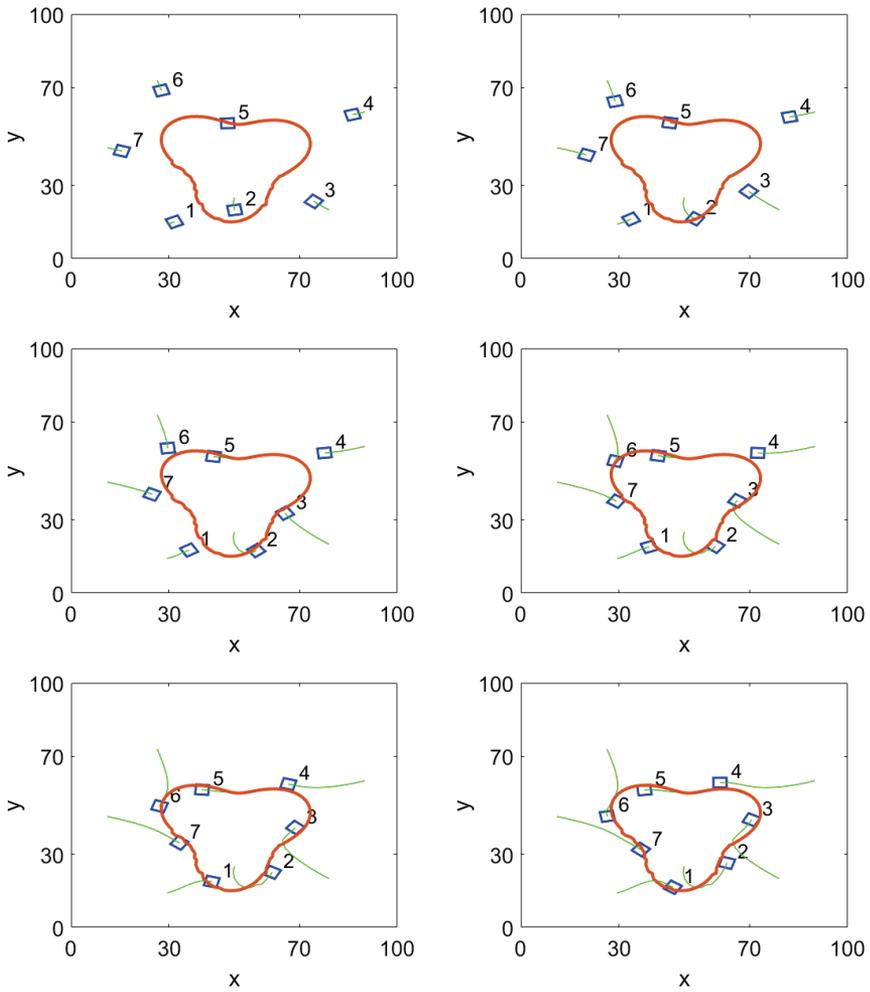


Figure 4.17: Time-lapse of five AUVs (blue rectangles) converging and starting to circumnavigate a moving target (red) with a representation of their paths (green).

static shape. Our algorithm succeeded for both simulations as we observed convergence with bounded and small tracking errors. The second simulation had bigger tracking errors than the first. This outcome is expected since the protocol relies on neighbor information for estimation, so the more irregular a shape is,

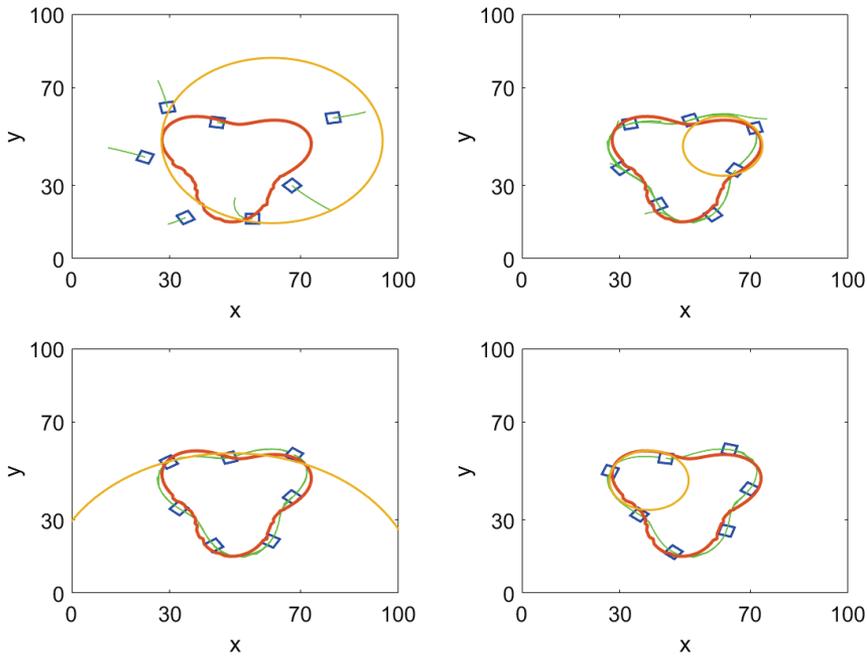


Figure 4.18: Time-lapse of seven AUVs (blue rectangles) circumnavigating a static target (red) with a representation of their trajectories (green) and the estimated circle of one of them (yellow).

the more vehicles are needed to estimate it accurately.

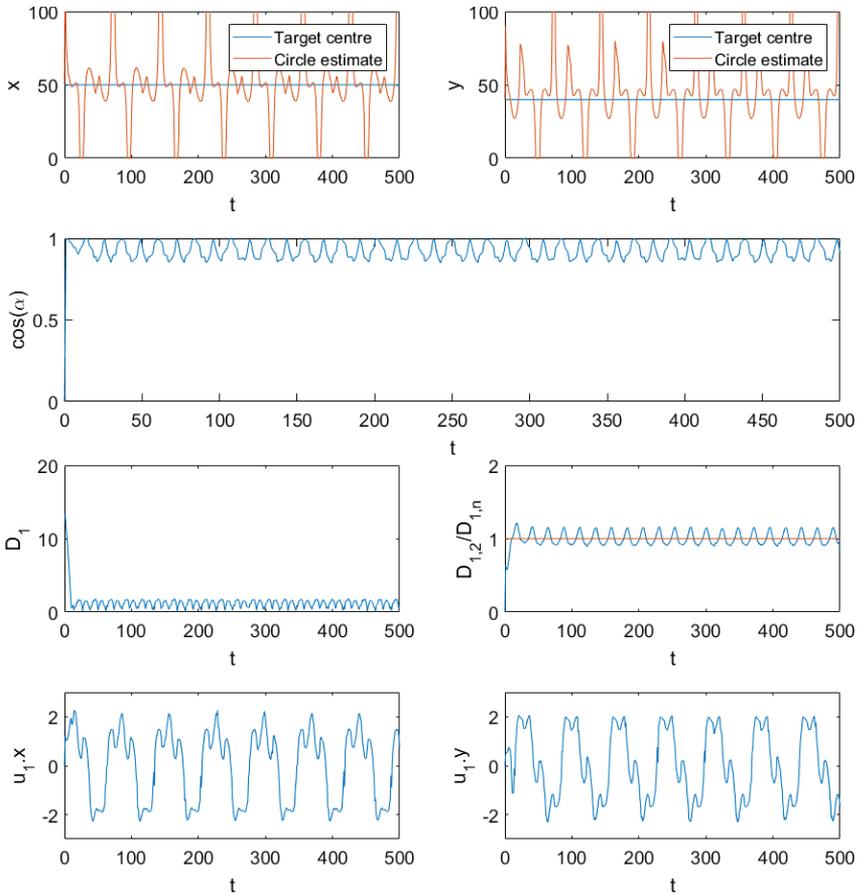


Figure 4.19: First row: target and estimated circle's center $c : x, y$. Second row: Cosine of the angle between the normals to the target and circle $\cos(\alpha)$. Third row: AUV A_1 's distance to the boundary D_1 and ratio of distances between neighbours $\frac{D_{1,2}}{D_{1,n}}$. Fourth row: control input of AUV A_1 , $\mathbf{u}_1 : x, y$.

Chapter 5

Linear Front Approximation for Single-agent Systems

This chapter considers the problem of tracking and mapping a river front with an AUV. The problem formulation is targeted at the scientific study of the processes by which the river and the ocean interact. The approach extends previous work in AUV front tracking to simultaneous tracking and mapping under different ocean and meteorological conditions. This is done with the help of parameterizable motion control algorithms to enable adaptation to these time-varying conditions. The approach is evaluated in simulation with the help of a high-resolution hydrodynamic model. The test plan covers over 300 test cases with the most representative combinations of the ocean and meteorological conditions.

This chapter is organized as follows. 5.1 describes the problem formulation. Section 5.2 is about the tracking and mapping approach, emphasizing the proposed algorithms and parameterizations of the ocean and meteorological conditions and how these conditions affect performance. It also involves deploying the tracking and mapping algorithms onboard an AUV for field testing. In Section 5.3, we present the test plan and discuss the results of the simulation tests. Concluding remarks come in Section 5.4.

5.1 Problem Formulation

In this chapter, we are interested in the processes by which fresh water (lower salinity) from the Douro River (Porto, Portugal) rich in nutrients, sediments, and

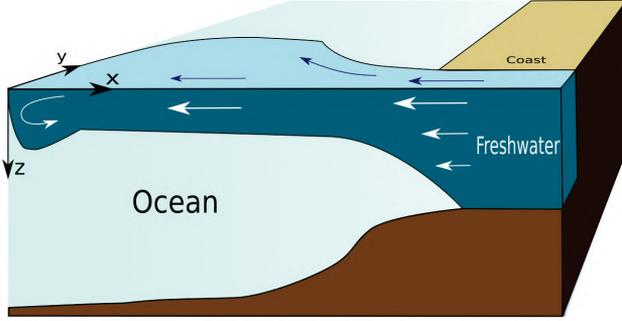


Figure 5.1: Diagram depicting a river front (adapted from [118], Fig. 1).

pollutants interact with denser ocean water with higher salinity (Figure 5.1). For this reason, it is important to study and map the dynamics of the front, which will help predict its influence on the coastal areas surrounding the river mouth. The front is quite dynamic, mainly because of winds and tidal forcing, and the front moves back and forth because of the tides. The Douro River front is just a thin layer of fresh water (typical thickness is around 2 meters) moving over ocean waters.

5.1.1 Problem Statement

We want to map the Douro River front, specifically in the salinity and associated temperature maps. Thus, we need a few definitions.

The river front is defined as a scalar field evolving with time:

$$Front : D \subset \mathbb{R}^4 \rightarrow \mathbb{R}^2 \quad (5.1)$$

where D is a closed set. $Front$ has two variables of interest, salinity, S , and temperature, T :

$$Front(x, y, z, t) = [T(x, y, z, t), S(x, y, z, t)] \quad (5.2)$$

No assumptions on the mathematical model that describes these variables are made. The evolution of $Front$ is determined by a set of parameters $Parameters_{Front} = \{\rho_1, \dots, \rho_n\}$. They describe wind, tide, salinity profile of the front, and velocity of the front, which directly influence the shape and position of the front. Some parameters are unknown, others are known a priori, and others are calculated during the mission. We do not have direct access to the function $front$, only to

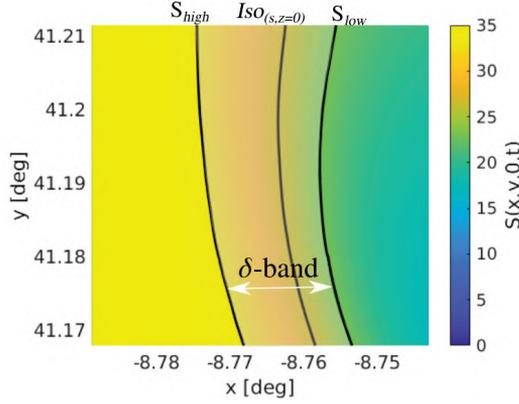


Figure 5.2: Representation of the δ -band defined by the salinity values S_{low} and S_{high} .

point-wise measurements taken by an AUV along a trajectory $Traj : [t_0, t_f] \rightarrow \mathbb{R}^3$. The position, p of the AUV at time t , is $p(t) = [x(t), y(t), z(t)]$. The evolution of p is governed by the equation of motion:

$$\dot{p}(t) = f(p(t), u(t), v(t), t), \quad u(t) \in U, \quad v(t) \in V \quad (5.3)$$

where $u(t)$ and $v(t)$ are, respectively, the AUV control inputs and the velocity of the currents; U and V are compact sets.

The measurements of salinity and temperature taken by the AUV at time t are, respectively, $\hat{S}(p(t), t)$ and $\hat{T}(p(t), t)$. The streams of salinity and temperature measurements, taken by the AUV up to time t are respectively

$$StreamS(t) : [t_0, t_f] \rightarrow C, \quad StreamS(t) = \hat{S}(Traj(t), t) \quad (5.4)$$

$$StreamT(t) : [t_0, t_f] \rightarrow C, \quad StreamT(t) = \hat{T}(Traj(t), t) \quad (5.5)$$

where C is the space of continuous functions $[\mathbb{R} \rightarrow \mathbb{R}]$.

The problem addressed in this chapter is:

Problem 5.1 Given an initial AUV position $p(t_0)$ and a set of parameters defined as $Parameters_{Front}$:

i) Track a δ -band centered on the s isoline of the front located at depth $z_d = 0$, $Iso_{(s, z_d=0)} = \{ (x, y, z, t) : S(x, y, z_d, t) = s \}$.

ii) Map the front in this δ -band defined by two salinity threshold values, $[S_{low}, S_{high}]$ as in Fig. 5.2.

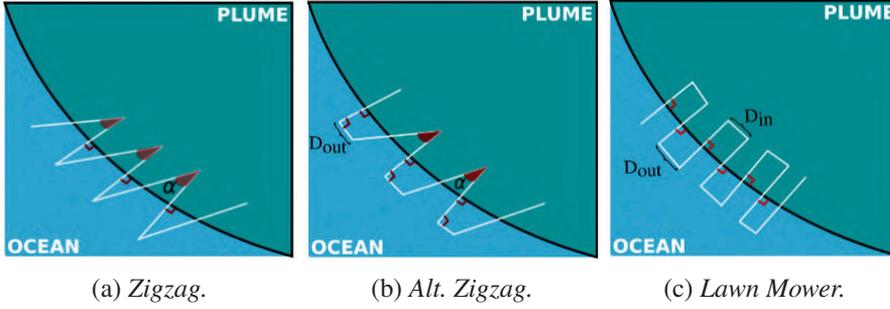


Figure 5.3: Schematic of the algorithms for tracking and mapping the Douro River front.

- iii) Ensure that the direction of travel of the AUV when crossing the front is perpendicular to $Iso_{(s,z_d)}$ for half of the total crossings.
- iv) Map the isosurface where $S(x, y, z, t) = s$ in the δ -band.

The requirement for perpendicular crossings of the front is because non-perpendicular crossings can lead to significant distortions in calculating the front characteristics, such as spreading width, length of the hydrodynamic mixing zone, and minimum dilution.

5.2 Front Detection Algorithm

The approach for tracking and mapping the Douro River front is based on three motion adaptation algorithms and a front estimation procedure. The algorithms are tuned with a few parameters to enable user-selected adaptation to ocean and meteorological conditions.

5.2.1 Algorithms

We propose three algorithms to track and map the Douro river front (Figure 5.3). These algorithms are characterized by the shape of the resulting paths. The paths depicted in this figure are horizontal projections of 3D yoyo paths to be performed by the vehicle.

The *Zigzag* is characterized by two straight line paths that make an angle α when the paths cross inside the front. In addition, this algorithm also guarantees,

up to some error, that the front is crossed in a direction perpendicular to it when the vehicle enters the front.

The *Alternative Zigzag* adds a straight line segment of length D_{out} parallel to the front for excursions outside the front. This trajectory results in a faster front progression since it increases the space between front crossing points compared to the *Zigzag* trajectory.

The ideal crossing of the front should be perpendicular to the front, as this is the situation that yields the ideal conditions for sampling the salinity profile of the front. For this reason, a path that always traverses the front with perpendicular crossings would be ideal. This is achieved by performing a parallel transect to the front of length D_{in} when moving inside the front. This path is referred to as *Lawn Mower*. Change of directions are triggered when the salinity thresholds $[S_{high}, S_{low}]$ are met and a minimal distance D_{min} or a maximum distance D_{max} from the front are reached.

5.2.2 Front estimation and detection

Initially, the vehicle does not have any information on the shape and parameters of the front. This poses two different problems. Firstly, an initial trajectory must be capable of crossing the front without any previous information on its shape. Secondly, the shape of the front must be predicted to determine the perpendicular segments to it.

The problem of finding the front is addressed by having the vehicle perform a classical zigzag trajectory with an angle α between all trajectory segments. The first three front crossings enable the estimation of some essential front parameters. After these three crossing points, the trajectory adaptation algorithms are initiated. For the front's prediction, the last and third last front crossing points, P_{cross_n} and $P_{cross_{n-2}} \in \mathbb{R}^2$, are used to perform a linear prediction. The vector, $\vec{f} = [f_1, f_2]^T$, is formed by the two points and represents the estimation of the front's direction, γ . Thus,

$$\begin{aligned}\vec{f} &= \vec{P}_{cross_n} - \vec{P}_{cross_{n-2}}, \\ \gamma &= \angle \vec{f} = \text{atan2}(f_2, f_1).\end{aligned}\tag{5.6}$$

The direction γ is used to compute the perpendicular segments. The error between the true perpendicular direction and the predicted is ϵ . Figure 5.4 shows

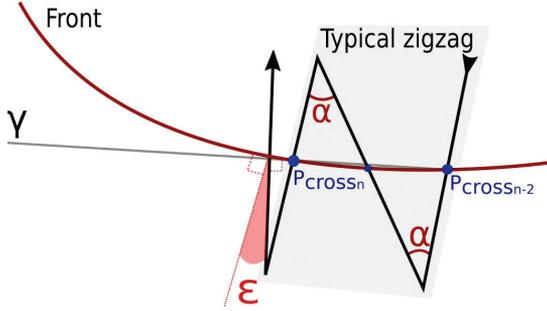


Figure 5.4: Schematic representation of the typical zigzag trajectory and the first front estimation γ after three crossings of the front.

a schematic example of the first zigzag trajectory and the process of predicting the direction of the front after detecting three crossing points.

The front is a thin layer of fresh water located at the surface. Hence, the first crossing of the front is done at the surface and will determine the front's reference salinity value s . This reference salinity value is calculated for every new mission.

Since the front is characterized by a steep change of the salinity field, the first survey at the surface is used to find the maximum derivative of the sampled values and the associated instant t_{max} where that maximum rate of change occurs. Let s be the value of salinity at instant t_{max} (5.7). This estimates the maximum gradient of the salinity field in that region.

$$t_{max} = \left\{ t : \operatorname{argmax} \left(\frac{\partial \hat{S}(x, y, 0, t)}{\partial t} \right) \right\} \quad (5.7)$$

$$s = \hat{S}(x, y, z, t_{max})$$

5.2.3 3D mapping

The mapping of the front has to be done in 3 dimensions. This is accomplished by having the vehicle perform a yoyo trajectory, defined by the pitch angle $\theta = 15^\circ$ between the path and the horizontal plane and cycling between the surface and some maximum depth z_{max} . The nominal value for z_{max} is chosen to be 10m because the thickness of the front is in the order of 2m in nominal conditions. The front crossing logic is enabled only when the depth of the vehicle

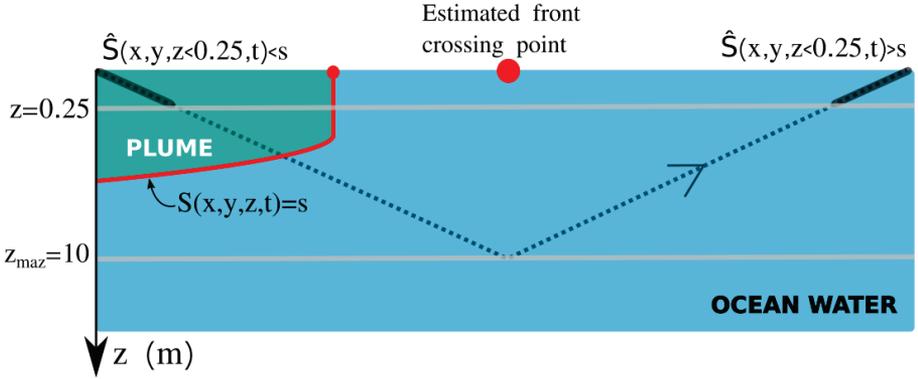


Figure 5.5: Vertical profile of the AUV yoyo trajectory (black dashed line) while crossing the front. The estimated crossing point is the middle point between the surface locations.

exceeds 25 centimeters ($z > 0.25$ m).

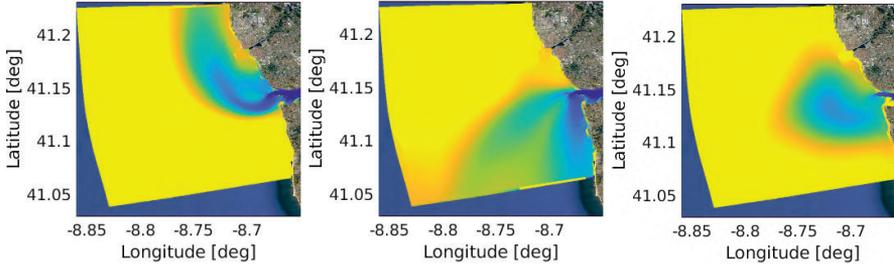
Figure 5.5 shows a typical situation where the vehicle crosses the front when starting inside the front. A front crossing, P_{cross} , is declared as the middle point between the two last surface locations, $P_{surface} \in \mathbb{R}^2$ (5.8).

$$P_{cross_n} = \frac{P_{surface_i} + P_{surface_{i-1}}}{2} \quad (5.8)$$

5.2.4 Environmental and algorithm parameters

In this subsection, we discuss the parameters affecting the performance of the algorithms. We consider three environmental parameters: tide, wind, and river outflow. Tides determine the time window the mission should occur, which is about 6 hours during the ebb phase. In what follows, we consider that missions take place during this phase.

Wind plays an important role in determining the shape of the front. There are three main shape fronts caused by southerly, northerly, and east or light wind conditions, as seen in Fig. 5.6. This knowledge is used to select the initial position of the AUV for any sampling mission. The initial direction of the vehicle is parallel to the coast. If there is a northerly wind, the vehicle starts in a northward direction. Conversely, if a southerly wind is identified, we start the vehicle in a southward direction. The initial location is always fairly close to the river



(a) 9th May 2016 - *S front*. (b) 15th May 2016 - *N front*. (c) 2nd May 2017 - *LW front*.

Figure 5.6: Typical front patterns for Southerly-S (a), Northerly-N (b), and East and/or light wind-LW (c) conditions.

Algorithm	Parameters	
<i>Zigzag</i>	α	
<i>Alt. Zigzag</i>	D_{out}	α
<i>Lawn Mower</i>	D_{out}	D_{in}

Table 5.1: Trajectory parameters.

Parameter	Value
z_{max}	10 m
θ	15°
$[S_{high}, S_{low}]$	[33, 23]
$[D_{min}, D_{max}]$	[500, 750]m

Table 5.2: Parameters used for mapping the Douro river front.

mouth to guarantee the vehicle is inside the front. The levels of the river outflow are a major consideration when it comes to field deployments of the vehicle. It may happen that the front is not well-defined for some regimes. This typically happens when the river discharge is too small or too large.

Next, we discuss the trajectory and mapping parameters used to tune the algorithms. These will directly influence the behavior and performance of the system. The user defines the settings of these parameters before deploying the AUV. The results presented in section 5.3 provide insights into optimal parameter selection. The trajectory parameters are presented in Table 5.1. The mapping parameters are presented next. These determine the mapping behavior of the vehicle. These are the maximum vertical distance (z_{max}); the yoyo/pitch angle (θ); front thresholds (S_{high}, S_{low}); and the minimum and maximum distance covered by the vehicle after crossing the front (D_{min}, D_{max}). Table 5.2 presents the parameters used for the case of the Douro river front.

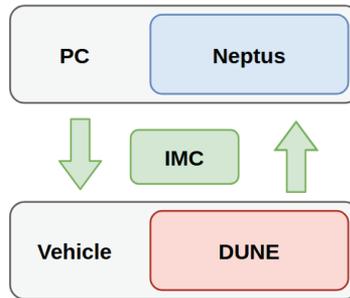


Figure 5.7: LSTS toolchain: components and interactions.

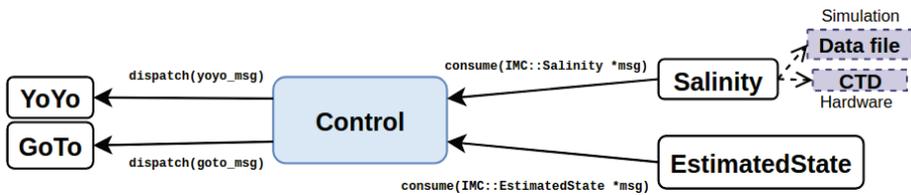


Figure 5.8: Control interactions – IMC messages exchanged with other DUNE tasks.

5.2.5 Implementation

The algorithms presented in this section were deployed on a Light Autonomous Underwater Vehicle (LAUV) with the help of the open-source software toolchain developed by LSTS [119]. The toolchain includes DUNE and Neptus, which communicate with the IMC message protocol [120], as depicted in Figure 5.7.

Neptus is a graphical user interface supporting planning and execution control [121]. The LAUV communicates with Neptus via acoustic modems (underwater) or Wi-Fi (at the surface). DUNE is the LAUV onboard software handling navigation, control, communications, logging, and vehicle hardware interactions. DUNE provides tasks able to interact with all sensors and actuators of the LAUV, as well as the software infrastructure to deploy additional control and navigation tasks. A new control task was developed to deploy the front-tracking algorithms and the front-estimation calculations. This task communicates with other tasks using four IMC messages, as described in Figure 5.8.

The low-level control of the vehicle is done with the help of two other control tasks, also called maneuver controllers: YoYo and GoTo. The new control

Set	Front				Ocean Currents	# of tests
	Static	Dynamic	2D	3D		
A	✓		✓			9
B.1		✓	✓			9
B.2		✓	✓			16
C.1		✓		✓		9
C.2	✓			✓		30
D	✓	✓	✓	✓		216
E		✓		✓	✓	18
						307

Table 5.3: Test plan.

task sends one message to each maneuver controller. The GoTo maneuver will control the vehicle to move to a given waypoint, and YoYo maneuver is about implementing a yoyo trajectory. The new control task consumes messages encoding the estimated state and salinity measurements periodically broadcast by the two other tasks depicted in Figure 5.8. DUNE has a simulation mode to facilitate the integration and validation of new onboard software with the help of models of the sensors and actuators. This is why the two tasks sending messages to the new control task can get data from simulation models and/or recorded files or from the hardware in field deployments. This mode validated the algorithms developed in this work before field deployments.

5.3 Simulation Results

We developed a simulation environment in MATLAB to test and validate the proposed approach. We used the DELFT3D ocean model [122] to generate salinity and water flow data in these simulations.

5.3.1 Test plan

The algorithms were tested in three main front shapes determined by different wind conditions: Southerly, Northerly, and Light Wind. Typically, Southerly winds result in faster propagation velocities for the front.

Table 5.3 describes 307 test cases grouped into seven different test sets, **A** to **D**, covering different front conditions and parameters. Front conditions included: i) static or dynamic fronts, ii) 2D or 3D tracking, and iii) the presence of ocean currents. Static fronts do not represent a realistic test case. Nevertheless, these tests proved useful in studying the properties of the three algorithms. Test sets **A** and **B.1** included nine tests, each analyzing the three algorithms for the three different front patterns. **B.2** analyzes the effect of the AUV velocity v (ranging between 0.5 to 5 m/s). **C.2** considers the effect of the length of the yoyo cycle (distances between consecutive samples at the surface) ranging from 3.7 m to 458.1 m. **D** tests different values of the trajectory parameters ($\alpha \in [20, 70]^\circ$, $D_{out} \in [100, 800]m$ and $D_{in} \in [100, 800]m$). Figure 5.9 shows simulation runs for test set **C.1** with a S front. As we can see, the AUV successfully tracks the front with the three algorithms. Also, note that this is the fastest propagating front and that the algorithms can still track the front.

Next, we present the performance metrics used in these simulations before briefly discussing the results for the seven test sets.

Performance metrics

We consider three performance metrics:

1) The quality of each mission is measured by the average error of the front prediction ϵ_{avg} presented below, where i refers to the error in each transect. We also label each transect as optimal if $|\epsilon_i| < 20^\circ$.

$$\epsilon_{avg} = \frac{1}{n} \sum_{i=1}^n |\epsilon_i|, \quad (5.9)$$

2) The speed of progression along the front v_{fp} (m/s) is given by the length of the front mapped so far (calculated from successive front crossings) divided by the duration of the mission t_M :

$$v_{fp} = \frac{1}{t_M} \sum_{i=2}^n \|\vec{P}_{cross_i} - \vec{P}_{cross_{i-1}}\|. \quad (5.10)$$

3) The percentage of optimal (perpendicular) crossings for a given mission.

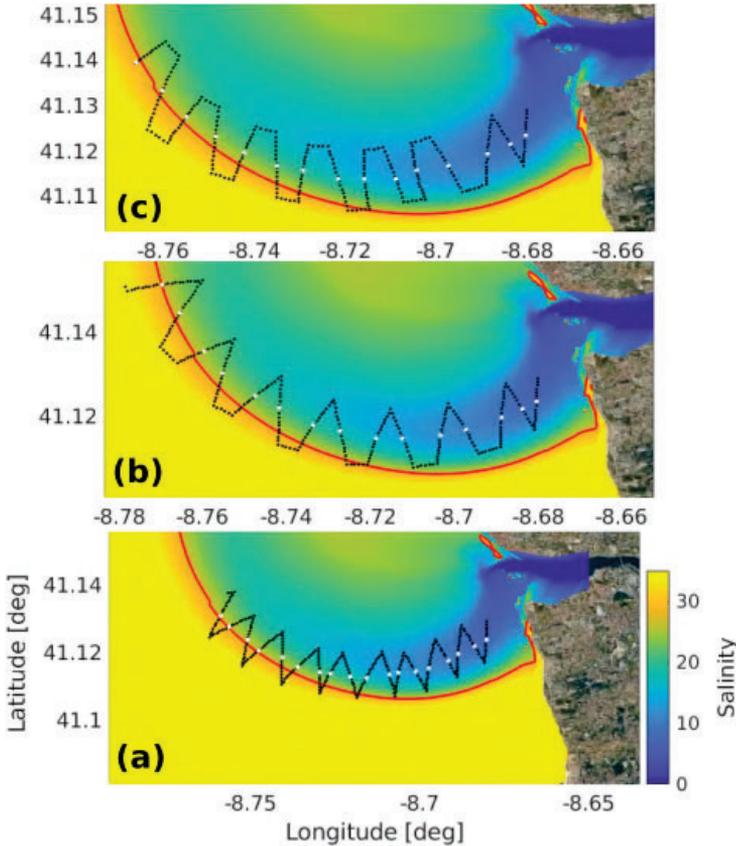


Figure 5.9: Results from set **C.1** 6-hour mission from 4:30 to 10:30 on the *S* front. Algorithms: (a) *Zigzag*, (b) *Alt. Zigzag*, (c) *Lawn Mower*. The vehicle (black) moves in the surface along the front (red).

5.3.2 Test sets A, B.1 and C.1: Performance of the algorithms

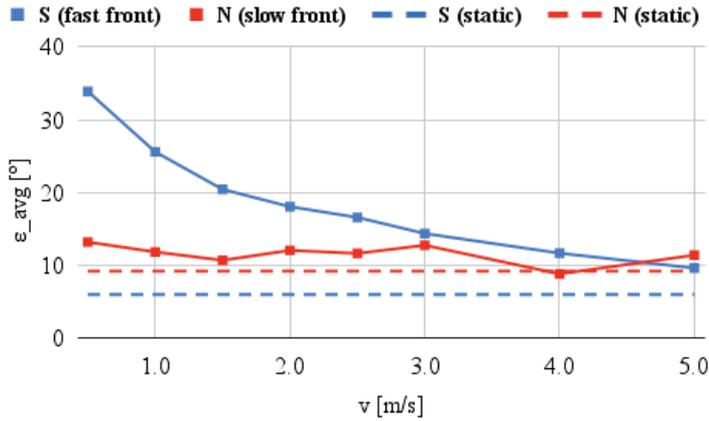
Test sets **A**, **B.1**, and **C.1** targeted a comparative study of the algorithms under the same front patterns and tuning parameters. The performance ranking of the algorithms under these conditions is briefly summarised in Table 5.4.

5.3.3 Test set B.2: Effects of the AUV velocity

Test set **B.2** aimed at studying how vehicle velocity impacts tracking performance. This was done for the fast *S* and slow *N* propagating fronts. The results

	<i>Zigzag</i>	<i>Alt. Zigzag</i>	<i>Lawn Mower</i>
Estimated average error	1	2	1
# of optimal crossings	2	3	1
Front progression	2	1	2

Table 5.4: Rankings of the algorithms: worst (3) to best (1).

Figure 5.10: Average error, ϵ_{avg} , for different vehicle velocities, v .

are shown in Figure 5.10.

Observe that some AUV velocities used in these tests are not feasible. However, these results provide insights into the role of the vehicle's speed in the tracking mission. Starting with the fast front (*S front*) represented in blue, we observe that the increase in velocity produces a better front prediction that converges to the performance on the static front (dashed line). As expected, the increase in the AUV velocity correlates to an increase in performance for fast-moving fronts; this effect is almost negligible for slow-moving fronts.

5.3.4 Test set C.2: Effects of the length of yoyo trajectory cycle

Test set C.2 aims to study the effects of the length of the yoyo cycle on performance. The vehicle samples the front's surface one time per yoyo cycle. For example, the distance between consecutive samples at the surface is $d_{surface} = 74.6$ m for a pitch angle of $\theta = 15^\circ$ and with $z_{max} = 10$ m. As expected, a

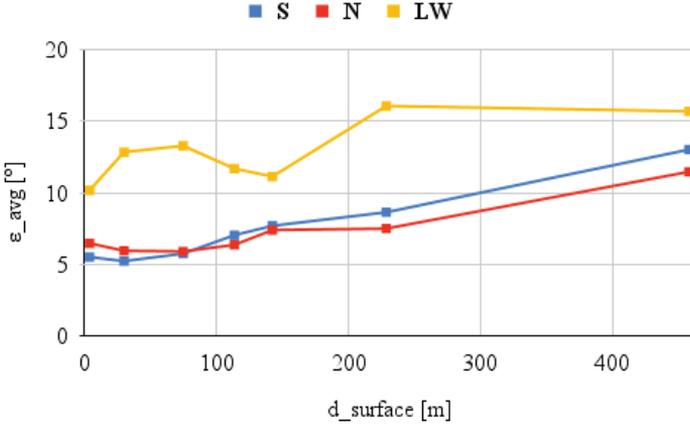


Figure 5.11: Effect of the distance between surface points $d_{surface}$ on the average error ϵ_{avg} for a *Zigzag* trajectory.

general upward tendency in ϵ_{avg} is verified with the increase of $d_{surface}$ (Figure 5.11). However, it is important to note that this trend is not noticeable when $d_{surface} < 100m$. This provides good evidence that in this range, the 3D mapping of the front does not heavily influence the tracking performance. One would expect the vehicle to be unable to track the front for larger values of $d_{surface}$ (e.g., 400m). Surprisingly, this was not the case for tests with a static front.

5.3.5 Test set D: Effects of different trajectory parameters

Test set **D** evaluated the effects of variations of the α trajectory parameter for static 2D and dynamic 3D fronts.

In the static 2D tests, larger α angles (and larger D_{in} and D_{out} distances) result in larger distances between crossing points, thus reducing tracking performance (measured by average error variation on α) as shown in Figure 5.12a for the *Zigzag* algorithm. Similar qualitative results were obtained for the other parameters and strategies.

In the dynamic 3D tests, the close-to-linear relations obtained for the static 2D tests are no longer valid. The best performance is not achieved when $\alpha = 20^\circ$ but when $\alpha \in [30, 50]^\circ$ (Figure 5.12b). Thus, the best performance is not guaranteed when the front crossing points are at a minimal distance from each other. The locations of P_{cross} are not as accurate as before, and the prediction

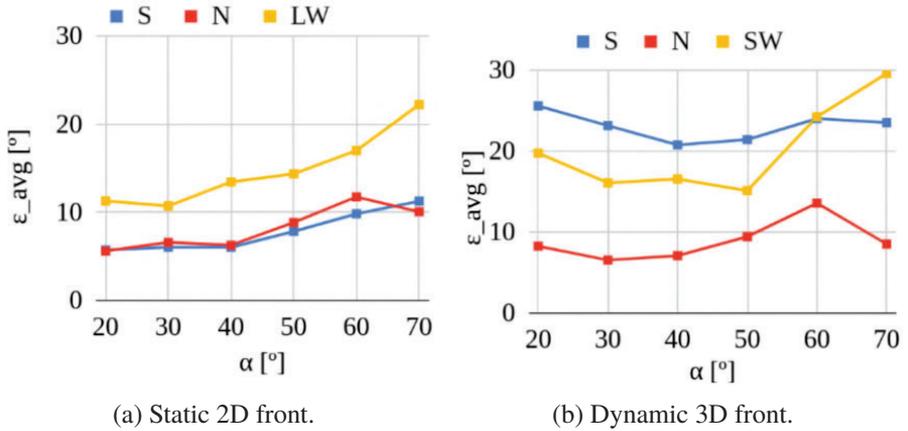


Figure 5.12: Effect of the trajectory angle, α , on the average error, ϵ_{avg} for a Zigzag strategy.

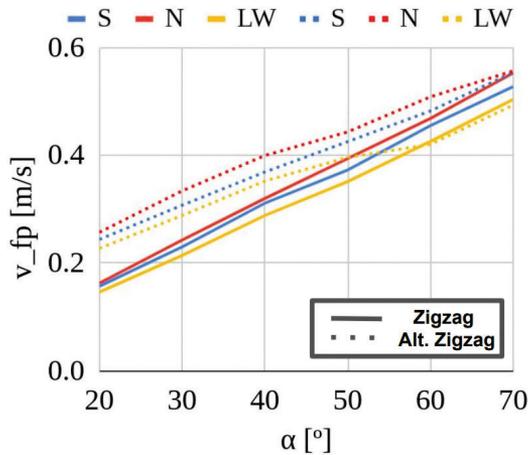


Figure 5.13: Effect of the trajectory angle α on the speed of progression v_{fp} .

with closer points does not always result in a more accurate description of the front's orientation.

The effects on the speed of progression v_{fp} are as expected for all test cases. The speed of progression increases with larger distances between crossing points, which, in turn, increases with α (Figure 5.13).

front	<i>Zigzag</i>			<i>Alt. Zigzag</i>			<i>Lawn Mower</i>		
	OC	PC	Δ	OC	PC	Δ	OC	PC	Δ
N	60	69	9	101	70	-31	8	-7	-15
S	37	28	-9	40	35	-5	-3	-9	-6
LW	37	-5	-42	3	-39	-42	-12	-36	-24

Table 5.5: Variation of the average estimation error, ϵ_{avg} , compared to the test where ocean currents are zero. The values are in percentage. *OC* - Only ocean currents are taken into account; *PC* - Path correction method active; Δ - Difference between the two results.

Test set E: effects of ocean currents

The motions of the AUV are affected by ocean currents (see Equation 5.3). Moreover, the AUV used in this work does not have sensors to measure the velocity with respect to the seabed. This affects the trajectory tracking performance. To minimize this effect, we used the path correction method (provided by DUNE) when the AUV reaches the surface: i) the position of the AUV is reset with the GPS coordinates when the AUV breaches the surface; ii) the AUV is commanded to move to the real surfacing position before diving again.

Test set **E** includes runs without and with the path correction method. Results are compared to the ones from test set **C.1**. Table 5.5 shows the effect of the ocean currents on the estimated average error. As expected, ocean currents have an impact on the tracking performance. Overall, the path correction method mitigates these effects. However, it does not completely correct the trajectory of the vehicle. In some cases, it does not even improve performance. In any case, these tests show that the effects of ocean currents do not compromise front tracking. Figure 5.14 shows one run with the same parameters and front conditions used in the run from Figure 5.9, but taking into account the effects of the water velocity. This simulation is the most realistic of the overall test plan. The maps built from the sampled salinity data characterize the front in the band of interest. Figure 5.15 shows the salinity map built with the data collected during the mission represented in Figure 5.14. Transitions from the front to ocean waters are easily identified in this map.

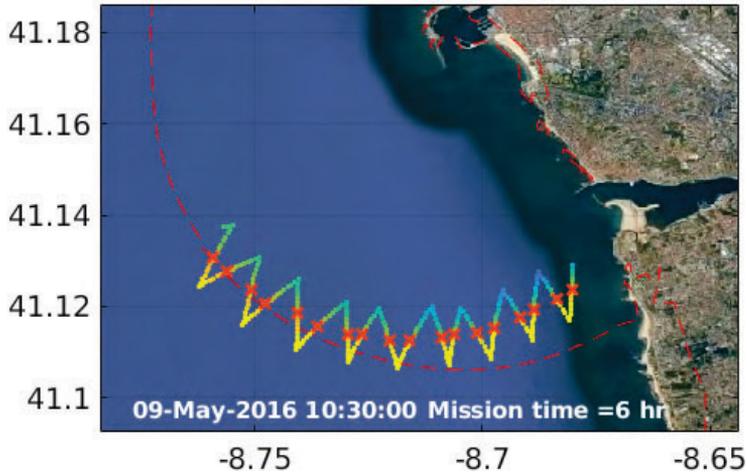


Figure 5.14: Simulation run from simulation set **E**. *Zigzag* algorithm in the presence of ocean currents (satellite image of the Douro estuary as background).

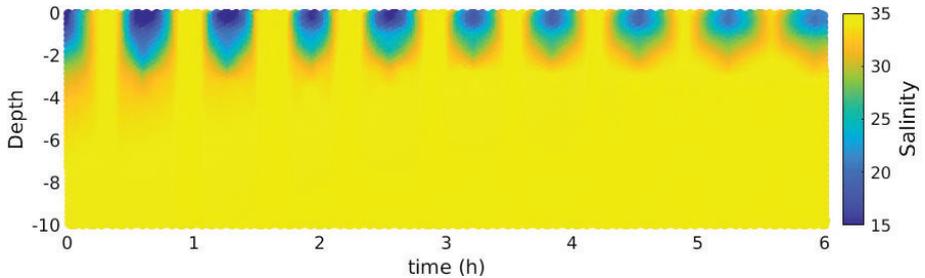


Figure 5.15: Salinity map obtained from the data collected in the test represented in Figure 5.14.

5.4 Summary

In this chapter, we considered the problem of tracking and mapping a river front with an AUV and solved it with the help of novel trajectory adaptation algorithms. The approach was evaluated and tested in simulation using a high-resolution hydrodynamic model of the front. The test plan included 307 test cases covering the most representative environmental conditions. This was an interdisciplinary endeavor at the intersection of robotics and oceanography. The focus was not only on robotic exploration but also on the scientific study of the

front. The test plan shed some light on the structure of the front propagation, evolving from an initial sharp variation of salinity to a situation in which the front has two sharp edges separated by a plateau, thus presenting an added difficulty to some tracking algorithms. This observation also provided insights into trajectory adaptation and, more importantly, into selecting the initial location for the AUV. Space limitations preclude a thorough discussion of the results, but our parameterizable approach and the study of the conditions for parameter selection will be at the heart of a decision support system for optimized front studies. Future work will also use machine learning techniques to optimize tracking and mapping campaigns. Extensions to multi-agent settings are also being considered.

Chapter 6

Least Squares Front Estimation for Single-agent Systems

In this chapter, we consider the problem of tracking moving algal bloom fronts using an AUV equipped with a sensor that measures the concentration of chlorophyll a . Chlorophyll a is a green pigment found in plants, and its concentration indicates phytoplankton abundance. Our algal bloom front-tracking mission consists of three stages: deployment, data collection, and front tracking. At the deployment stage, a satellite collects an image of the sea from which the location of the front, the reference value for the concentration at this front, and, consequently, the appropriate initial position for the AUV is determined. At the data collection stage, the AUV collects data points to estimate the local algal gradient as it crosses the front. Finally, at the front tracking stage, an adaptive algorithm based on recursive least squares fitting using recent past sensor measures is executed. We evaluate the algorithm's performance and sensitivity to measurement noise through MATLAB simulations. We also present an implementation of the algorithm on a realistic software platform for marine robots and validate it using simulations with satellite model forecasts from Baltic sea data.

This chapter is organized as follows. In Section 6.1, the main problem is formulated, and we give an overview of the components included in the experimental setup. We explain the front tracking algorithm in Section 6.2. In Section 6.3, we describe the implementation of the algorithm and simulations using satellite data of chlorophyll a concentration. Concluding remarks and future directions follow in Section 6.4.

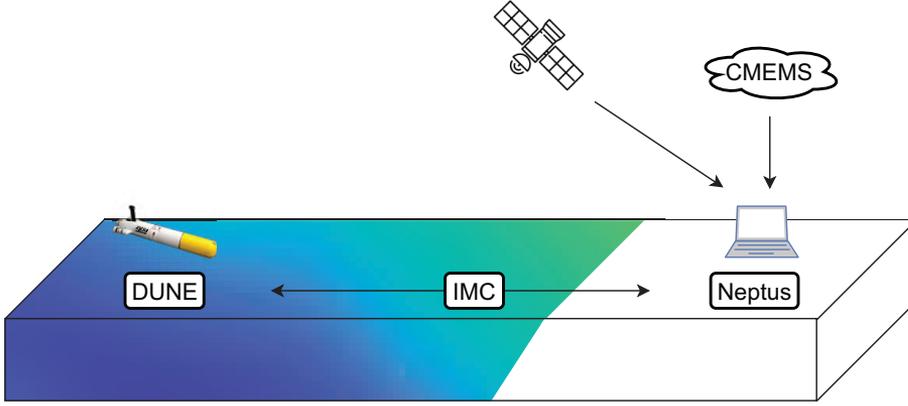


Figure 6.1: Experimental setup including the AUV, satellite, DUNE, IMC, and Neptus.

6.1 Problem Formulation

We consider the problem of detecting and tracking irregular, moving, and time-varying algal bloom fronts. We propose a solution that consists of an experimental setup as in Fig. 6.1, composed of an AUV with a chlorophyll a concentration sensor, a control and detection algorithm implemented in the software platform DUNE [123], a message protocol implemented in IMC [124], a visualization tool implemented on Neptus [125], satellite data from the previous day, and the CMEMS simulated chlorophyll a data of the region from the past months [126].

6.1.1 Algal blooms

In Fig. 6.2, we plotted two time instances of a forecasted chlorophyll a concentration field, part of the *Baltic Sea biogeochemistry analysis and forecast* product [126]. The spatial resolution is 2 km by 2 km, the time resolution is hourly, and we selected data from the east coast of Sweden, near Stockholm, from February 2020. The range from 0 (dark blue) to 1 (yellow) indicates the chlorophyll a concentration. The white areas represent land and correspond to the archipelago near Stockholm, Sweden.

We define a front (red) as a level set of a time-varying scalar field $\delta : \mathbb{R} \times \mathbb{R}^2 \rightarrow \mathbb{R}$:

$$F(t) = \{\mathbf{p} \in \mathbb{R}^2 : \delta(t, \mathbf{p}) = \delta_{\text{ref}}\}, \quad (6.1)$$

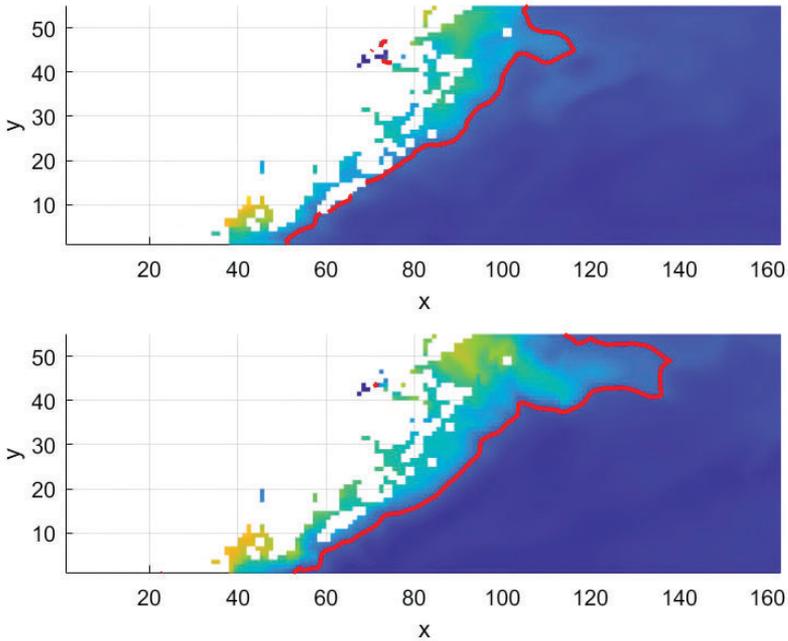


Figure 6.2: CMEMS simulation data of chlorophyll a in the Baltic Sea, at different time instances, with higher concentration (yellow), lower concentration (blue), land (white), and the front $F(t)$ (red line).

where δ_{ref} is some reference value, \mathbf{p} the position and t time.

The reference δ_{ref} is chosen according to the latest satellite data of the location before the mission starts. It can be seen as a calibration of the AUV to different algal bloom situations.

6.1.2 Experimental setup

The components of the experimental setup are described next.

Satellite data – collected in the experiment area is used to initialize the controller, including the reference value and initial gradient estimate. CMEMS provides forecasts used for simulations before mission execution.

The **AUV** – has a sensor to measure the concentration of chlorophyll a . See the UAV SAM from SMaRC (Swedish Maritime Robotics Centre, KTH [127]) with the Total Algae sensor from YSI [128] in Fig. 6.3.



Figure 6.3: Top: SAM UAV from SMaRC. Bottom: Total Algae sensor from YSI

The **AUV onboard software** contains the sensing, communications, navigation, and control software used during the operation. We used the LSTS toolchain containing DUNE, IMC, and Neptus for the implementation. For a detailed overview of the toolchain’s components and capabilities, see [119]. DUNE is a platform- and architecture-independent runtime environment for the robot’s onboard computer. It provides a simple and unified programming interface for writing embedded software components for marine robotics, such as navigation filters, controllers, or sensor drivers. Each software component is represented as a DUNE Task, an isolated code section executed in its operating system thread. DUNE tasks communicate exclusively using IMC messages exchanged through a global shared message bus. Tasks can expose parameters (e.g., controller gains) that may be set in plain text configuration files and changed on the fly in the Neptus’ operator console. DUNE contains an implementation of a navigation and control suite for the AUV and a detailed full-order AUV simulator. The simulator is used as a drop-in replacement of the sensor and actuator drivers, which would interact with the real vehicle hardware, allowing us to simulate the same code that will later be deployed on the real robot.

The **mission control software** monitors the system’s position and operating state during the mission and retrieves collected data from the vehicle’s storage. Neptus is a command and control application software providing a configurable and extensible graphical interface for mission planning, simulation, control, and review analysis.

Finally, the **mission control systems** are comprised of all the operators, support staff, and systems involved in the mission. This may include research vessels or other manned or unmanned systems used to deploy and recover the AUV.

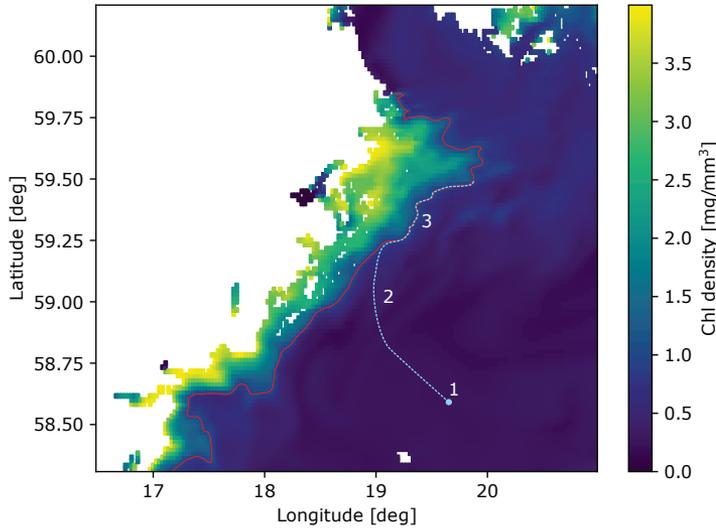


Figure 6.4: Illustration of the three stages of the front tracking algorithm.

6.1.3 Problem

The problem considered in this chapter is how to track irregular and dynamic algal bloom fronts using the described experimental setup. The solution is a front-tracking algorithm consisting of a control law and a gradient estimator, as presented in the next Section.

6.2 Front Tracking Algorithm

Given the abovementioned problem, we present a solution split into three stages, as illustrated in Fig 6.4. The first stage (indicated by 1 in the figure) is initialization and deployment aided by the satellite or forecast data, the second one (2) is finding and approaching the front, and the final one (3) is persistent front tracking.

The initialization stage selects the chlorophyll a reference value δ_{ref} and the vehicle's initial position and heading. We assume that suitable values for these parameters can be obtained by examining satellite or forecast data corresponding to a point in time sufficiently close to the mission start time.

The front finding and approaching stage are led by the control law, which

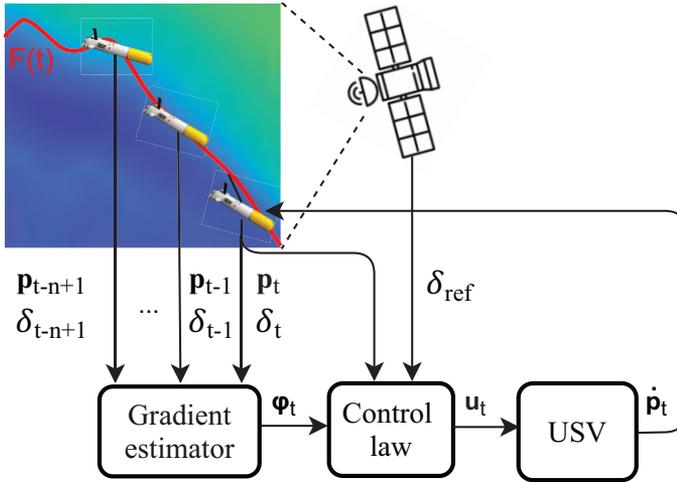


Figure 6.5: AUV control architecture

gives a velocity reference as a function of the measured chlorophyll a concentration at the vehicle's position. This velocity reference will lead the vehicle to the front location when it is away from the front and make it travel along the front when it is close. An essential ingredient in the control law is the concentration gradient, estimated from the measurements taken by the vehicle. The estimator computes an approximate gradient value at the vehicle's location using a local linear approximation of the chlorophyll a concentration. The final stage is persistent front tracking, which consists of keeping the AUV in and around the front for the mission's duration.

We summarize the overall architecture of the front-tracking algorithm in Fig. 6.5. In the remaining subsections, we describe the control law, the gradient estimator, and the AUV model.

6.2.1 Control Law

Assume for the moment that the reference value δ_{ref} is known and that the AUV is holonomic, that is, the dynamics are given by

$$\dot{\mathbf{p}} = \mathbf{u},$$

where $\mathbf{p} = (x, y)$ is the vehicle's position and \mathbf{u} is the velocity control. We define the feedback velocity law as:

$$\begin{aligned}\mathbf{u}(t, \mathbf{p}) &= \alpha_{\text{seek}} \mathbf{u}_{\text{seek}}(t, \mathbf{p}) + \alpha_{\text{follow}} \mathbf{u}_{\text{follow}}(t, \mathbf{p}) \\ \mathbf{u}_{\text{seek}}(t, \mathbf{p}) &= -(\delta(t, \mathbf{p}) - \delta_{\text{ref}}) \nabla \delta(t, \mathbf{p}) \\ \mathbf{u}_{\text{follow}}(t, \mathbf{p}) &= R_{\pi/2} \nabla \delta(t, \mathbf{p}),\end{aligned}\tag{6.2}$$

where $\nabla \delta$ is the gradient of δ with respect to \mathbf{p} and $R_{\pi/2}$ is a mapping which rotates vectors by 90 degrees. This feedback law has two components: the \mathbf{u}_{seek} component controls the vehicle to the level set of δ corresponding to the front by following the gradient vector field, while $\mathbf{u}_{\text{follow}}$ makes the vehicle travel along the front. The direction in which the vehicle travels along the front after having reached it is determined by the orientation of $R_{\pi/2}$.

It can be seen that if the front F (in (6.1)) is static (i.e., $\partial \delta / \partial t \equiv 0$) then the feedback law \mathbf{u} achieves convergence of the vehicle's position to the front (i.e., $\delta(t, \mathbf{p}(t)) \rightarrow \delta_{\text{ref}}$) as long as $\nabla \delta \neq 0$ so that the vehicle does not get stuck in a critical point of δ . When the front is not static, there is no such guarantee. We assume that the vehicle can move and take measurements at a time scale much faster than that at which the chlorophyll a field is changing so that we can view the time variation of δ as a perturbation.

6.2.2 Gradient estimator

In order to realize the control law (6.2), the gradient $\psi(t) := \nabla \delta(t, \mathbf{p}(t))$ is needed. The vehicle takes noisy measurements of the concentration at discrete instants of time:

$$y_k = \delta(t_k, \mathbf{p}(t_k)) + \epsilon_k,$$

where t_k are the measurement times and ϵ_k is the measurement noise. We assume the position of the vehicle at the measurement times, $\mathbf{p}_k := \mathbf{p}(t_k)$ is perfectly known. We can then define the data available to the vehicle at time $t \in [t_k, t_{k+1})$ as

$$\mathcal{D}(t) = ((\mathbf{p}_0, y_0), (\mathbf{p}_1, y_1), \dots, (\mathbf{p}_k, y_k)).$$

The gradient estimation problem is then to construct an estimate $\hat{\psi}(t)$ of $\psi(t)$ based on $\mathcal{D}(t)$.

We propose to construct such an estimate as follows. Let t_k be the time of the most recent measurement, and take n such that the set of measurements

$$\mathcal{D}_n(t) = ((\mathbf{p}_{k-n+1}, y_{k-n+1}), \dots, (\mathbf{p}_k, y_k))$$

satisfies the following conditions:

- the measurements are taken sufficiently close together in time so that the concentration is approximately constant on $[t_{k-n+1}, t_k]$;
- the measurements are taken close together in space.

These assumptions allow us to replace δ by its first-order Taylor approximation on a set containing the measurement positions:

$$\delta(t, \mathbf{p}) \approx \delta(t^*, \mathbf{p}^*) + \nabla \delta(t^*, \mathbf{p}^*) \cdot (\mathbf{p} - \mathbf{p}^*),$$

where \mathbf{p}^* is some position in this set and $t^* \in [t_{k-n+1}, t_k]$ is some time instant. We define

$$\begin{aligned} \hat{\psi} &= \nabla \delta(t^*, \mathbf{p}^*) \\ \delta_0 &= \delta(t^*, \mathbf{p}^*) - \nabla \delta(t^*, \mathbf{p}^*) \cdot \mathbf{p}^*, \end{aligned}$$

so that

$$\delta(t, \mathbf{p}) \approx \delta_0 + \hat{\psi} \cdot \mathbf{p}.$$

Applying this equation to the n measurements in $\mathcal{D}_n(t)$, we get a set of equations which are linear in δ_0 and $\hat{\psi}$, which can be solved with standard least squares methods. An alternative is to use recursive least squares with exponential forgetting.

6.2.3 AUV model

We adopt a typical 3-degree of freedom (surge, sway, and yaw) model for the AUV [129]. It is represented as

$$\begin{aligned} \dot{\boldsymbol{\eta}} &= \mathbf{R}(\psi) \mathbf{v} \\ M \dot{\mathbf{v}} + \mathbf{C}(\mathbf{v}) \mathbf{v} + \mathbf{D}(\mathbf{v}) \mathbf{v} &= \boldsymbol{\tau}, \end{aligned} \tag{6.3}$$

where $\boldsymbol{\eta} = [x, y, \psi]^T$ is the cartesian position $[x, y]$ and angle ψ , $\mathbf{v} = [u, v, r]^T$ are the velocities, and $\mathbf{R}(\psi) := \mathbf{R}_{z,\psi}$ is the rotational matrix.

6.3 Simulation Results

6.3.1 MATLAB simulations

In this subsection, we approximate the kinematics of the AUV by a single integrator, $\dot{\mathbf{p}} = \mathbf{u}$. We simulated a 20 hour mission, having the length of saved data $\mathcal{D}_n(t)$ as $n = 20$ and the algal front concentration reference as $\delta_{\text{ref}} = 2$. Having measurements taken with a period of 3 minutes, we choose to start the gradient estimation after the first hour. We set the parameters $\alpha_{\text{seek}} = 6$ and $\alpha_{\text{follow}} = 2$ of the control law $\mathbf{u}(t)$. We deploy the AUV at $\mathbf{p}(0) = [65, 1]$ with an initial algal gradient estimate of $\hat{\boldsymbol{\psi}}(0) = [1, -1]$. We introduce a measurement error in the chlorophyll a sensor of the form $\delta_{\text{measured}}(t) = \delta_{\text{real}}(t) + \delta_{\text{noise}}(t)$ with maximum noise of about 0.4.

Fig. 6.6 shows five instances of the algal front tracking mission. The first shows the initial position at which the AUV was deployed and its convergence towards the algal front $F(t)$. The following figures indicate a constant and accurate tracking of the algal bloom front. Here, we can also see the gradient estimator $\hat{\boldsymbol{\psi}}$ indicating a fair estimation of the normal vector to the algal front on the AUV's position.

Tracking and estimation errors are depicted in Fig. 6.7. The first figure contains three chlorophyll a concentration values: the measurement $\delta_{\text{measured}}(t)$, the real chlorophyll a concentration $\delta(t)$, and the reference δ_{ref} . The oscillation around the reference value represents the deviation to the front. Here, notice two things: first, the measured and real values have a difference corresponding to the sensor noise; second, the AUV starting point is far from the front, but it then oscillates around the reference value of chlorophyll a concentration.

The second figure illustrates the distance of the AUV to the closest point of the front $F(t)$: $\inf_{f \in F(t)} \|f - \mathbf{p}(t)\|$. Here, the initial distance is large as the AUV starts far from the front. Then, the distance oscillates around zero, indicating a zig-zag motion around the front. The last figure depicts the angle of the estimated gradient, $\angle \boldsymbol{\psi}(t)$. By plotting the angle, we can evaluate the variation of directions towards the front $F(t)$.

6.3.2 DUNE controller implementation

In the DUNE control implementation, a two-step waypoint generation scheme is used, where the robot performs a 'zig-zag' motion of amplitude θ and horizontal displacement d around a mean bearing angle $\psi_{\mathbf{u}}$. This is depicted in Fig. 6.8.

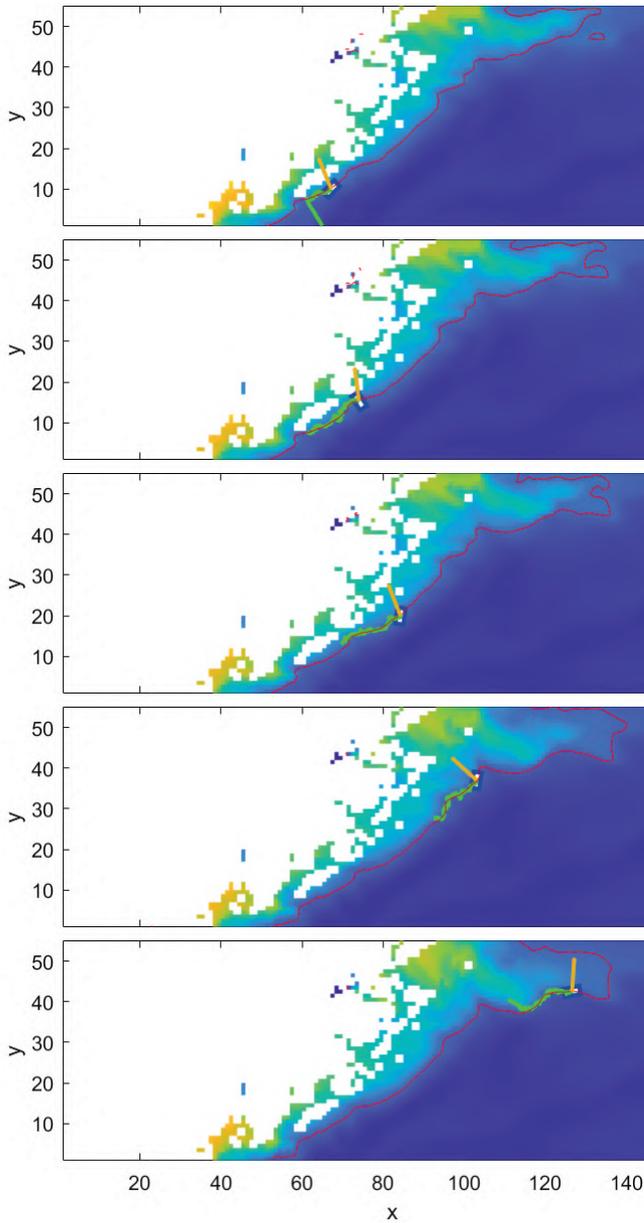


Figure 6.6: Time-lapse of the AUV (blue rectangle) and its gradient estimator ψ (orange arrow) tracking the algal front (red contour) with a representation of the AUV's path (green).

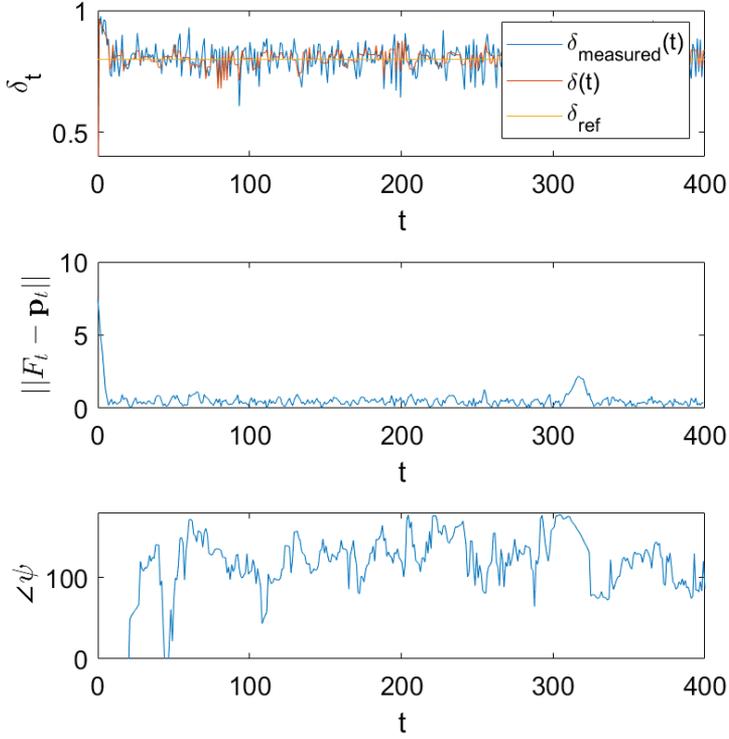


Figure 6.7: First plot: measured chlorophyll a concentration $\delta_{\text{measured}}(t)$, local chlorophyll a concentration $\delta(t)$, and reference chlorophyll a concentration δ_{ref} . Second plot: distance from the AUV to the closest point in the front, $\|F(t) - \mathbf{p}(t)\|$. Third plot: angle of the estimated gradient, $\angle\psi(t)$.

After the vehicle reaches the second waypoint, the measurements collected during the motion are used to estimate the gradient as described above, and a new bearing reference $\psi_{\mathbf{u}}$ is computed using (6.2). The vehicle travels at a constant speed so that only the relative size of α_{seek} and α_{follow} in (6.2) is relevant, and we fix $\alpha_{\text{follow}} = 1$. The value of δ used in the computation of \mathbf{u} is the most recent sample.

As above, the gradient estimate $\hat{\psi}$ is initialized with a given value. Gradient estimation is performed only after the average concentration is within a threshold error δ_{thr} of δ_{ref} . Thus, initially, the vehicle travels in a straight line (i.e., $d, \theta = 0$), and a fixed track distance is used to compute the next waypoint, with

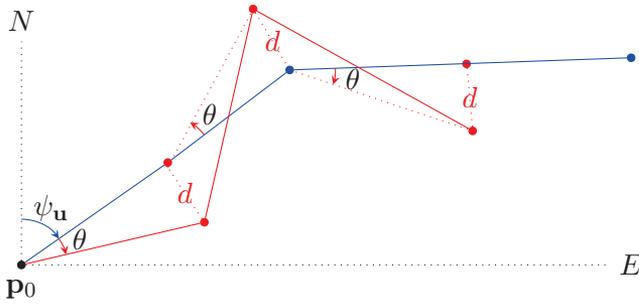


Figure 6.8: Waypoint generation scheme.

the track bearing given by (6.2).

The chlorophyll a concentration field is simulated using numerical data on a regular time-latitude-longitude grid. A DUNE task simulating the chlorophyll a concentration sensor reads messages containing the vehicle's current position, linearly interpolates the numerical data to the vehicle position and current time, and dispatches an IMC message containing the current concentration value. The controller task then reads these messages, which stores them together with the corresponding vehicle position.

6.3.3 DUNE simulation results

α_{follow}	d	θ	Speed	δ_{ref}	δ_{thr}
25	250 m	45 deg.	5 m/s	1 mg/m ³	0.1 mg/m ³

Table 6.1: Controller parameters used in the DUNE simulation.

We simulated an approximately 32 hour mission with the controller parameters shown in Table 6.1. The vehicle samples the chlorophyll a concentration at its position every 3 seconds with Gaussian measurement noise of variance 0.001 mg/m³. Fig. 6.9 shows the vehicle's position and the chlorophyll a concentration field at two instants of time. Blue regions indicate low concentration values at the corresponding position, while green regions indicate high concentration values. The red curves represent the front $F(t)$ at the corresponding instant. After the initial approach phase, the vehicle successfully tracks the time-varying front.

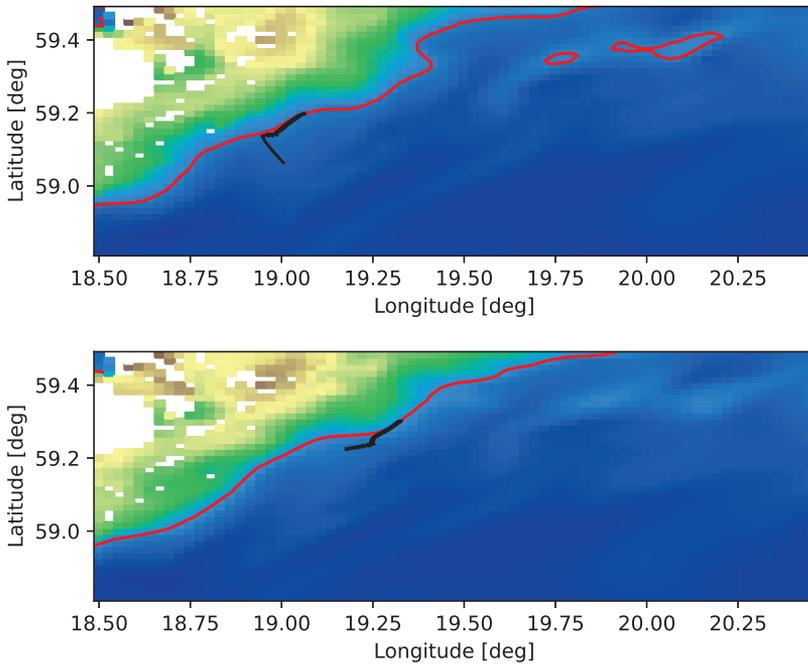


Figure 6.9: Two time instants ($t = 10$ h and 25 h) of the mission. The black line shows the AUV path over the preceding 10 hours.

Fig. 6.10 shows the chlorophyll a concentration measured by the vehicle, normalized to the maximum value contained in the data. The shaded blue area indicates the value of δ_{thr} . One can see that the vehicle loses track of the front at around $t = 18$ h. This is because the chlorophyll a field changes significantly between $t = 18$ h and $t = 19$ h, so the assumptions considered in the estimator design are no longer valid. After this sudden change, the vehicle recovers the front and tracks it successfully again, showing that the algorithm is robust to temporary assumption violations.

6.4 Summary

In this chapter, we considered the problem of algal bloom front tracking using a sensing AUV. We assumed the AUV has a GPS receiver that reports its position and a chlorophyll a concentration sensor, which measures the local algal

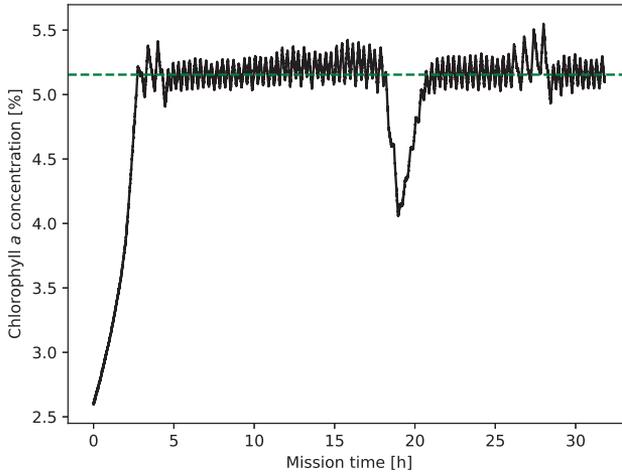


Figure 6.10: Chlorophyll *a* concentration measured by the vehicle.

concentration. We proposed an experimental setup composed of satellite data, AUV hardware, software, CMEMS forecasted data, and an algorithm in which the AUV estimates the local algal gradient using recent past measurements and least squares fitting. We provided a MATLAB simulation and analyzed convergence given sensor noise. The AUV converged and moved along the detected algal bloom front for the mission's duration. The algorithm was also implemented on the LSTS Toolchain using model forecasts of the chlorophyll *a* concentration in the Baltic Sea in February 2020 from CMEMS. Our plan is to do tests on the Baltic Sea using our algorithm implemented on DUNE and algal bloom forecasting using satellite data of different water properties, such as salinity, temperature, and water currents.

Chapter 7

Gaussian Process Front Estimation for Single-agent Systems

This chapter investigates using satellite data to improve adaptive sampling missions for front tracking. Our system finds and tracks algal bloom fronts using an Autonomous Underwater Vehicle (AUV) equipped with a chlorophyll a concentration sensor and satellite data. The proposed method learns the kernel parameters for a Gaussian process (GP) model using satellite images of chlorophyll a from the previous days. Then, using online data collected by the AUV, it takes the gradient of the concentration to obtain the direction of the algal bloom front. The AUV tracks the front using a novel gradient estimator and motion controller. The performance of this method is evaluated through realistic simulations for an algal bloom front in the Baltic sea. We compare the performance of different estimation methods. A sensitivity analysis is performed to evaluate the impact of sensor noise.

This chapter is organized as follows. In Section 7.1, the main problem is formulated, and we give an overview of the components included in the setup. In Section 7.2, we introduce the proposed front tracking algorithm. This includes the high-level system architecture, the dataset we use, the GP model for the chlorophyll a concentration in the Baltic sea, and the path planning guidance law. In Section 7.3, we provide results from realistic simulations, and in Section 7.4, a sensitivity analysis on the impact of sensor noise on algorithm performance, comparing and evaluating different gradient estimation methods. Concluding remarks, discussion, and future directions follow in Section 7.5.

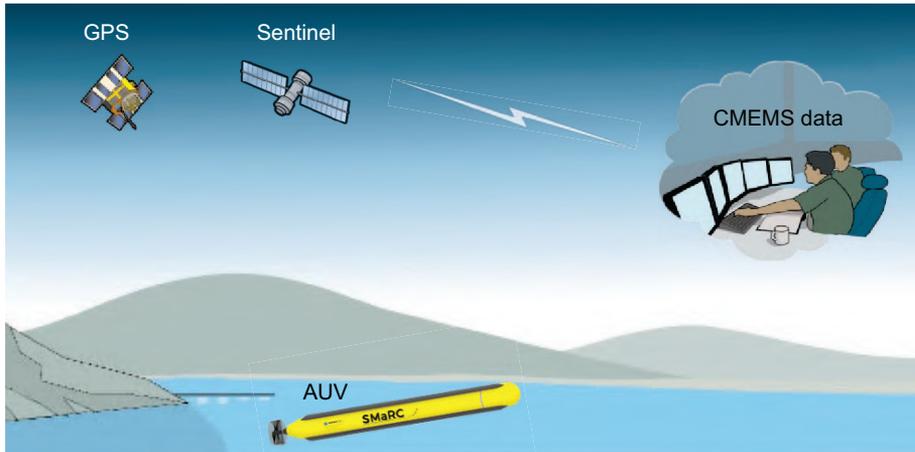


Figure 7.1: System overview, including the AUV, the GPS, the Sentinel satellite, and the CMEMS data.

7.1 Problem Formulation

This chapter proposes an approach for adaptive sampling of algal bloom fronts using an AUV informed by satellite imagery. The proposed algorithm enables tracking multiple water parameters, including chlorophyll *a* concentrations, turbidity, dissolved oxygen, and salinity. Out of these broad sampling possibilities, we are particularly interested in chlorophyll fronts due to their connection with harmful algal blooms (HABs) in the Baltic [130]. In Fig. 7.1, we illustrate this cyber-physical system, consisting of the AUV, the GPS signal that it uses for localization, the Sentinel satellite, which provides raw imagery of the region, and the Copernicus Marine Environment Monitoring Service (CMEMS), which re-analyses the imagery from the Sentinel satellite into more accurate datasets that can be used to inform our AUV.

7.2 Front Tracking Algorithm

This chapter considers algal bloom front tracking as an adaptive environmental sampling problem. Our algal bloom front tracking aims to find and track a front with limited global information on its location and shape but to use local information collected by the AUV as it moves to explore the environment.

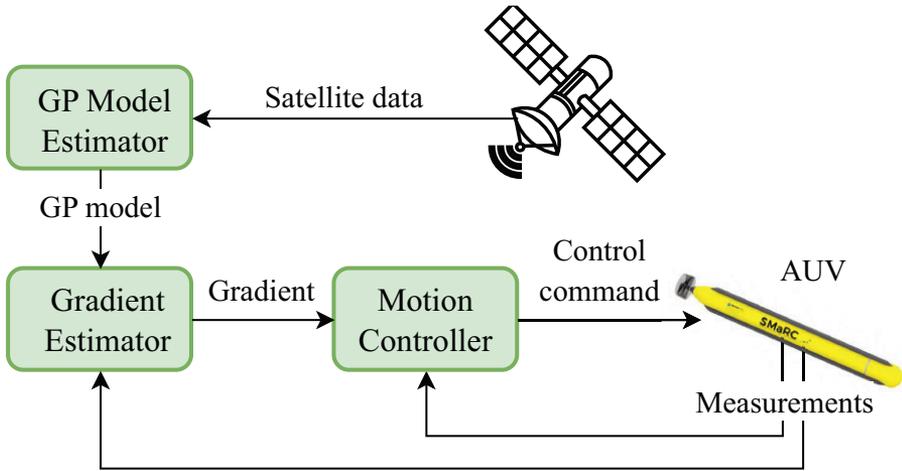


Figure 7.2: Front tracking system architecture with its main components: satellite, GP model estimator, gradient estimator, motion controller, and AUV.

This limited global information consists of satellite imagery from previous days. Then, the AUV has to decide where to explore next, given the information it has collected so far. We approach this problem using an AUV with a chlorophyll a sensor and remote satellite data from CMEMS. Our solution consists of a novel system to be introduced in the following Subsections.

7.2.1 System Architecture

The system architecture for the algal bloom front tracking system is summarised in Fig. 7.2. Its main components are the AUV, motion controller, gradient estimator, GP model estimator, and satellite data collector.

The AUV has a chlorophyll a sensor that measures the chlorophyll a concentration at a set frequency as it moves in the field. The AUV movement is dictated by the control command received from the motion controller. The motion controller uses the AUV's past measurements and a gradient estimate to compute the control command. The gradient estimator uses the past measurements taken by the AUV and a model of the chlorophyll a concentration to estimate the concentration gradient. The GP model estimator uses the previous days of satellite data to train kernel parameters of a concentration GP model. The satellite data consists of chlorophyll a concentration field from a few days preceding the mission.

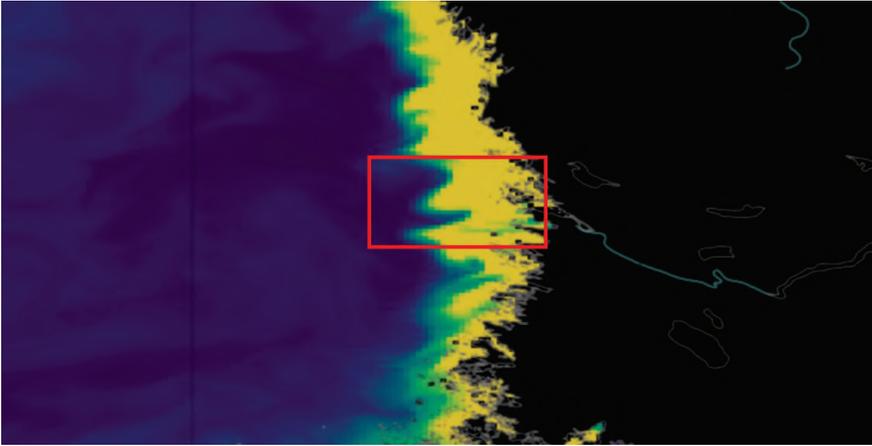


Figure 7.3: CMEMS data of chlorophyll a in the Baltic Sea (blue-yellow) and land (dark grey).

7.2.2 Satellite Data

The satellite data concerns chlorophyll a surface concentration for a given region. We denote this concentration field by $\delta(\mathbf{p})$, where \mathbf{p} denotes the position. Fig. 7.3 shows a sample of chlorophyll a concentration data, where regions of high concentration are yellow, and regions of low concentration are blue. The dark grey area represents the land. The data has a spatial resolution of 2 km by 2 km and is obtained from CMEMS [131]. The location is on the west coast of Finland, near the coastal city *Pori*. A clear chlorophyll a bloom front can be observed here, which can be due to the nutrients that the river *Kokemäenjoki* carries into the Baltic Sea [132]. This paper will focus on the region marked by the red square from April 17, 2021.

7.2.3 GP Model Estimator

The GP model estimator models the chlorophyll a concentration for the given region and time, exploiting prior information from satellite data of the previous days and measurements taken by the AUV in real time. Given the slow time scale, we assume that the chlorophyll a concentration on different days has similar statistical distributions.

To obtain the chlorophyll a concentration GP model, we must first define the kernel that will accurately depict the process. The kernel represents *a priori*

knowledge by specifying how the chlorophyll a concentration is related to the spatial location. Among the many kernels described in the literature, only some fit the characteristics of the biogeochemical data we consider. We use the Matérn kernel [133], which will be shown to be capable of modeling different degrees of smoothness across both vertical and horizontal length scales [134]. The covariance matrix $K \in \mathbb{R}^{N \times N}$ is defined for two points x_i and x_j in the field map. Each matrix element is given by the kernel function $k(x_i, x_j)$, $1 \leq i, j \leq N$. The kernel is defined as

$$K_{i,j} = k(x_i, x_j) = \sigma^2(1 + r_{i,j})e^{-r_{i,j}}, \quad (7.1)$$

where $r_{i,j}^2 = (x_i - x_j)^\top M(x_i - x_j)$, with

$$M = \begin{bmatrix} \frac{3}{l_0^2} & 0 \\ 0 & \frac{3}{l_1^2} \end{bmatrix}. \quad (7.2)$$

The kernel hyper-parameter σ^2 represents the variance of the chlorophyll a concentration process, and (l_0, l_1) the length scales. The hyper-parameters are estimated by maximizing the log marginal likelihood function of the prior distribution using satellite data from previous days. This training set consists of N positions $\mathbf{X} = [\mathbf{p}_1, \dots, \mathbf{p}_N]$ and their respective chlorophyll a concentration values $\mathbf{y} = [\delta_1, \dots, \delta_N]$. The log marginal likelihood [133] to maximize is

$$\log p(\mathbf{y}|\mathbf{X}) = -\frac{1}{2}\mathbf{y}^\top (K + \sigma^2 I)^{-1} \mathbf{y} - \frac{1}{2} \log |K + \sigma^2 I| - \frac{N}{2} \log 2\pi. \quad (7.3)$$

Using the trained kernel, the GP model for the chlorophyll a concentration is obtained from the standard conditioning formulae [133]. Consider the n most recent measurements taken by the AUV. It contains the positions $\mathbf{P} = [\mathbf{p}_1, \dots, \mathbf{p}_n]$ and its measurements $\Delta = [\delta_1, \dots, \delta_n]$. The mean of $\delta(\mathbf{p})$ is denoted $\bar{\delta}(\mathbf{p})$ and the covariance $\text{cov}(\delta(\mathbf{p}))$. The mean and covariance at some point \mathbf{p}_* are given by

$$\bar{\delta}(\mathbf{p}_*) = K_* (K + \sigma^2 I)^{-1} \Delta, \quad (7.4)$$

$$\text{cov}(\delta(\mathbf{p}_*)) = K_{**} - K_* [K + \sigma^2 I]^{-1} K_*^\top, \quad (7.5)$$

where $K \in \mathbb{R}^{n \times n}$ corresponds to the covariance between the data in points \mathbf{P} , $K_* \in \mathbb{R}^{1 \times n}$ the covariance between the data in points \mathbf{p}_* and \mathbf{P} , $K_{**} = \sigma^2$

corresponds to the variance at \mathbf{p}_* , and σ^2 is the variance of the measurement noise.

To evaluate the accuracy of the GP model applied to chlorophyll a concentration fields, the kernel is trained using both low- and high-resolution datasets. These datasets correspond to satellite data, analyzed and processed with different algorithms by CMEMS. Then, the goodness of fit of both sets of parameters is evaluated by comparing the respective predictions of the chlorophyll a concentration to the ground truth. Using past satellite data, we construct the set \mathbf{X} and \mathbf{y} in (7.3) from data from multiple process realizations. Such division of the training dataset prevents overfitting. Then, \mathbf{X} and \mathbf{y} are composed of data from 3 days before the prediction date, selecting non-overlapping randomly scattered sub-datasets of the same size each day. The optimization algorithm for the maximization of (7.3) is L-BFGS-B [135]. The resulting parameters follow in Table 7.1.

Table 7.1: Kernel hyper-parameters obtained through maximum likelihood estimation, using the low-resolution and high-resolution datasets.

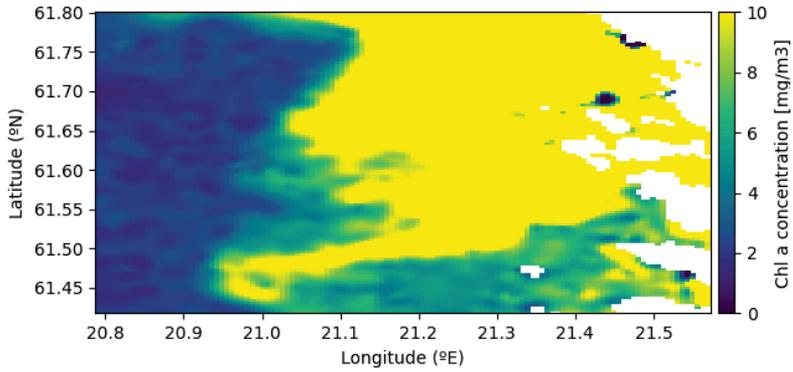
	σ^2	l_0	l_1
Low resolution	44.2959	0.5465	0.2890
High resolution	18.2106	0.0559	0.0245

The results of this comparison are presented in Fig. 7.4a and Fig. 7.4b. The observations and test datasets are approximately of size 1500 and 13500, respectively, where the former is a set of scattered samples from the ground truth data in Fig. 7.4c, having a standard deviation of $\sigma_n^2 = 10^{-3}$.

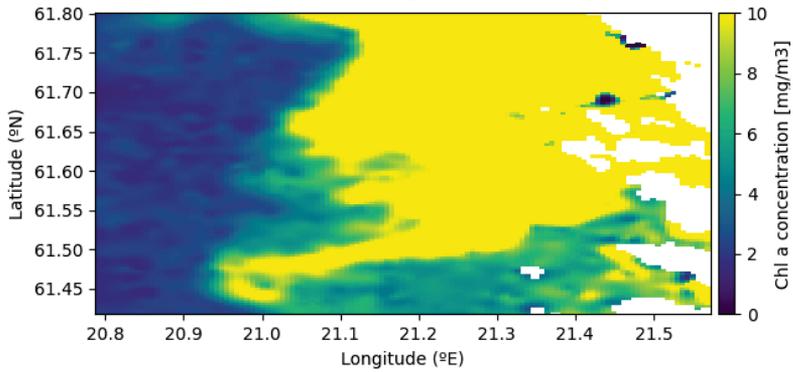
Visually, the results are very similar. The average relative error of the prediction compared to the ground truth data in Fig. 7.4c was approximately 12% and 11% using low- and high-resolution data, respectively. Based on the similarity between the results, we conclude that the proposed GP model accurately represents the statistical properties of the chlorophyll a concentration in the operations scenario, even when the training dataset is different from the ground truth data.

7.2.4 Gradient Estimator

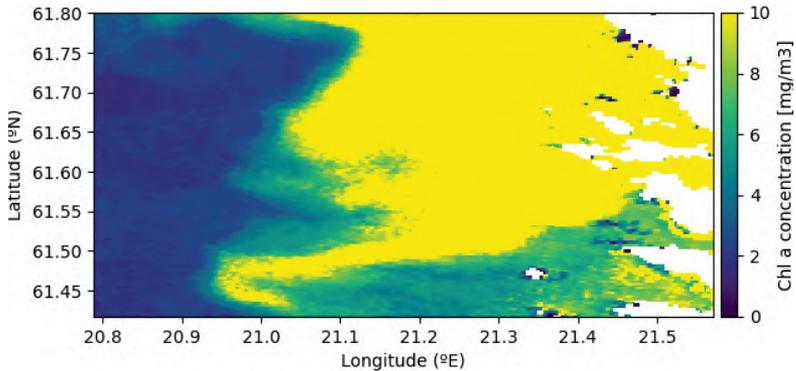
The gradient estimator uses the previously obtained model of the chlorophyll a concentration to estimate the chlorophyll a concentration gradient. From the



(a) Predicted mean using low resolution data.



(b) Predicted mean using high resolution data.



(c) High resolution dataset that simulates the ground truth.

Figure 7.4: Predicted mean from scattered measurements compared to the ground truth (high-resolution data).

equation (7.4), the gradient $\nabla\bar{\delta}(\mathbf{p}_*)$ is obtained by computing the derivative of the predicted chlorophyll *a* concentration with respect to position \mathbf{p}_* ,

$$\nabla\bar{\delta}(\mathbf{p}_*) = \nabla_{\mathbf{p}_*} \left[K_* (K + \sigma^2 I)^{-1} \Delta \right]. \quad (7.6)$$

Since the second and third terms inside the gradient in (7.6) are constant relative to \mathbf{p}_* , we only need to compute $\nabla_{\mathbf{p}_*} K_*$. Each element of the K_* matrix is given by (7.1), in which x_i corresponds to \mathbf{p}_* and x_j corresponds to $\mathbf{p}_j \in \mathbf{P}$. So \mathbf{p}_* ,

$$\nabla_{\mathbf{p}_*} k(\mathbf{p}_*, \mathbf{p}_i) = -\sigma^2 e^{-r} M(\mathbf{p}_* - \mathbf{p}_j).$$

Note that the gradient of the kernel equation is not defined when the test point in \mathbf{P} is equal to the current position \mathbf{p}_* . To account for this, the current position \mathbf{p}_* is not included in \mathbf{P} when computing (7.6). Then the gradient estimate at position \mathbf{p}_* is

$$\nabla\bar{\delta}(\mathbf{p}_*) = \nabla_{\mathbf{p}_*} K_* (K + \sigma^2 I)^{-1} \Delta. \quad (7.7)$$

7.2.5 AUV

The AUV receives the control command \mathbf{u} from the motion controller, specifying the direction and velocity reference. Then, using its internal lower-level controller, the AUV turns this reference \mathbf{u} into thrust commands τ_C . We consider a 6DOF AUV model in which the state is the velocity vector given by $\boldsymbol{\nu} = [u \ v \ w \ p \ q \ r]^T$ containing the translational and rotational velocities. The AUV nonlinear system following Fossen [129] is as follows:

$$(\mathbf{M}_{RB} + \mathbf{M}_A)\dot{\boldsymbol{\nu}} + (\mathbf{C}_{RB}(\boldsymbol{\nu}) + \mathbf{C}_A(\boldsymbol{\nu}))\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{g}(\eta) = \boldsymbol{\tau}_C, \quad (7.8)$$

where \mathbf{M}_{RB} is the rigid body mass and inertia matrix and \mathbf{C}_{RB} is the matrix of Coriolis and centripetal terms on the left-hand side. \mathbf{M}_A and $\mathbf{C}_A(\boldsymbol{\nu})$ represent the added mass effect, $\mathbf{D}(\boldsymbol{\nu})$ represents the damping matrix, and $\mathbf{g}(\eta)$ is the vector of gravitational and buoyancy forces and moments. $\boldsymbol{\tau}_C$ is a vector of external control forces based on the AUV's actuator configuration. The damping matrix $\mathbf{D}(\boldsymbol{\nu})$ has a significant effect on the nonlinear hydrodynamics of the AUV [136].

7.2.6 Motion Controller

The control law we propose is summarised in Fig. 7.5. It relies on the chlorophyll *a* gradient $\nabla\delta$ and the latest chlorophyll *a* concentration measurement δ to

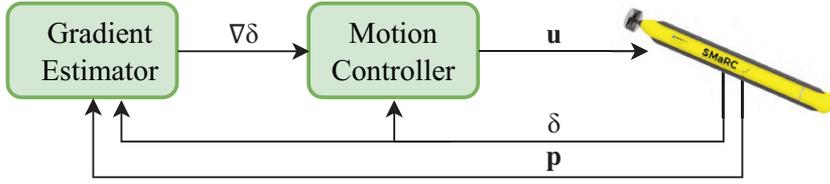


Figure 7.5: Control architecture with the motion controller, the gradient estimator, and the AUV.

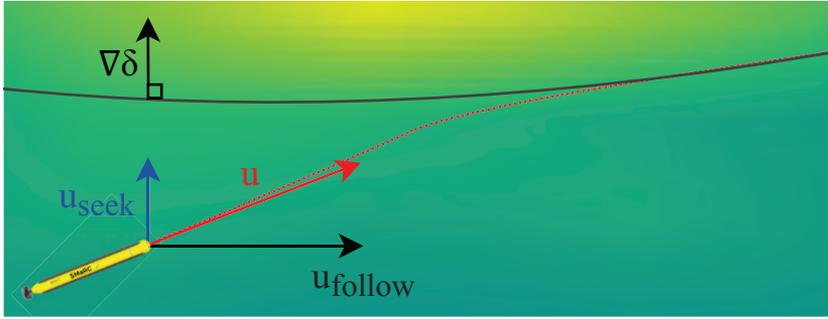


Figure 7.6: Seek and follow components of the control law and gradient.

produce a control command \mathbf{u} . We define a front as a level set of a time-varying scalar field $\delta : \mathbb{R} \times \mathbb{R}^2 \rightarrow \mathbb{R}$ with a slight abuse of notation:

$$F(t) = \{\mathbf{p} \in \mathbb{R}^2 : \delta(t, \mathbf{p}) = \delta_{\text{ref}}\}, \quad (7.9)$$

where δ_{ref} is some reference value, \mathbf{p} the position and t time.

Assuming that the reference value δ_{ref} is known, we use the control law from [137].

$$\begin{aligned} \mathbf{u}(t, \mathbf{p}) &= \mathbf{u}_{\text{seek}}(t, \mathbf{p}) + \mathbf{u}_{\text{follow}}(t, \mathbf{p}) \\ \mathbf{u}_{\text{seek}}(t, \mathbf{p}) &= -\alpha_{\text{seek}}(\delta(t, \mathbf{p}) - \delta_{\text{ref}})\nabla\delta(t, \mathbf{p}) \\ \mathbf{u}_{\text{follow}}(t, \mathbf{p}) &= \alpha_{\text{follow}}R_{\pi/2}\nabla\delta(t, \mathbf{p}), \end{aligned} \quad (7.10)$$

where $\nabla\delta$ is the gradient of δ with respect to \mathbf{p} , $R_{\pi/2}$ is a mapping which rotates vectors by 90 degrees, and α_{seek} and α_{follow} are tunable parameters.

As seen in Fig. 7.6, the control command consists of two components: \mathbf{u}_{seek} , which controls the AUV towards the front by following the gradient field, and $\mathbf{u}_{\text{follow}}$ which controls the AUV to move along the front, perpendicular to the gradient field. These two components ensure convergence to the front [138].

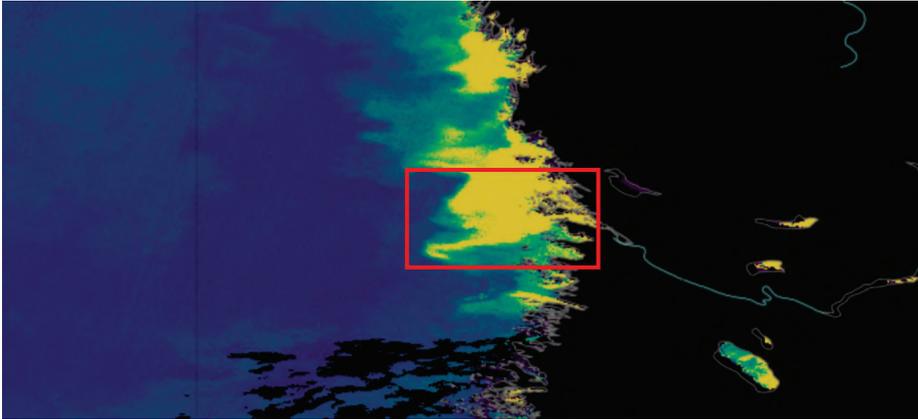


Figure 7.7: CMEMS data of chlorophyll *a* concentration in the Baltic Sea (blue-yellow), clouds and cloud coverage (black), and land (dark grey).

7.3 Simulation Results

In this section, the control and estimation components of the proposed system architecture are tested in a front-tracking procedure in the operational area in Fig. 7.4c. The simulation starts by deploying the vehicle close to the front and providing an initial heading setpoint towards it. When the AUV reaches the front, the gradient estimation is triggered, and the control law receives the estimated value from the GP model as an input. The section is divided into three subsections: simulation setup, numerical results, method analysis, and comparison.

7.3.1 Simulation setup

In this Subsection, we introduce the setup for the simulations. We consider the environment illustrated in Fig. 7.7 in which we will deploy the AUV and track the algal bloom front. Here, the chlorophyll *a* concentration is represented by a map that goes gradually from a high concentration in yellow to a low concentration in blue. The simulated mission occurs inside the red square. The data used to simulate this environment has a spatial resolution of 300 m by 300 m [139] from the exact satellite data location considered earlier. This environment is modeled within the Stonefish simulation environment. The chlorophyll *a* concentration map from the satellite image is integrated as a lookup table to enable

AUV sampling.

The source code implementing the algorithm is available as an open-source contribution on two repositories. The first one is the Gaussian Processes for Adaptive Environmental Sampling (GP4AES) library, which includes the GP model estimator, the gradient estimator, and the motion controller <https://github.com/JoanaFonsec/gp4aes>. The second one is the ROS [140] interface, which uses the GP4AES library and handles the connection with the AUV's software <https://github.com/JoanaFonsec/algalbloom-tracking>.

The simulation starts by deploying the AUV close to the front and providing an initial estimated gradient. When the AUV is near the front, the gradient estimator is triggered. The AUV travels at a constant speed of $v = 1$ m/s. Moreover, based on the available satellite data, we consider $\delta_{\text{ref}} = 7.45$ mg/m³. While tracking the front, the AUV collects measurements at a frequency $f = 1$ Hz while considering a standard deviation of the measurement noise of $\sigma = 10^{-3}$ mg/m³. The measurements are filtered using a weighted moving average filter, with $w = [0.2, 0.3, 0.5]$:

$$\delta_{\text{filtered}}(t) = w_{-2}\delta(t-2) + w_{-1}\delta(t-1) + w_0\delta(t). \quad (7.11)$$

With the same sampling rate, the gradient is estimated as in (7.6), using data from the last $n = 200$ measurements, applying a first-order low pass filter, with $\alpha = 0.97$,

$$\nabla\delta_{\text{filtered}}(t) = \alpha\nabla\delta(t-1) + (1-\alpha)\nabla\delta(t). \quad (7.12)$$

7.3.2 Numerical results

In this subsection, we present and analyze the results from simulated missions using two gradient estimation methods.

We illustrate the AUV mission in Fig. 7.8. The AUV follows the front while collecting chlorophyll *a* concentration measurements, estimating the chlorophyll *a* concentration and its gradient. The complete mission has a duration of approximately 23 hours. The starting position is far from the bloom and represented by the white star, while the final position is on the front and represented by the white square. The AUV closely follows the algal bloom front.

In Fig. 7.9, we zoom in on a region of the longer mission, previously defined by a blue square, to focus on the performance of the gradient estimation and front tracking algorithm. This region corresponds to about 5 hours of mission

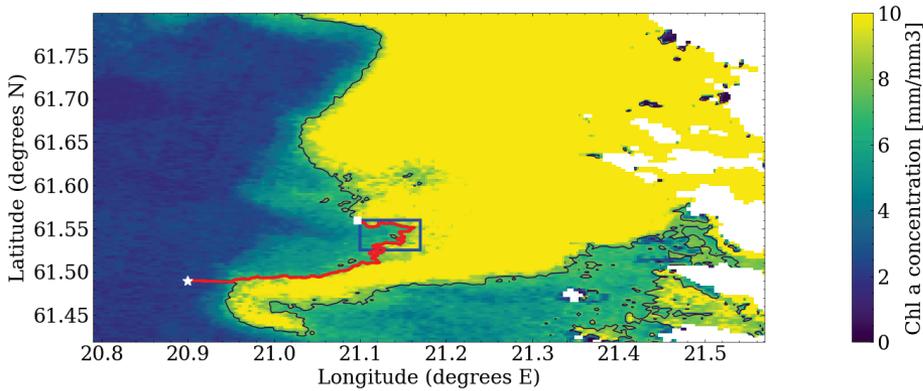
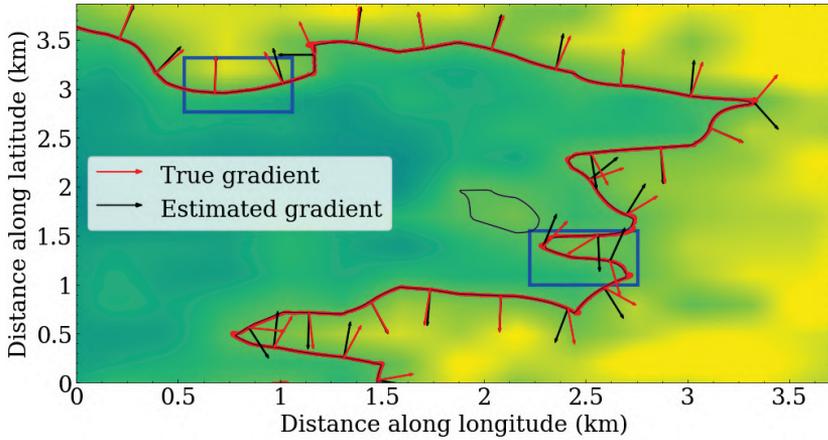


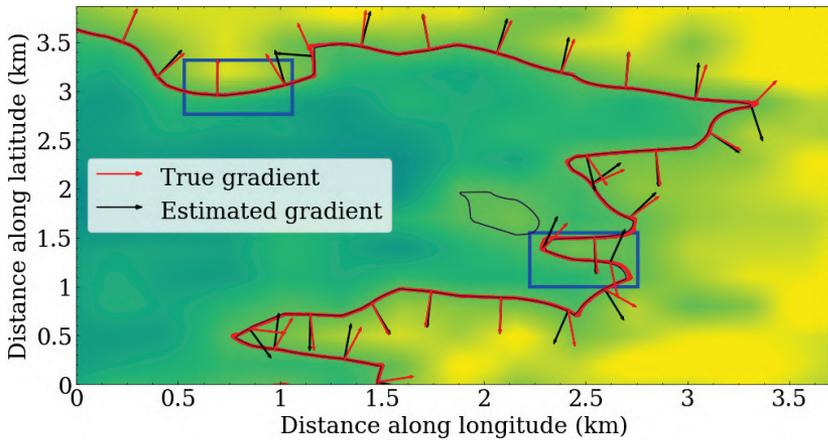
Figure 7.8: Overview of the full mission having the trajectory of the AUV (red) tracking the front (black) in the chlorophyll *a* field (blue-yellow). The white star indicates the initial position, and the white square the final position.

time. Here, we illustrate the gradient performance through arrows representing the true and estimated gradients along the path. The true gradient refers to the gradient that the AUV would be able to compute if it had access to the global information of the field. We compute it by taking the spatial derivative of the chlorophyll *a* concentration. The estimated gradient refers to the output of the gradient estimator (7.7). The angle between the true and estimated gradient arrows indicates the gradient error. Note that the chlorophyll *a* concentration is fast-changing even in small areas. The gradient error is large at paths with high curvature. Let us zoom in on the two areas inside the blue squares.

Fig. 7.10 corresponds to the two zoom-in locations in the previous figure; the front is the thin black line, and a thicker red line represents the AUV path. We also plot the *seek* and *follow* components of the control law using arrows along the AUV path. The control law is constructed as in (7.10) and is a sum of the seek component, which has the same direction as the estimated gradient, and the follow component, which has a perpendicular direction with respect to the estimated gradient. The first zoom-in corresponds to about 15 minutes of mission time. Here, for both estimators in Fig. 7.10a and Fig. 7.10b, the AUV follows the front closely, with small errors and without visible differences between the estimators. This is expected as the front has a small curvature. The control seek component accounts for small adjustments in the trajectory. The second zoom-in corresponds to about 30 minutes of mission time. In Fig. 7.10c



(a) GP estimator.



(b) LS estimator.

Figure 7.9: Trajectory of the AUV (red) tracking the front (black line), with arrows representing the true and estimated gradient.

and Fig. 7.10d, the AUV remains on top of the front most of the time; thus, the control follow component dominates the control law. On the other hand, once the curvature changes faster, the AUV cannot track the front. Two leading causes for this behavior are the AUV's turning radius and the update function with the update rate on the gradient. The gradient's update function in (7.12) introduces a delay and a cut-off frequency. This cut-off frequency is inversely proportional

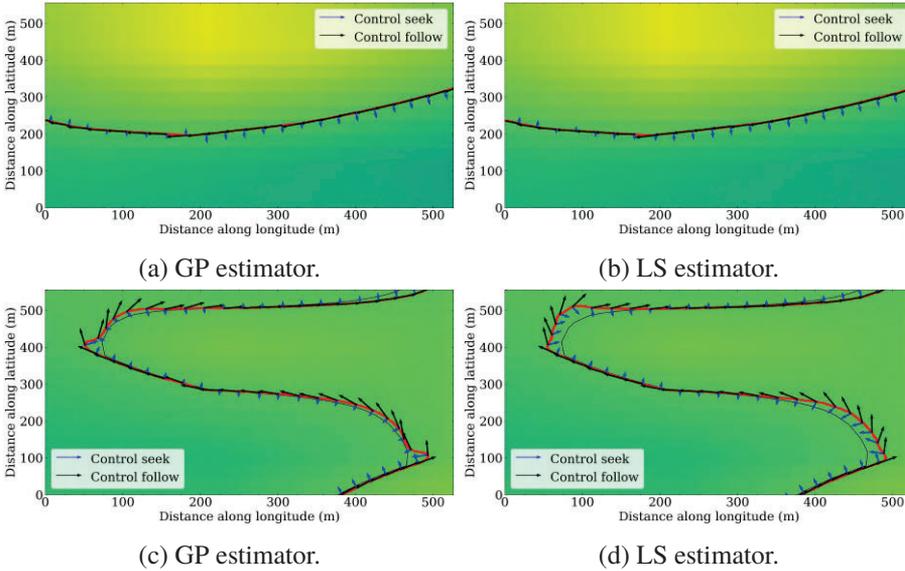
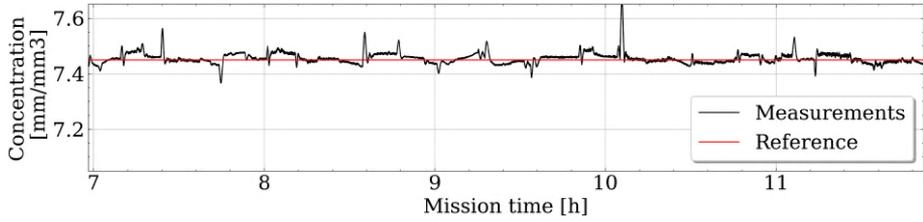


Figure 7.10: AUV path (red) tracking the front (black), with arrows representing seek and follow components of the control law.

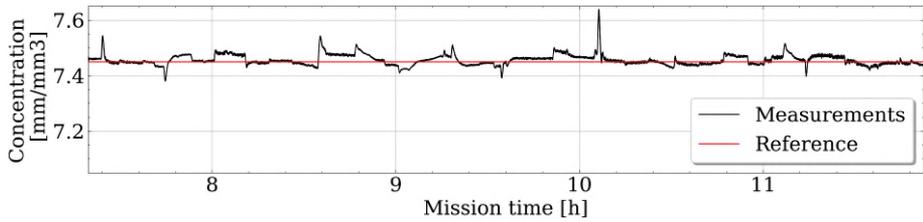
to the update rate. Hence, the algorithm's performance becomes a trade-off between the smoothness introduced by the update function with a lower update rate and the delay it introduces. As for the performance comparison between the two gradient estimation methods, it is apparent from Fig. 7.10 that the GP gradient estimator allows for closer tracking of the front, most notably at regions with fast-changing curvature.

We further analyze the algorithm's behavior through time series plots in Fig. 7.11 and Fig. 7.12 corresponding to the zoom-in area in Fig. 7.9. First, we consider the chlorophyll a concentration measurements taken along the path in Fig. 7.11. The time series indicates that the AUV can track the desired concentration of $\delta_{\text{ref}} = 7.45\text{mg/m}^3$ well with an error lower than $\pm 0.1\text{mg/m}^3$. Second, we consider the gradient field estimation in Fig. 7.12. The gradient estimators perform equally well.

Finally, let us analyze the control law, considering the time series of the two control components \mathbf{u}_{seek} and $\mathbf{u}_{\text{follow}}$, in Fig. 7.13, for the regions defined in Fig. 7.10. In Fig. 7.13a and Fig. 7.13b, we consider the first zoom-in area with an almost linear segment of the front. Here $\mathbf{u}_{\text{follow}} \approx 1$, and \mathbf{u}_{seek} is near zero

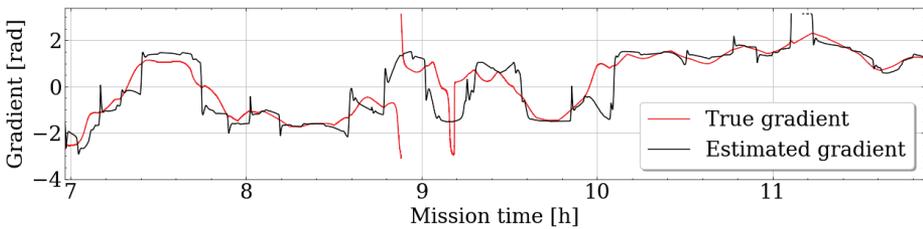


(a) GP estimator.

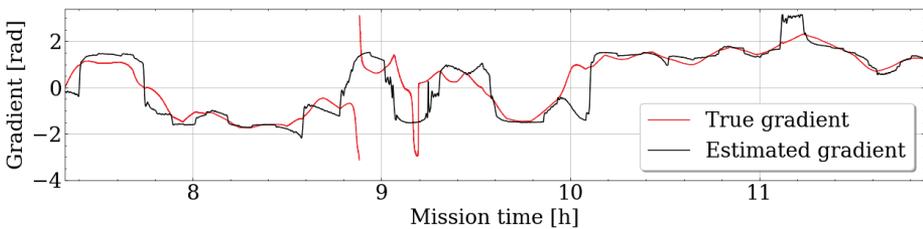


(b) LS estimator.

Figure 7.11: Concentration of chlorophyll a : measurements from the AUV, and reference value.

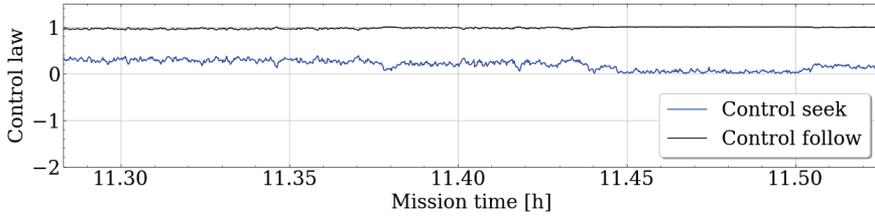


(a) GP estimator.

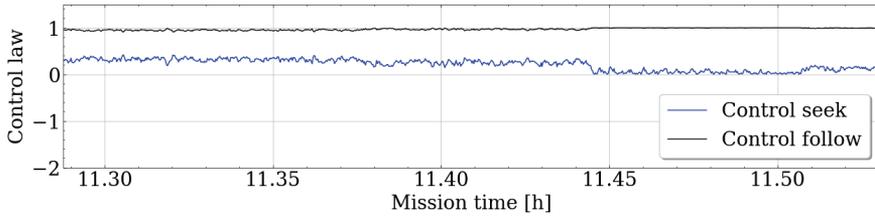


(b) LS estimator.

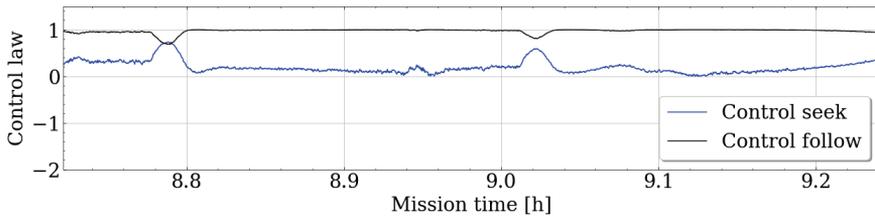
Figure 7.12: Gradient of chlorophyll a : AUV estimated gradient, and true gradient.



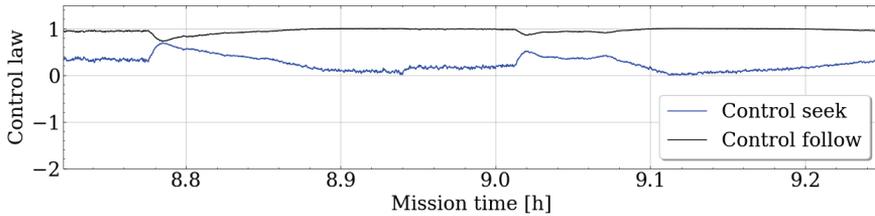
(a) GP estimator.



(b) LS estimator.



(c) GP estimator.



(d) LS estimator.

Figure 7.13: Control law components: seek and follow.

most of the time. In Fig. 7.13c and Fig. 7.13d, we consider the second zoom-in, which contains two tight curves. Here $u_{\text{follow}} \approx 1$, with exceptions at $t = 8.79$ and $t = 9.02$, corresponding to the two peaks in the trajectory curvature in Fig. 7.10c and Fig. 7.10d. Note how the seek component increases when the AUV is far from the front.

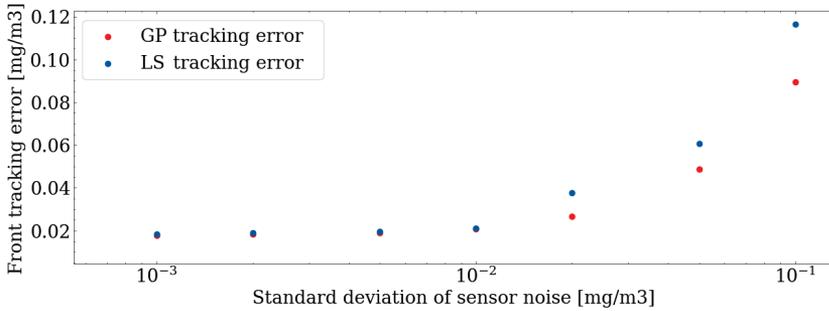


Figure 7.14: Influence of sensor noise in the tracking error, for two different estimation algorithms: GP and LS.

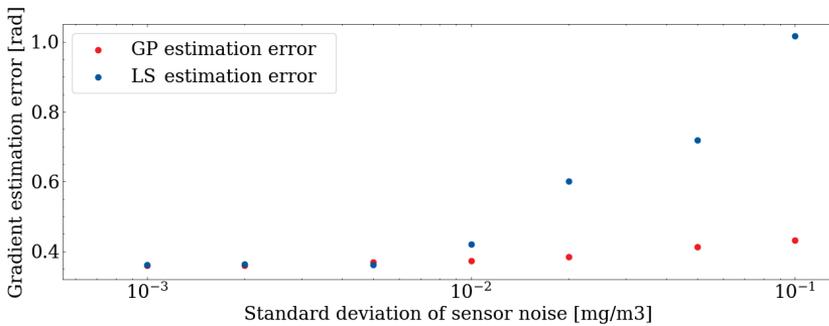


Figure 7.15: Influence of sensor noise in the gradient estimation error, for two different estimation algorithms: GP and LS.

7.4 Sensitivity Analysis

In this Subsection, we analyze the performance of the estimation algorithms for varying levels of chlorophyll a concentration sensor noise. The sensor noise we consider for this analysis varies between 10^{-3}mg/m^3 and 10^{-1}mg/m^3 . This is in line with the chlorophyll a concentration sensors in the market, which typically have a resolution of at least 10^{-1}mg/m^3 [141]. We then run multiple simulations using different standard deviations of the sensor noise and estimation methods. We obtain both the gradient estimation and tracking errors from each simulation.

The gradient estimation error corresponds to the difference between the true gradient $\nabla\delta(t)$ and the estimated gradient $\nabla\bar{\delta}(t)$. Fig. 7.14 illustrates the impact of sensor noise on this error. Here, the tracking error increases with the sensor

noise for both estimators. The difference in performance between the estimators also grows. The tracking error corresponds to the difference between the measured chlorophyll a concentration $\delta(t)$ and the chlorophyll a reference δ_{ref} . In Fig. 7.15, we illustrate the impact of sensor noise on this error, which increases with the sensor noise for both estimators.

This comparison shows that the GP estimator performs better than the LS-based estimator, particularly for a standard deviation of sensor noise of more than 0.01mg/m^3 .

7.5 Summary

In this chapter, we considered the problem of how to use satellite data to improve adaptive sampling missions of an AUV equipped with a chlorophyll a sensor. We developed the adaptive sampling algorithm and software packages to build a solution for the environmental sampling problem. Our solution uses GPs to model chlorophyll a fronts using satellite data and integrates such model into a front tracking algorithm. This integration is done using the estimate of the chlorophyll a gradient field in the control law. We confirmed the goodness of fit of the GP model by using scattered data points from a higher resolution satellite data and were able to reconstruct the chlorophyll a field using the GP model. We implemented the developed algorithm in the AUV's software and ran realistic simulations using the model of our AUV and chlorophyll a sensor model. These simulations resulted in accurate front tracking with low gradient estimation error.

We considered the two most important performance metrics for our objective to be gradient estimation and front tracking errors. Concerning these metrics, the sensor noise analysis indicated that the gradient estimation using GP results in smaller errors compared to when using LSQ, mainly when the sensor noise is bigger or equal to 0.01mg/m^3 , which corresponds to most chlorophyll a concentration sensors on the market. We did not consider other performance metrics, such as computation time, because both methods appeared fast enough to generate an estimate in real-time. We also did not consider the computation time for model fitting prior to the survey as it's not running in real-time and thus not affecting performance. However, for some applications with fast-changing environments, it could be relevant to train the model during the survey. The GP estimator requires a GP model to be fit prior to the mission, which takes a few minutes, as opposed to the LSQ algorithm, which requires no prior fitting.

Chapter 8

Marine Experiments

In this chapter, we implement the method developed in Chapter 7 on an AUV and run experiments in the Stockholm archipelago in the summer of 2022. We design an experimental setup consisting of a cyber-physical system integrating the AUV software, the AUV hardware, the user interface, and a realistic simulator. Numerous packages are included in the AUV software, which can be divided into the behavior tree, the algal bloom front tracking, the onboard controllers, and the dead-reckoning. The algal bloom front tracking library has been developed for the present work and includes the control law and two implemented estimation methods. We provide experimental results from two surveys in the Stockholm archipelago in the Baltic Sea. In these experiments, we demonstrate that the proposed algorithm performs well in the real-time real-world scenario and compare them to a simulation under experiment conditions. We also examine the sources of error, namely surface waves that influence the AUV's movement but also partially occlude the GPS receiver, which introduces Gaussian noise on the GPS-measured position of the AUV.

This chapter is organized as follows. In Section 8.1, we introduce the components included in the experimental setup. This includes the AUV hardware, the Stonefish simulator, the user interface, and the AUV software containing a behavior tree, our algal bloom front tracking algorithm, onboard controllers, and dead reckoning. In Section 8.2, we provide the results from the experimental surveys. Concluding remarks and future directions follow in Section 8.3.

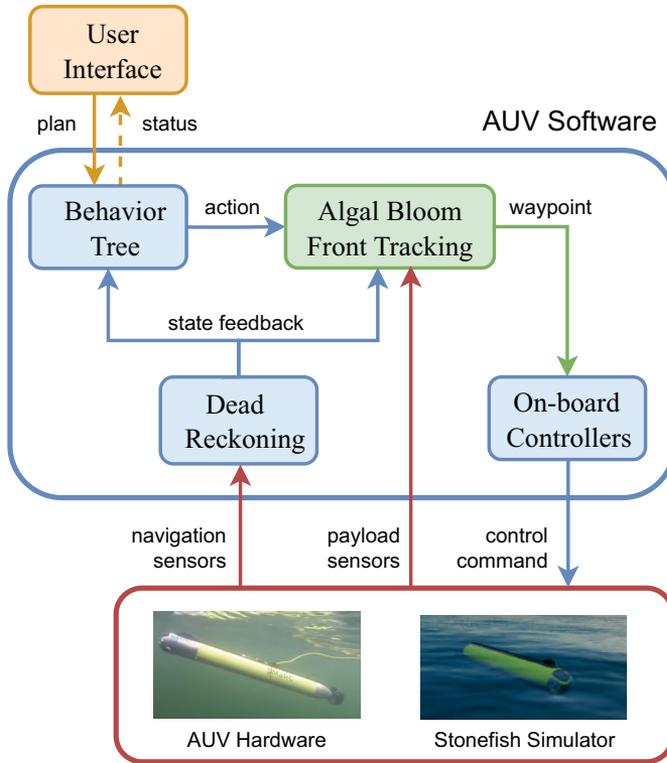


Figure 8.1: The cyber-physical system architecture integrates the user interface, AUV software, AUV hardware, and simulator.

8.1 Experimental Setup

The test platform used in these experiments is the Small and Affordable Maritime (SAM) research AUV, a platform that was developed at the Swedish Maritime Robotics Centre (SMaRC). The algal bloom front tracking system is integrated with the AUV to be deployed in the field and validated experimentally.

Fig. 8.1 illustrates the system architecture of the experimental setup. The user interacts with the AUV software system through an interface that enables the user to send mission plans and monitor the current status. A behavior tree monitors the mission status and delegates actions to an algal bloom front tracker. The algal bloom front tracker reads payload measurements of chlorophyll *a* data and sends waypoints to onboard feedback controllers. The entire software sys-



Figure 8.2: The SAM AUV developed by SMaRC.

tem can also be executed via the Stonefish simulator [142]. Each component will be further described in the following Subsections.

8.1.1 AUV Hardware

We begin by describing the SAM AUV hardware shown in Fig. 8.2. It is a torpedo-shaped, under-actuated AUV [143], [144].

Its five key actuator subsystems are depicted in Fig. 8.3

1. The *battery pack*.
2. The *longitudinal center of gravity (LCG)* system uses the movable battery pack to change the center of gravity position longitudinally and enable static pitch control.
3. The *variable buoyancy system (VBS)* facilitates buoyancy regulation and static depth control by pumping water in and out of a tank.
4. The *transversal center of gravity (TCG)* system contains rotating counterweights that enable static roll control or changes to the AUV's stability margin.
5. The *thrust vectoring* system contains a servo-actuated nozzle for steering in the horizontal and vertical planes. The counter-rotating propellers provide propulsion while compensating for propeller-induced roll.

Sensors have been mounted on the AUV for navigation and environmental sensing. Navigation sensors include an Inertia Measurement Unit (IMU), a compass, a GPS receiver, a Doppler Velocity Logger (DVL) for bottom tracking, and

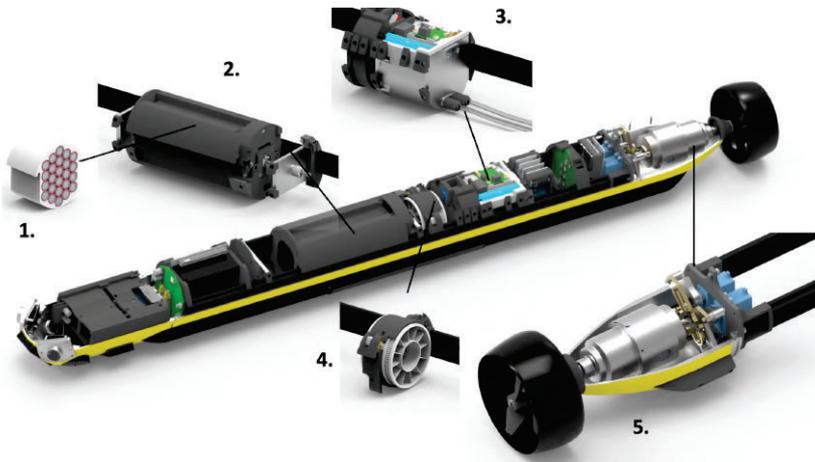


Figure 8.3: AUV subsystems: 1. Battery pack, 2. Longitudinal Center of Gravity Trim System, 3. Variable Buoyancy System, 4. Transversal Center of Gravity System, 5. Thrust Vectoring System with Counter-rotating Propellers.

pressure sensors for depth measurements. Payload sensors include cameras and sidescan sonar for inspection and surveying and a Conductivity-Temperature-Depth (CTD) probe for water-column monitoring. For the algal bloom front tracking application, we use a chlorophyll *a*-turbidity-phycoerythrin fluorometer for phytoplankton sensing.

8.1.2 Stonefish Simulator

Simulations of mission scenarios are performed using the Stonefish simulator, see Fig. 8.4. The AUV's dynamics and sensors are modeled within the simulator. Objects, environmental features, and bathymetry can be imported into the simulator to create mockups of planned environments. Within Stonefish, perception and planning software can be validated before deployment on the hardware. The satellite data for algal blooms is modeled in Stonefish as a lookup table of chlorophyll *a* values over a grid encompassing the entire mission environment. A simulated chlorophyll *a* sampler interpolates the relevant chlorophyll *a* measurement from this grid based on the AUV's position. The software interfaces to the Stonefish simulator and the real AUV are identical, thus enabling the virtual validation of a full mission sequence for algal bloom tracking.

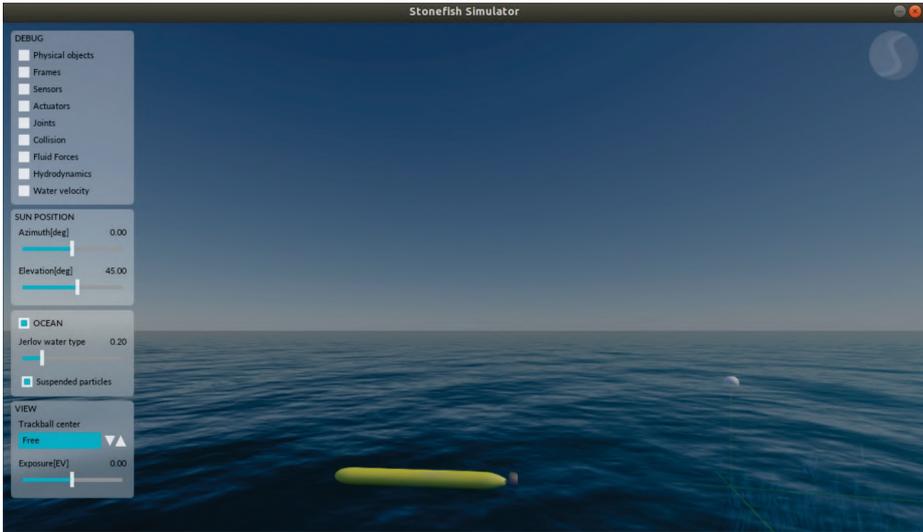


Figure 8.4: Print screen of the Stonefish simulator.

8.1.3 AUV Software

The four software components presented in the center of Fig. 8.1 are further detailed in this Section. The autonomy software runs on the Robot Operating System (ROS) [140] environment. The components include a *behavior tree* for decision-making and mission execution, a path planner for *algal bloom front tracking*, *onboard controllers* for path following, and a *dead-reckoning* for navigation.

Behavior Tree

The AUV uses a behavior tree (BT) to ensure safe and transparent mission execution. A BT is a reactive decision-making structure that is comprised of sequences, fallbacks, actions, and conditions. The main objective of the BT is to receive a mission plan and delegate actions (e.g., waypoints) to lower-level systems (e.g., motion planners and controllers) while monitoring the AUV safety. In an unsafe situation, the BT executes emergency actions to bring the system back to a safe state. A BT ensures safety and compliance requirements during mission execution by disallowing unsafe behaviors autonomously. Further information on designing BTs for underwater robots can be found in [145], [146].

The GP path planner presented in Section 7.2.6 is integrated into the BT with additional conditions and actions considering satellite data ingestion, chlorophyll *a* sampling, and front tracking. Each algal bloom tracking experiment follows an operational sequence. Considering a relevant area with algal blooms, the following workflow is used for sampling and tracking the bloom:

1. An area of interest is specified, and satellite imagery is downloaded.
2. An initial AUV mission plan is set via a user interface based on the initial algal bloom feature.
3. The AUV is launched, and a GPS fix is acquired.
4. The AUV starts the mission and follows user-defined waypoints to reach the vicinity of the algal bloom front.
5. The AUV detects the front through its chlorophyll *a* sensor, and then the front tracking algorithm is engaged.
6. The path planner generates new waypoints for the AUV to track the front.

The BT for the algal bloom tracking is summarised in Fig. 8.5. A sanity check on chlorophyll *a* measurements is performed in the first sub-tree. Second, safety conditions are checked. If either fails, the mission is aborted, and emergency actions are performed. Third, the user-defined waypoint mission is followed if the system is safe and measurements are available. Fourth, when the AUV reaches the front, the algal bloom front following action is performed. This is further detailed below.

Algal Bloom Front Tracking

This action ingests payload data on chlorophyll *a* concentration and sends live waypoints to the onboard controllers so that the AUV follows the front. When the AUV crosses the algae front, the front tracking behavior is enabled, with a higher priority than following the original waypoints (see Fig. 8.5). A path planner for front tracking sends new waypoints to the AUV based on real-time measurements. The AUV samples the front and follows the edge of the bloom. Once the AUV has exited the front, the vehicle will fall back to the operator's plan.

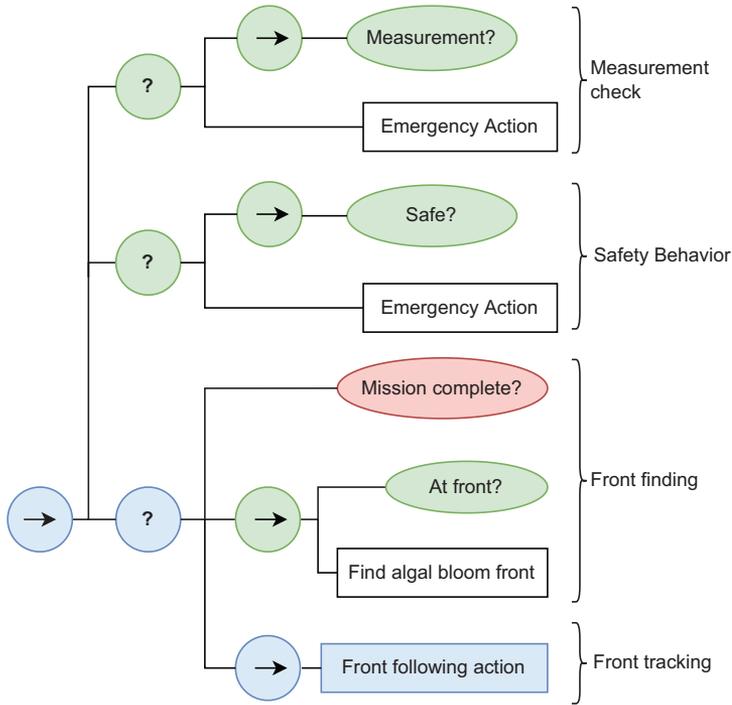


Figure 8.5: High-level view of the BT used for algal bloom front tracking. Inner nodes are sequences (arrows) and fallbacks (question marks). Leaf nodes are actions (rectangle) and conditions (ellipse). All nodes can return Success (green), Failure (red), and Running (blue).

Onboard Controllers

The front tracking algorithm provides input to the onboard controllers. These controllers enable the AUV to follow pre-defined waypoints and track the algal front. Given a set of waypoints, a line-of-sight guidance law minimizes cross-track error and ensures the vehicle approaches each waypoint at a set heading and depth. Further information on the waypoint following guidance law can be found in [147].

The control force vector $\tau_C(c)$ in equation (7.8) is a function of the actuator input

$$c = [rpm_1 \quad rpm_2 \quad d_e \quad d_r \quad LCG \quad VBS], \quad (8.1)$$

where rpm_1 and rpm_2 represent the propeller speeds, d_e and d_r are vertical

and horizontal thrust vector angles, and *LCG* and *VBS* specify the position and buoyancy level, respectively.

The low-level feedback control architecture is presented in Fig. 8.6. The flight control regulates the heading and depth where cascaded Proportional-Integral-Derivative (PID) controllers are used to command the thrust vector angles. In the outer loop, the controllers provide a yaw rate and pitch setpoint, translated to actuator commands to the thrust vectoring system in the inner loop. These account for couplings between states for flight control. To stabilize the AUV in pitch and depth, additional PID controllers control the trim (*LCG*) and buoyancy (*VBS*) subsystems. Finally, coupled roll and velocity control is realized using parallel PIDs to command the counter-rotating propellers. These provide an average propeller rpm to achieve the desired velocity while also providing a differential rpm between the two propellers that causes the AUV to hold a roll angle. It is possible to directly command constant rpm values instead of a desired velocity. The combination of flight and trim controllers enables the AUV to track the algal front at a specified velocity or propeller rpm but depends on reliable state feedback.

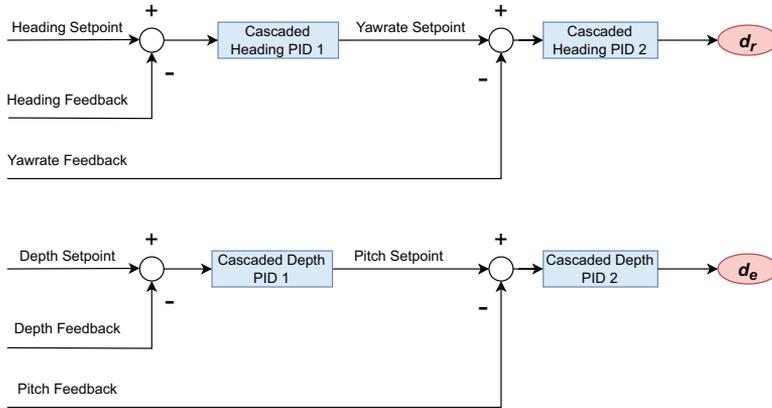
Dead Reckoning

Underwater navigation is challenging because radio waves attenuate rapidly in water. This means that GPS-based positioning and navigation are unavailable underwater, which entails that we need to use inertial and acoustic sensors to estimate AUV position, orientation, and velocities. Dead reckoning is thus used to support feedback control through an extended Kalman filter. It fuses acoustic and inertial measurements collected by the onboard sensors to estimate the vehicle's position, orientation, and velocity. In particular, the IMU and compass are used to obtain orientations, angular velocities, and accelerations, the DVL is used to obtain linear velocities, and the pressure sensor is used to measure the depth.

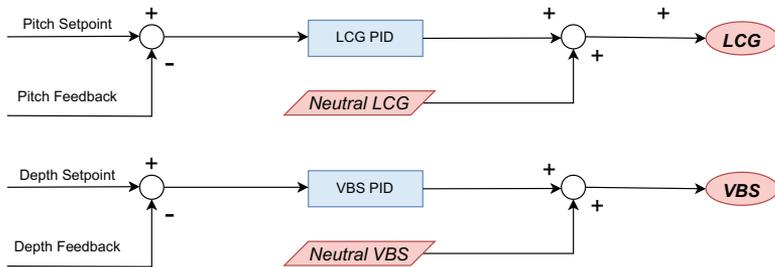
8.1.4 User Interface

The user interacts with the software system via the user interface in the upper left corner of Fig. 8.1. As represented in Fig. 8.7, it consists of a web-based graphical interface based on *Node-RED*. It enables the operator to plan the mission on a world map and monitor the vehicle's status during the mission. AUV variables can be tracked, new missions can be run, and measurements can be observed.

Heading and Depth Control



Pitch and Buoyancy Control



Velocity and Roll Control

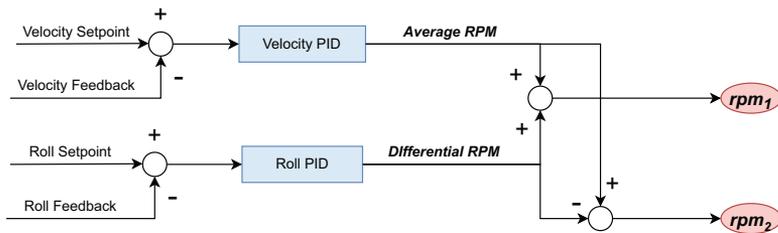


Figure 8.6: The feedback control architecture on SAM with cascaded heading and depth PIDs for flight control (top), trim stabilization with pitch and buoyancy control (middle), and coupled velocity and roll control with the counter-rotating propellers (bottom).

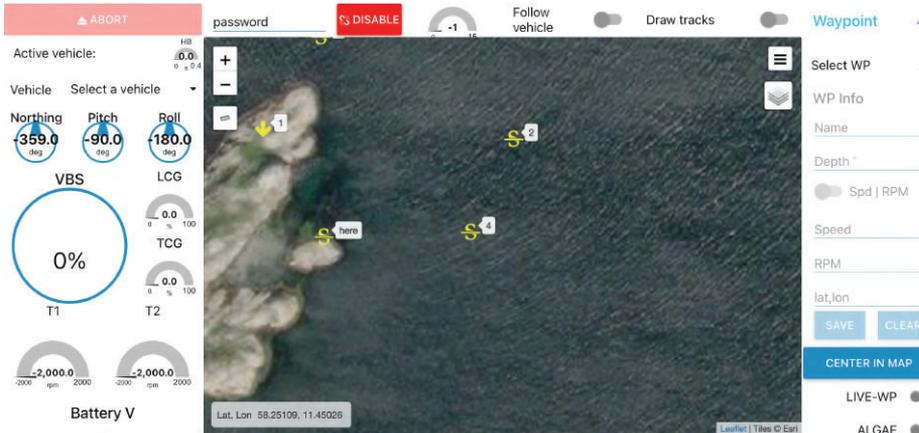


Figure 8.7: Print screen of the user interface based on *Node-RED*. It contains a map to visualize the mission, AUV variables, and measurements.

A second graphical user interface allows test engineers to run specific hardware drivers and controllers to validate low-level controllers.

8.2 Experimental Results in the Baltic Sea

This Section contains experimental results obtained using the front tracking system. The gradient estimator and motion controller were validated in field experiments. These experiments took place in the Stockholm archipelago in the Baltic Sea, near the island of Djurö. The AUV during an experiment is shown in Fig. 8.8. There were no algal blooms in the area of the experiments, so we scaled the data in Fig. 7.7 to simulate the chlorophyll *a* concentration. For these experiments, we applied a low-level controller that guarantees area coverage around the front. This controller introduces sinusoidal movement around the front.

Fig. 8.9 shows two experiments and two simulations under experimental conditions. The experiment in Fig. 8.9a was conducted on July 18, 2022. It corresponds to about 12 minutes of mission time at an average speed of 0.11m/s. The experiment in Fig. 8.9b was conducted on August 11, 2022. It corresponds to about 10 minutes of mission time at an average speed of 0.11m/s. Fig. 8.9c shows a simulation in which we set the conditions to match the conditions of the experiments, including the low-level sinusoidal controller. Fig. 8.9d illustrates simulations with the same experimental conditions but using a linear low-level



Figure 8.8: The AUV in the water on the mission day, while tethered to the boat.

controller, resulting in tighter front tracking performance.

Let us now analyze the results. In Fig. 8.10, we evaluate the trajectory of the AUV and the gradient estimator. The trajectories of the AUV in Fig. 8.10a and Fig. 8.10b appear jittery. This is due to the noise in the GPS signal and water currents. Water currents drag the AUV to move in a different direction than the algorithm calculated. These currents are also one of the causes for the jittery GPS signal as the GPS receiver is a few centimeters above the surface, and the existence of waves directly influences the signal quality. In Fig. 8.10c and Fig. 8.10d, we emulate this phenomenon by introducing Gaussian noise on the GPS receiver. Notice that the AUV oscillates around the front due to the implemented sinusoidal low-level controller. Thus, the front tracking error appears large. The scenario with a linear low-level controller in Fig. 8.10d illustrates the performance of the developed motion controller more clearly. The implications of these sinusoidal and linear low-level controllers are shown in the chlorophyll *a* concentration tracking error in Fig. 8.12. In Fig. 8.13, we can see how the estimated gradient resembles a low-pass filtered version of the true

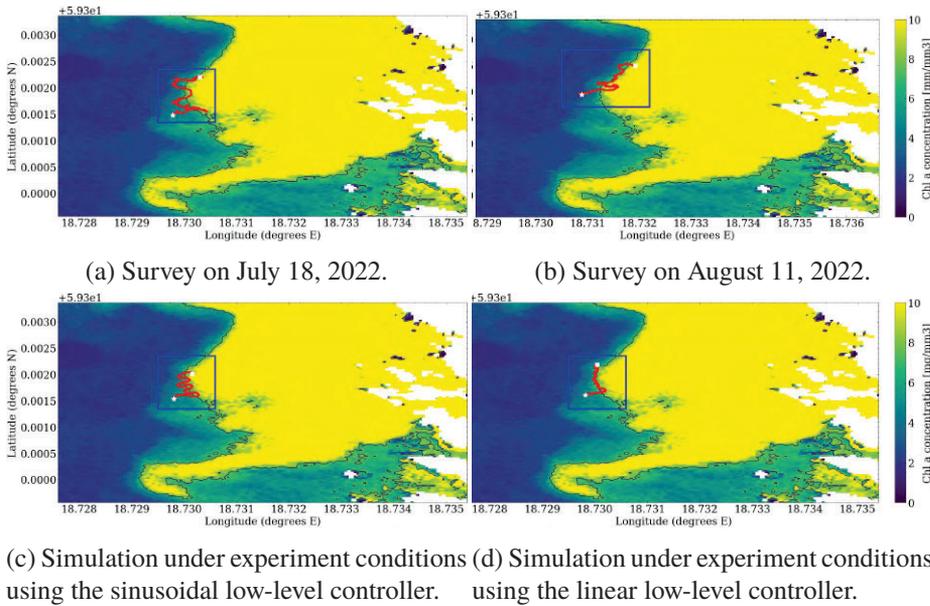


Figure 8.9: Two experimental surveys and two simulated scenarios under experimental conditions with AUV trajectory (red) tracking the front (black line) in the chlorophyll a map (blue-yellow).

gradient. We illustrate the control law and its components in the second zoom-in in Fig. 8.11. Here, the control law's seek and follow components match the jittery trajectories. The follow component tends to be parallel to the front, pointing forward, and the seek component tends towards the front. Analyzing Fig. 8.14, we can see how the over- and undershoots correspond to times where seek is the dominating component.

8.3 Summary

In this chapter, we considered the problem of how to use satellite data to improve adaptive sampling missions of an AUV equipped with a chlorophyll a sensor and how to design a survey in the Baltic Sea to test our approach. After the development, implementation, and sensitivity analysis, we designed a survey in the Baltic Sea, near Stockholm, next to the island of Djurö. Similarly to the simulations, the experiments confirmed that the algorithm and software pack-

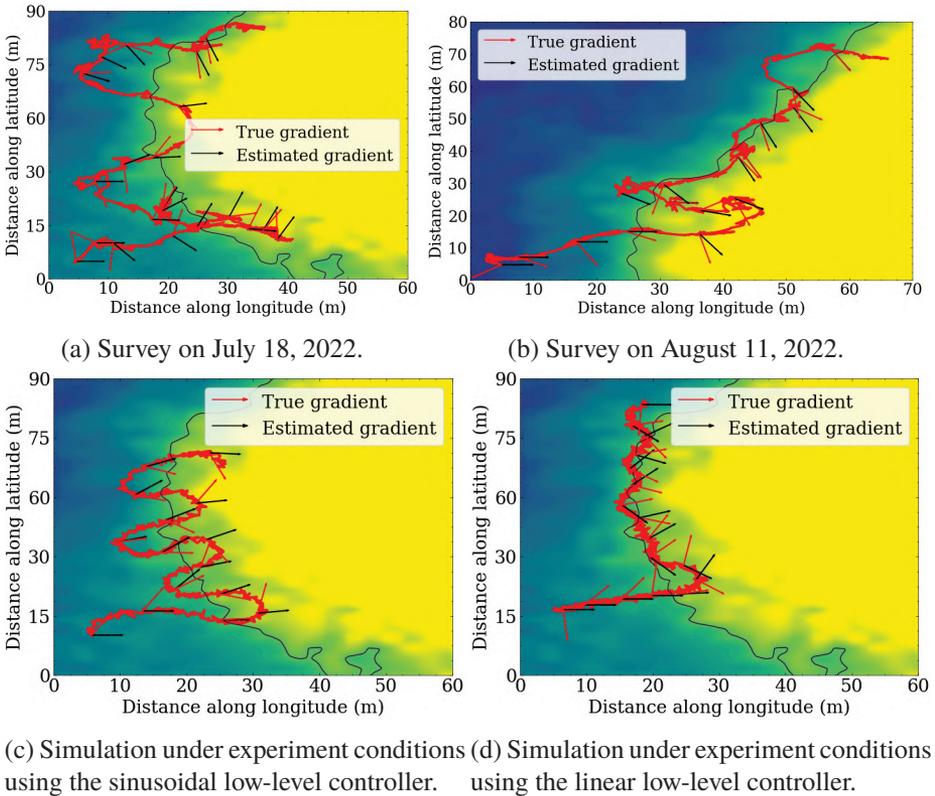
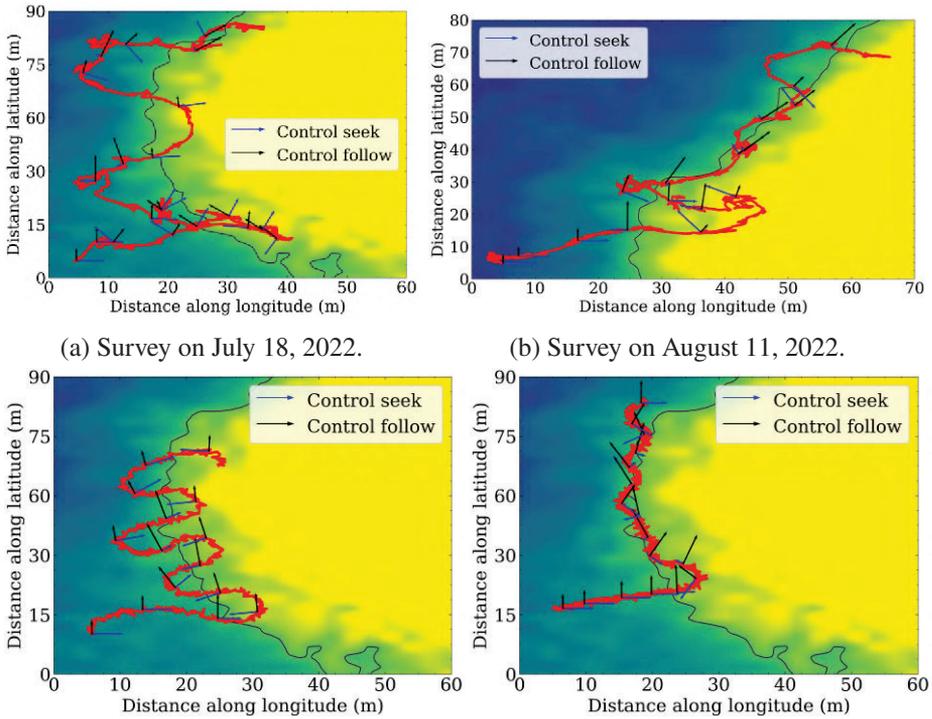


Figure 8.10: Trajectory of the AUV (red) tracking the front (black line), with arrows representing the true and estimated gradient.

age work as desired in a controlled environment. On the implementation side, future work would include integrating the algae sensor into an AUV, tuning it in a controlled setting, and running experiments with the complete system. One relevant contribution would be to work towards a higher degree of autonomy. There are several steps to reach a level of robustness sufficient for real-world deployment of higher autonomy. Some are reliable collision avoidance for islands, boats, and people, robust autonomous dock-in for charging, and a cloud-based data storage solution.



(c) Simulation under experiment conditions using the sinusoidal low-level controller. (d) Simulation under experiment conditions using the linear low-level controller.

Figure 8.11: Trajectory of the AUV (red) tracking the front (black line), with arrows representing the two components of the control law.

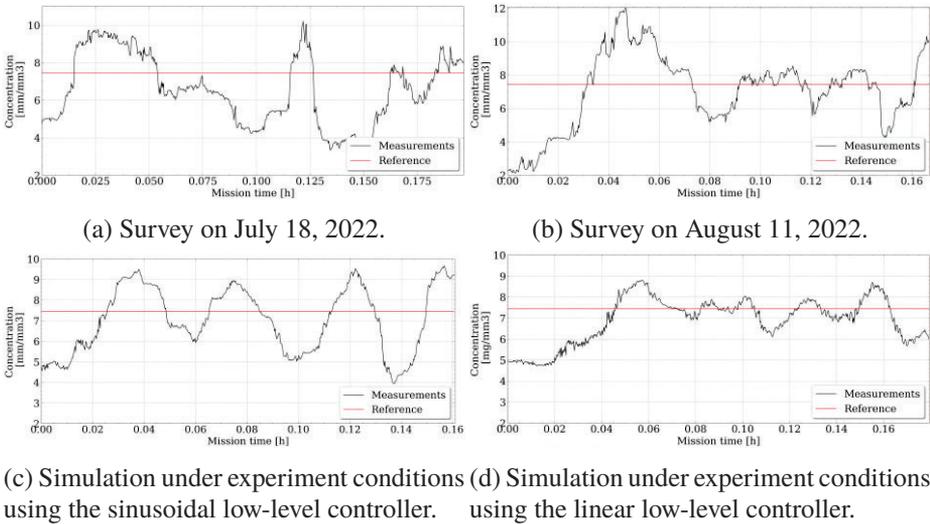


Figure 8.12: Concentration of chlorophyll a: measurements from the AUV, and reference value.

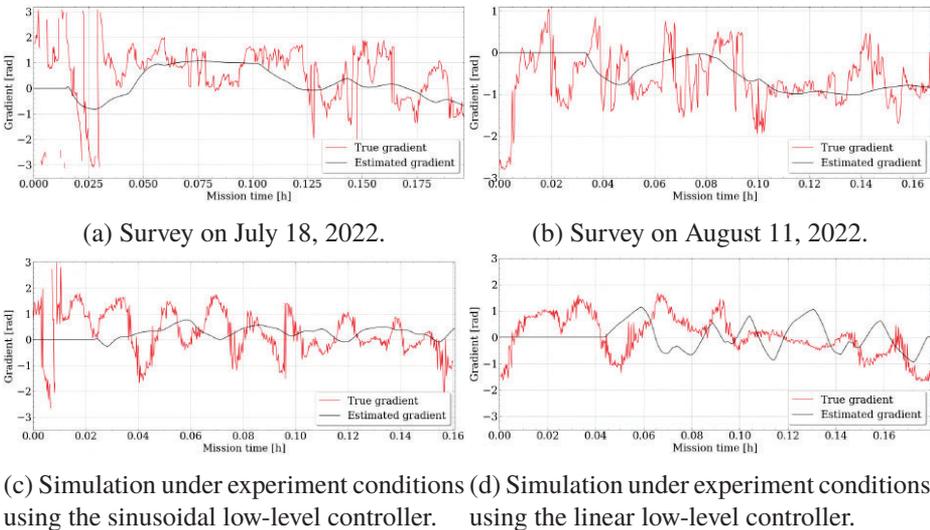


Figure 8.13: Gradient of chlorophyll a: AUV estimated gradient, and true gradient.

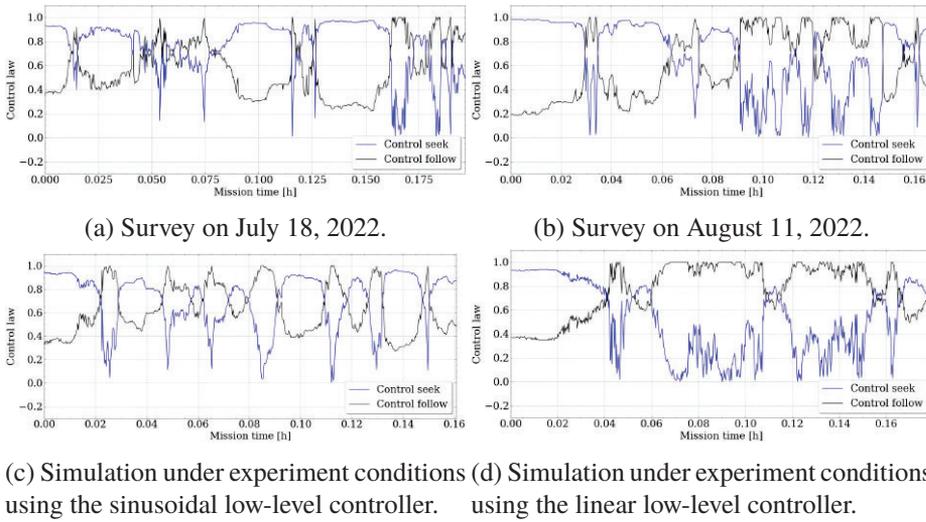


Figure 8.14: Control law components: seek and follow.

Chapter 9

Conclusions

In this chapter, we conclude the thesis. In Section 9.1, we summarise and discuss the presented results, Section 9.2 outlines potential extensions and future work, and, finally, Section A provides a thorough analysis of the articles cited in the present thesis.

9.1 Summary

The central question of this thesis was how to optimize ocean feature estimation and tracking using AUVs through adaptive sampling and formation control techniques. The ocean features we considered were algal blooms in two different scenarios: a target and a front. The approaches to tracking these ocean features were multi- and single-agent and considered the available satellite imagery. We defined estimation and control protocols for each scenario.

In Chapter 3 and Chapter 4, we considered the problem of tracking and circumnavigating a mobile target using a multi-agent system. The considered mobile target was an irregular dynamic shape approximated by a circle with a moving center and varying radius. A satellite image indicated the algal bloom's existence and initial location for deploying the multi-agent system. Chapter 3 considered an approach with a multi-agent system containing one far-sighted leader that can measure the distance to both the boundary and the center of the target. In this scenario, the leader was an AUV equipped with a UAV. This AUV used adaptive estimation to calculate the location and size of the mobile target. Then, the multi-agent system circumnavigated the boundary of the target while forming a regular polygon. We designed two algorithms: One for the adaptive

estimation of the target using the UAV's measurements and another for the control protocol to be applied by all agents in their navigation. The convergence of both algorithms to the desired state was proved up to a limit bound. Two simulated examples were provided to verify the performance of the designed algorithms. Chapter 4 considered a decentralized approach in which all agents measured their distance to the boundary of the target. A circle with time-varying location and size approximated the algal bloom area. The proposed algorithm comprises a decentralized least squares estimation of the target and a controller for circumnavigation. We proved the convergence of the agents to the circle and equally spaced positions around it. Simulation results with data from the SIN-MOD ocean model were used to illustrate the theoretical results. We extended this result to the case of non-circular targets. Each agent estimated the local curvature of the target using the measurements of its neighboring agents. At the same time, the control law accounted for the equal spacing between agents by driving the agent to be equidistant between its neighbors. We illustrated these results with two simulated scenarios.

In Chapters 5, 6, and 7, we considered the problem of tracking a mobile front using a single-agent system. Similarly to the multi-agent target tracking scenario, satellite imagery aided the front tracking protocol. In contrast to the multi-agent target tracking scenario, there were no assumptions on the shape of the algal bloom front. We assumed that any front can be locally approximated by a curvature and thus represented by a normal vector, the gradient, given that the front is an isoline of a concentration field. Chapter 5 discussed a local linear approximation of the gradient. In this chapter, we considered a river front. The approach was evaluated and tested in simulation using a high-resolution hydrodynamic front model. The test plan included 307 test cases covering the most representative environmental conditions. Space limitations precluded a thorough discussion of the results, but our parameterizable approach and the study of the conditions for parameter selection will be at the heart of a decision support system for optimized front surveys. Chapter 6 considered a local quadratic approximation of the gradient using least squares regression. The agent had a GPS receiver that reported its position and a chlorophyll *a* concentration sensor, which measures the local algal concentration. We proposed an experimental setup composed of satellite data, AUV hardware, software, CMEMS forecasted data, and an algorithm in which the agent estimates the local algal gradient using recent past measurements and least squares fitting. We provided a simulation and analyzed convergence given sensor noise. The agent converged and moved

along the detected algal bloom front. The algorithm was also implemented on the LSTS Toolchain using model forecasts of the chlorophyll *a* concentration in the Baltic Sea in February 2020 from CMEMS. Chapter 7 considers a Gaussian process regression of the gradient. Beyond GPS and sensors, we considered using satellite imagery to improve the survey. Our solution used satellite imagery and GPs to model chlorophyll *a* fronts and integrated such model into the front tracking algorithm. This integration was done using the estimate of the chlorophyll *a* gradient field in the control law. We confirmed the goodness of fit of the GP model by using scattered data points from a higher resolution satellite data and were able to reconstruct the chlorophyll *a* field using the GP model. We implemented the developed algorithm in the AUV's software and ran realistic simulations using the model of our AUV and chlorophyll *a* sensor model. These simulations resulted in accurate front tracking with low gradient estimation error.

Finally, in Chapter 8, we implemented the algorithms developed in Chapter 7 on an AUV and ran a survey in the Baltic Sea, near Stockholm, next to the island of Djurö. The algorithms were implemented into two software packages that interface with the remaining AUV software. Similarly to the simulations, the experiments confirmed that the algorithm and software packages work as desired in a controlled environment.

In conclusion, this thesis addressed optimizing ocean feature estimation and tracking using AUVs through adaptive sampling and formation control strategies. Focusing on algal blooms in different scenarios, the research explored multi-agent and single-agent approaches, incorporating available satellite imagery. In the multi-agent target tracking scenario, the adaptive estimation and control protocols were effective in accurately estimating and circumnavigating mobile targets for both leader-follower and distributed scenarios. Furthermore, the front-tracking study using a single agent resulted in three suitable methods: linear approximation, quadratic optimization, and Gaussian Processes. Comprehensive simulations using numerical models confirmed the viability and accuracy of the proposed algorithms. These were further validated by successfully implementing the algorithms on an AUV and subsequent field survey. The achievements presented in this thesis pave the way for future advancements in AUV-based ocean research and provide valuable insights into the advancement of environmental monitoring. The development of more autonomous and robust algorithms promises to revolutionize ocean exploration and contribute significantly to society's efforts in preserving and managing marine ecosystems.

9.2 Future work

The research presented in this thesis lays the foundation for advancing the field of ocean feature estimation and tracking through adaptive sampling and formation control of AUVs. The successful development and implementation of the proposed methods have revealed several exciting directions for further exploration and enhancement. In this section, we outline the potential future work that can extend this thesis's contributions and address some of the limitations and challenges encountered during the research. The proposed extensions are enhanced adaptive sampling strategies, heterogeneous multi-agent collaboration, integration of advanced sensing technologies, real-time data handling and data fusion, long-term AUV autonomy, large-scale field experiments, and environmental impact assessment.

In this thesis, we considered several adaptive sampling strategies for single and multi-agent systems under various assumptions. This work effectively improved the efficiency of data collection by guiding the AUVs to informative regions of the map while maintaining a desired formation or behavior. Further investigation could result in more advanced adaptive sampling strategies. For example, by introducing real-time environmental data such as wind, waves, and ocean currents as modeled in [129]. Embedding these uncertainties into the control protocol will make the control laws more robust and thus enhance sampling efficiency in diverse oceanic environments. These improved control methods would be able to reject disturbances [148] and achieve robust consensus [149]. Considering more challenging oceanic environments with high ocean current uncertainty and obstacles, it is relevant to investigate uncertainty-aware motion planners that can estimate online the unknown fluid field uncertainty and aid in choosing safe and effective actions. For example, by considering Bayes Adaptive Markov Decision Process for robust decision making [150].

In the multi-agent scenario, we evaluated decentralized and leader-follower approaches, always focusing on optimizing feature tracking and estimation. Extending this work to multi-AUV collaboration strategies with high-level decision-making would further improve the developed protocols to maximize the utilization of available resources and reduce overall mission time. For example, in large or dynamic environments, the exploration-exploitation balance is crucial to unlock high levels of flexibility, adaptivity, and swarm intelligence. There are several methods to control the level of exploration and exploitation carried out by multi-agent systems [151]. Another relevant extension for large or dynamic

environments is high-level task allocation. There are different task allocation strategies depending on the application [152]. A fitting option for this decision-making layer would be to introduce auction-based multi-agent task allocation with multi-constraint, in which the constraints could be the time window for task execution, the fuel cost, the availability, and others as developed in [153]. Finally, a very relevant consideration for extension is heterogeneous multi-agent systems. These systems may be teams consisting of AUVs, USVs, UAVs, gliders, and buoys that may carry different types of sensors with varying sensor ranges, vehicle sizes, and maneuverability [154]. This variability allows for better coverage and smarter decision-making but also introduces more constraints in the optimal coverage problem, making it a challenging yet compelling extension.

This thesis considered tracking algal bloom fronts and salinity fronts. We assumed the available sensors were state-of-the-art chlorophyll *a* and conductivity sensors. Indeed, these are the most commonly used sensors for both applications. However, novel methods for analyzing the underwater environment require novel control strategies. For example, collecting water samples for later analysis may be necessary to understand the species of algae bloom. In that case, the AUVs must develop adaptive triggering methods to capture water samples at certain relevant locations [155]. Given the recent development of the Environmental Sample Processor (ESP), it is now possible to analyze the water sample in situ and, for example, determine the algae bloom species [156]. This advancement will result in the need for more advanced adaptive control strategies.

The primary emphasis of the present thesis has been the development of intelligent path planning and adaptive sampling strategies for AUVs to track and estimate ocean features. Data assimilation emerges as a relevant extension after verifying the efficacy of the proposed estimation and control protocols for data collection. Though relevant, it is a broad research field that constitutes “A central research challenge for the mathematical sciences in the twenty-first century,” according to [157]. Therefore, it was not considered in this thesis. However, an extension that enables integrating diverse datasets obtained from AUVs, buoys, satellites, and other sources can enhance the accuracy of ocean feature estimation and thus facilitate a more holistic approach to ocean exploration and monitoring. Ocean data assimilation is increasingly recognized as crucial for the accuracy of real-time oceanic prediction systems [158]. In fact, given the vast amount of data collected and its variability in space and time, one of the most interesting extensions would be to predict the state of water parameters

occurring in marine environments as in [159].

Improving AUV autonomy is crucial for extending the operational time and covering wider areas. Future work should include developing energy-efficient navigation and control algorithms. For example, energy can be considered a cost in an optimal control problem that generates lower-level control laws as in [160]. Other options include exploring alternative energy sources such as solar-powered [161], or incorporating autonomous recharging through underwater docking stations [162].

The algorithms developed in this thesis have been thoroughly tested through simulations and limited small-scale field experiments. To demonstrate their real-world effectiveness and robustness, there are some necessary extensions, such as large-scale field testing of area scanning missions in diverse oceanic conditions as achieved in [163] and smaller-scale field testing of adaptive sampling strategies as achieved in [164]. For a complete approach to autonomous data collection, there is also the need to research efficient underwater data transmission and communication protocols and sensors [165].

As integrating AUVs into marine research and exploration becomes more prevalent and brings several benefits, as previously discussed, evaluating the potential ecological and geopolitical consequences of AUV deployments becomes imperative. Some possible examples of ecological consequences are underwater noise, habitat disruption, and possible interactions with marine fauna. Some possible examples of geopolitical consequences are maritime security and surveillance and international regulations. There's a need to conduct an in-depth analysis to quantify the extent of these impacts and identify potential mitigation strategies to minimize any adverse effects. Such an analysis would help us design missions and controllers with eco-friendly considerations, promoting sustainable practices for future marine research efforts. However, AUVs are a novel technology, and more research is needed to understand their impact on marine ecosystems. As of 2023, there aren't any published articles that could help us understand this question.

In conclusion, the work presented in this thesis offers significant contributions to optimizing ocean feature estimation and tracking through adaptive sampling and formation control of autonomous underwater vehicles. The future work outlined above provides a roadmap for further advancements in this field, ultimately leading to more efficient, accurate, and environmentally conscious exploration of the world's oceans.

Appendices

Appendix A

Citation analysis

This Section considers the articles cited in this thesis and, more broadly, the article diversity of the decision and control systems and maritime robotics fields. This reflection invites dialogue about intersectional equality and bias in our research fields while introducing awareness of these problems. As with any other STEM field, we can't think of ourselves as disconnected from the world as our research directly impacts innovation, development, and society. It is intertwined with human beings, social life, and future life on earth. That is why it is part of our job to analyze and understand the impact that our research may have. The relevance of this reflection also stems from the lack of data and statistics on diversity in academia and STEM regarding topics such as paper citation and acceptance. First, we focus on the problem of bias in paper acceptance and double-blind review as a possible solution, and second, we focus on the problem of bias in paper citation and offer a reflection on this thesis's and this field's diversity.

2020's president of the IEEE Control Systems Society (CSS), Anuradha Annaswamy, wrote a message for CSS regarding the pertinence of actively discussing this topic [166]. In this message, she noted that, across all IEEE members, women accounted for 30.3% of Students, 20.9% of Graduate Students, 6.4% of Seniors, and 4.5% of Fellows. She reflected that there is a steady attrition of women as their career levels advance from entry-level positions to leadership roles and that it is observed across engineering, the IEEE, and the CSS. She also mentioned that the gradients are small despite these percentages increasing during the past few decades. She considered that the contributing factors include lack of mentorship and role models, uncomfortable work cultures, evaluative practices, discrimination, and harassment directed at women or present in the overall environment, often of sexual nature [167]. This is also

the case in our university, KTH, as the 2022 study indicates that among women students, 6 percent say they have been subjected to “unwanted sexual attention in the place of work/study” [168].

Paper acceptance and double-blind review

In this section, we focus on paper acceptance and diversity bias. Different studies [169] concluded that male researchers exhibited a bias toward citing same-sex authors more frequently, whereas female researchers cited authors of both sexes in proportion to the pool of publications available for being referenced. However, other works [170] have also shown that both sexes tend to evaluate women less favorably in performance contexts. Several studies consider a possible solution for this paper acceptance gap is implementing double-blind review (DBR).

One example is the 2017 study in the journal *Behavioral Ecology* (BE) before and after implementing DBR [171]. The data consists of publications between 1997 and 2005, divided into 1997–2000 (before DBR) and 2002–2005 (after DBR). The results indicate a 7.9% increase in the proportion of women’s first-authored papers after introducing DBR. This study also considered a control group of publications in *BES*, a similar journal that didn’t implement DBR. No significant difference in gender representation was found across the same period in *BES*, which strongly suggests that the change is directly related to the review policy. Another study in 2016 [172] analyzed the impact of introducing DBR in *EvoLang* 11 (*Evolution of Language* conference) by comparing it to *EvoLang* 9 and 10. Similarly, the results indicated a clear difference for female first-authored papers before and after DBR. Namely, for *EvoLang* 9 and 10, male-authored articles scored on average 0.04 above their female counterparts, while *EvoLang* 11 male-authored articles scored on average 0.17 below their female counterparts. Also, a 2013 study [173] investigated how much citation and publication patterns differ between men and women in international relations literature. Data from the Teaching, Research, and International Policy project on peer-reviewed publications between 1980 and 2006 indicates that women are systematically cited less than men after controlling for various factors. This research suggests that an article written by a woman and published in any of the top journals will still receive significantly fewer citations than if a man had written that same article. This is likely because women tend to cite themselves less than men, and men (who make up a disproportionate share of IR

scholars) tend to cite men more than women. A 2020 article [174] considered gender bias in exam grading in Economics at Stockholm University in Sweden. The data collected was from the macroeconomics exam between 2008 and 2014. The results indicate that being graded by an examiner of the same gender increased the exam score by 0.087 standard deviations before DBR. They also suggested that, when separating the effects by grader's gender, both groups of graders favor male students, although male graders favor male students to a larger extent than female graders. In all cases, after DBR, the effect was close to zero.

Paper citation and thesis statistics

This subsection evaluates the diversity of the publications in the present thesis, considering estimated gender and ethnicity [175]. This thesis contains 163 citations, excluding the citations referring to the present section of Citation Analysis. Out of these 163 citations, we considered 148 for analysis. The remaining 15 corresponded to online articles, websites, or others in which the authors were not clearly identified. These 148 citations correspond to mostly academic articles and books, with few exceptions. Each of these citations includes one or more authors. Since most of them are articles, we focus on the first author, which tends to be the article's main author, and on the last author, which tends to be the supervisor or group/lab director. In cases where there's only one author, we consider them both the first and last author. In cases with more than two authors, we disregard the middle ones.

From each publication, we extract the first and last names of the first and last author. For each of these names, we estimate the gender and ethnicity using the `ethnicolr` package [175] and the CleanBib repository [176]. Afterward, we manually checked for inconsistencies and adapted according to the available knowledge. For example, the name Joana was considered female with 99% certainty. However, the ethnicity wasn't as easy to establish, with the following probabilities: 20% white, 0% asian, 88% hispanic, 2% black. Therefore, given the available information, I manually adjusted these probabilities to fit reality better by searching each author. Having said that, the results in this section indicate trends but always correspond to a probability.

Fig. A.1 indicates the results for the gender analysis of this thesis citations. The first observation confirms the expected from the previous field statistics; men dominate the publications with man-first authors at 84% and man-last au-

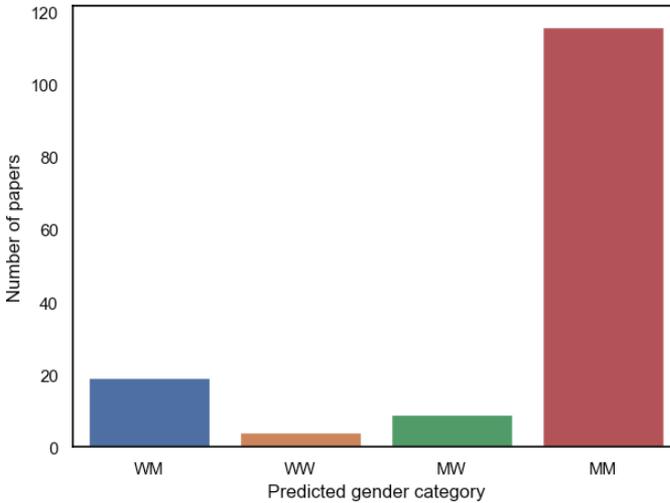


Figure A.1: Analysis of gender diversity in this thesis citations. WM: Woman first author, man last author; WW: Woman first author, woman last author; MW: Man first author, woman last author; MM: Man first author, man last author.

thors at 91%. The second observation is that there's a much stronger incidence of female first authors than last authors. This confirms the previous statistic that women tend to be at higher percentages in entry-level positions, in contrast to leadership positions.

Fig. A.2 indicates the results for the ethnicity and gender analysis of this thesis citations. As in the previous figure, the gender distribution is visible, with men covering most cited publications. Beyond that, there is a clear trend for white men, with white man-first authors covering 54% of all first authors and white man-last authors covering 61% of all last authors.

As previously mentioned, IEEE senior members are 93.6% men, and IEEE fellows are 95.5% men. With this data, we can extrapolate that IEEE last authors would be between 93.6% and 95.5% men. Without data on how much each person publishes, the conclusions are limited. We can only suggest that the values obtained for the present thesis (man last authors at 91%) align with the field's diversity. These results would be especially relevant if compared to the field's standards and historical progress. However, as mentioned at the beginning of this section, such data and corresponding statistics do not exist yet as it is common practice in the field. We suggest following Ph.D. thesis to consider

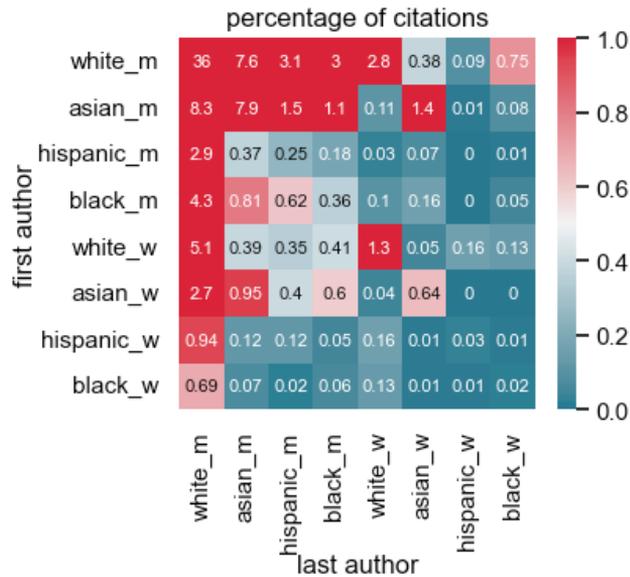


Figure A.2: Analysis of ethnicity and gender diversity in this thesis citations. white_m stands for estimated white man.

running a similar analysis for self-reflection and introducing the possibility of benchmark comparisons in the future.

These results indicate that the papers cited in this thesis are predominantly from white male authors, particularly when considering the last author. Together with the field's statistics, this suggests that the members of our research community may not represent our diverse society very accurately. This limited diversity results in consequences for the research itself, as some research questions may remain unanswered, and novel research paths may remain unexplored. We wonder what future research within control and robotics could become when considering more interpretations, priorities, biases, and topics when writing research papers, defining research proposals, or evaluating research impact.

In conclusion, this section has provided a practical analysis of paper acceptance and its commonly applied solution of double-blind review, as well as valuable data on paper citation that prompted critical self-reflection on the references used in this thesis. By acknowledging and addressing biases in the paper acceptance and citation, the members of our field can contribute to a more in-

clusive and impactful research landscape. Ultimately, a more diverse representation of authors and perspectives will lead to a richer and more well-rounded exploration of the field, fostering innovation and progress.

Bibliography

- [1] *SMHI satellite data on algae*, <https://www.smhi.se/data/oceanografi/algsituationen>.
- [2] I. M. Belkin, P. C. Cornillon, and K. Sherman, “Fronts in large marine ecosystems,” *Progress in Oceanography*, vol. 81, no. 1, pp. 223–236, 2009.
- [3] C. C. Chapman, M.-A. Lea, A. Meyer, J.-B. Sallée, and M. Hindell, “Defining southern ocean fronts and their influence on biological and physical processes in a changing climate,” *Nature Climate Change*, vol. 10, no. 3, pp. 209–219, 2020.
- [4] National Oceanic and Atmospheric Administration, *What is a harmful algal bloom?* <https://www.noaa.gov/what-is-harmful-algal-bloom>.
- [5] S. E. Shumway, J. Burkholder, and S. L. Morton, *Harmful Algal Blooms: A Compendium Desk Reference*. John Wiley & Sons, 2018.
- [6] M. L. Wells, V. L. Trainer, T. J. Smayda, B. S. Karlson, C. G. Trick, R. M. Kudela, A. Ishikawa, S. Bernard, A. Wulff, D. M. Anderson, and W. P. Cochlan, “Harmful algal blooms and climate change: Learning from the past and present to forecast the future,” *Harmful Algae*, vol. 49, pp. 68–93, 2015.
- [7] W. Munk, *Testimony to the U.S. commission on ocean policy*. http://govinfo.library.unt.edu/oceancommission/meetings/apr18_19_02/munk_statement.pdf, 2002.
- [8] P. Wassmann, D. Slagstad, C. W. Riser, and M. Reigstad, “Modelling the ecosystem dynamics of the Barents Sea including the marginal ice zone: II. Carbon flux and interannual variability,” *Journal of Marine Systems*, vol. 59, no. 1, pp. 1–24, 2006.

- [9] P. J. S. Franks, “Recent advances in modelling of harmful algal blooms,” in *Global Ecology and Oceanography of Harmful Algal Blooms*, P. M. Glibert, E. Berdalet, M. A. Burford, G. C. Pitcher, and M. Zhou, Eds. Springer International Publishing, 2018, pp. 359–377.
- [10] *SMHI algal bloom report from satellite*, <https://www.smhi.se/kunskapsbanken/oceanografi/algblomning-1.1734>, 2011.
- [11] H. Loisel, V. Vantrepotte, C. Jamet, and D. N. Dat, “Challenges and new advances in ocean color remote sensing of coastal waters,” *Topics in Oceanography*, pp. 1–38, 2013.
- [12] M. Johansen, *Algal report number 7, 2019*, <https://www.smhi.se/publikationer/publikationer/algrapporter/algrapport-nummer-7-2019-1.150075>, 2019.
- [13] R. Ringbäck, “Multi-agent autonomous target tracking using distance-based formations,” M.S. thesis, KTH, Automatic Control, 2017.
- [14] I. C. Azevedo, A. A. Bordalo, and P. M. Duarte, “Influence of river discharge patterns on the hydrodynamics and potential contaminant dispersion in the Douro estuary (Portugal),” *Water Research*, vol. 44, no. 10, pp. 3133–3146, 2010.
- [15] R. Mendes, N. Vaz, D. Fernández-Nóvoa, J. Da Silva, M. Decastro, M. Gómez-Gesteira, and J. M. Dias, “Observation of a turbid plume using MODIS imagery: The case of Douro estuary (Portugal),” *Remote Sensing of Environment*, vol. 154, pp. 127–138, 2014.
- [16] J. Pinto, R. Mendes, J. C. da Silva, J. M. Dias, and J. B. de Sousa, “Multiple autonomous vehicles applied to plume detection and tracking,” *IEEE OCEANS*, pp. 1–6, 2018.
- [17] J. Hwang, N. Bose, and S. Fan, “AUV adaptive sampling methods: A review,” *Applied Sciences*, vol. 9, no. 15, 2019.
- [18] *Infographic: Ocean sampling technologies*, <https://www.dfo-mpo.gc.ca/about-notre-sujet/publications/infographics-infographies/soto-rceo-national/2020/figure-12-sampling-echantillonnage-eng.html>.

- [19] B. A. Kaiser, M. Hoeberechts, K. H. Maxwell, L. Eerkes-Medrano, N. Hilmi, A. Safa, C. Horbel, S. K. Juniper, M. Roughan, N. T. Lowen, K. Short, and D. Paruru, "The importance of connected ocean monitoring knowledge systems and communities," *Frontiers in Marine Science*, vol. 6, 2019.
- [20] M. Ashphaq, P. K. Srivastava, and D. Mitra, "Review of near-shore satellite derived bathymetry: Classification and account of five decades of coastal bathymetry research," *Journal of Ocean Engineering and Science*, vol. 6, no. 4, pp. 340–359, 2021.
- [21] A. K. Liu, C. Y. Peng, and S.-S. Chang, "Wavelet analysis of satellite images for coastal watch," *IEEE Journal of Oceanic Engineering*, vol. 22, no. 1, pp. 9–17, 1997.
- [22] A. Purser, Y. Marcon, S. Dreutter, U. Hoge, B. Sablotny, L. Hehemann, J. Lemburg, B. Dorschel, H. Biebow, and A. Boetius, "Ocean floor observation and bathymetry system (ofobs): A new towed camera/sonar system for deep-sea habitat surveys," *IEEE Journal of Oceanic Engineering*, vol. 44, no. 1, pp. 87–99, 2018.
- [23] H. Sun, D. C. Hendry, M. A. Player, and J. Watson, "In situ underwater electronic holographic camera for studies of plankton," *IEEE Journal of Oceanic Engineering*, vol. 32, no. 2, pp. 373–382, 2007.
- [24] R. N. Smith and V. T. Huynh, "Controlling buoyancy-driven profiling floats for applications in ocean observation," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 3, pp. 571–586, 2013.
- [25] D. Roemmich, G. C. Johnson, S. Riser, R. Davis, J. Gilson, W. B. Owens, S. L. Garzoli, C. Schmid, and M. Ignaszewski, "The argo program: Observing the global ocean with profiling floats," *Oceanography*, vol. 22, no. 2, pp. 34–43, 2009.
- [26] J. Sherman, R. E. Davis, W. Owens, and J. Valdes, "The autonomous underwater glider "spray"," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 437–446, 2001.
- [27] D. C. Webb, P. J. Simonetti, and C. P. Jones, "Slocum: An underwater glider propelled by environmental energy," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 447–452, 2001.

- [28] C. C. Eriksen, T. J. Osse, R. D. Light, T. Wen, T. W. Lehman, P. L. Sabin, J. W. Ballard, and A. M. Chiodi, "Seaglider: A long-range autonomous underwater vehicle for oceanographic research," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 424–436, 2001.
- [29] N. E. Leonard and J. G. Graver, "Model-based feedback control of autonomous underwater gliders," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 633–645, 2001.
- [30] F. P. Chavez, J. T. Pennington, R. Herlien, H. Jannasch, G. Thurmond, and G. E. Friederich, "Moorings and drifters for real-time interdisciplinary oceanography," *Journal of Atmospheric and Oceanic Technology*, vol. 14, no. 5, pp. 1199–1211, 1997.
- [31] J. S. Willcox, J. G. Bellingham, Y. Zhang, and A. B. Baggeroer, "Performance metrics for oceanographic surveys with autonomous underwater vehicles," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 711–725, 2001.
- [32] J. Das, F. Py, T. Maughan, T. O'reilly, M. Messié, J. Ryan, G. S. Sukhatme, and K. Rajan, "Coordinated sampling of dynamic oceanographic features with underwater vehicles and drifters," *The International Journal of Robotics Research*, vol. 31, no. 5, pp. 626–646, 2012.
- [33] Ö. Özkahraman and P. Ögren, "Efficient navigation aware seabed coverage using AUVs," *IEEE International Symposium on Safety, Security, and Rescue Robotics*, pp. 63–70, 2021.
- [34] P. Yao, L. Qiu, J. Qi, and R. Yang, "AUV path planning for coverage search of static target in ocean environment," *Ocean Engineering*, vol. 241, no. 110050, 2021.
- [35] A. Sivertsen, S. Solbø, R. Storvold, A. Tøllefsen, and K. S. Johansen, "Automatic mapping of sea ice using unmanned aircrafts," *ReCAMP Flagship Workshop Book of Abstracts*, pp. 1–30, 2016.
- [36] J. B. de Sousa and F. L. Pereira, "On the future of ocean observation," in *O mar no futuro de Portugal: Ciência e visão estratégica*, Lisboa: Centro de estudos estratégicos do Atlântico, 2014.
- [37] *LSTS website*, <https://www.lsts.pt>.
- [38] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, 2015.

- [39] A. Lucieer, D. Turner, D. H. King, and S. A. Robinson, "Using an unmanned aerial vehicle (UAV) to capture micro-topography of antarctic moss beds," *International Journal of Applied Earth Observation and Geoinformation*, vol. 27, A:53 – 62, 2014.
- [40] A. Zolich, D. Palma, K. Kansanen, K. Fjörtoft, J. Sousa, K. H. Johansson, Y. Jiang, H. Dong, and T. A. Johansen, "Survey on communication and networks for autonomous marine systems," *Journal of Intelligent & Robotic Systems*, 2018.
- [41] R. M. Murray, "Recent research in cooperative control of multivehicle systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, no. 5, pp. 571–583, 2007.
- [42] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 427–438, 2013.
- [43] M. Egerstedt and X. Hu, "Formation constrained multi-agent control," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 947–951, 2001.
- [44] I. Shames, B. Fidan, and B. D. O. Anderson, "Close target reconnaissance using autonomous UAV formations," *IEEE Conference on Decision and Control*, pp. 1729–1734, 2008.
- [45] D. V. Dimarogonas and K. H. Johansson, "On the stability of distance-based formation control," *IEEE Conference on Decision and Control*, pp. 1200–1205, 2008.
- [46] M. Cao, A. S. Morse, C. Yu, B. D. O. Anderson, and S. Dasgupta, "Controlling a triangular formation of mobile autonomous agents," *IEEE Conference on Decision and Control*, pp. 3603–3608, 2007.
- [47] M. Cao, A. S. Morse, C. Yu, B. Anderson, and S. Dasgupta, "Maintaining a directed, triangular formation of mobile autonomous agents," *Communications in Information and Systems*, vol. 11, no. 1, 2011.
- [48] Z. Sun, *Cooperative Coordination and Formation Control for Multi-agent Systems*. Springer International Publishing, 2018.
- [49] M. Ani Hsieh, V. Kumar, and L. Chaimowicz, "Decentralized controllers for shape generation with robotic swarms," *Departmental Papers (MEAM)*, vol. 26, 2008.

- [50] G. Li, D. St-Onge, C. Pinciroli, A. Gasparri, E. Garone, and G. Beltrame, “Decentralized progressive shape formation with robot swarms,” *Autonomous Robots*, pp. 1–17, 2018.
- [51] N. E. Leonard, D. A. Paley, F. Lekien, R. Sepulchre, D. M. Fratantoni, and R. E. Davis, “Collective motion, sensor networks, and ocean sampling,” *Proceedings of the IEEE*, vol. 95, no. 1, pp. 48–74, 2007.
- [52] G. Gu, P. R. Chandler, C. J. Schumacher, A. Sparks, and M. Pachter, “Optimal cooperative sensing using a team of UAVs,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 4, pp. 1446–1458, 2006.
- [53] A. S. Matveev, H. Teimoori, and A. V. Savkin, “Range-only measurements based target following for wheeled mobile robots,” *Automatica*, vol. 47, pp. 177–184, 2011.
- [54] I. Shames, S. Dasgupta, B. Fidan, and B. D. Anderson, “Circumnavigation using distance measurements under slow drift,” *IEEE Transactions on Automatic Control*, vol. 57, no. 4, pp. 889–903, 2012.
- [55] A. S. Matveev and K. S. Ovchinnikov, “Distributed communication-free control of multiple robots for circumnavigation of a speedy unpredictably maneuvering target,” *IEEE European Control Conference (ECC)*, pp. 1797–1802, 2018.
- [56] R. Zheng and D. Sun, “Circumnavigation by a mobile robot using bearing measurements,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4643–4648, 2014.
- [57] M. Deghat, I. Shames, B. D. O. Anderson, and C. Yu, “Target localization and circumnavigation using bearing measurements in 2D,” *IEEE Conference on Decision and Control*, pp. 334–339, 2010.
- [58] M. Deghat, E. Davis, T. See, I. Shames, B. D. O. Anderson, and C. Yu, “Target localization and circumnavigation by a non-holonomic robot,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1227–1232, 2012.
- [59] M. Deghat, L. Xia, B. D. O. Anderson, and Y. Hong, “Multi-target localization and circumnavigation by a single agent using bearing measurements,” *International Journal of Robust and Nonlinear Control*, vol. 25, no. 14, pp. 2362–2374, 2015.

- [60] A. Boccia, A. Adaldo, D. V. Dimarogonas, M. di Bernardo, and K. H. Johansson, "Tracking a mobile target by multi-robot circumnavigation using bearing measurements," *IEEE Conference on Decision and Control*, pp. 1076–1081, 2017.
- [61] J. O. Swartling, I. Shames, K. H. Johansson, and D. V. Dimarogonas, "Collective circumnavigation," *Unmanned Systems*, vol. 2, no. 3, pp. 219–229, 2014.
- [62] M. Fahad, N. Saul, Y. Guo, and B. Bingham, "Robotic simulation of dynamic plume tracking by unmanned surface vessels," *IEEE International Conference on Robotics and Automation*, pp. 2654–2659, 2015.
- [63] S. Li, Y. Guo, and B. Bingham, "Multi-robot cooperative control for monitoring and tracking dynamic plumes," *IEEE International Conference on Robotics and Automation*, pp. 67–73, 2014.
- [64] A. Franchi, P. Stegagno, and G. Oriolo, "Decentralized multi-robot encirclement of a 3D target with guaranteed collision avoidance," *Autonomous Robots*, vol. 40, 2015.
- [65] S. Bhat and I. Stenius, "Hydrobatics: A review of trends, challenges and opportunities for efficient and agile underactuated AUVs," *IEEE OCEANS*, pp. 1–8, 2018.
- [66] M. Freilich, "Forecasting where ocean life thrives: Scientists focus on seams in the ocean called 'fronts'," *SIAM Journal on Control and Optimization*, vol. 53, no. 2, 2018.
- [67] M. Carlowicz, *Something fishy in the atlantic night—south atlantic ocean*, 2013. [Online]. Available: <https://earthobservatory.nasa.gov/features/Malvinas>.
- [68] *NASA Ocean Color Group*. [Online]. Available: <https://oceancolor.gsfc.nasa.gov/>.
- [69] Y.-H. Ahn, P. Shanmugam, J.-H. Ryu, and J.-C. Jeong, "Satellite detection of harmful algal bloom occurrences in korean waters," *Harmful Algae*, vol. 5, no. 2, pp. 213–231, 2006.
- [70] I. Belkin, J. Sousa, J. Pinto, R. Mendes, and F. López-Castejón, "A new front-tracking algorithm for marine robots," *IEEE/OES Autonomous Underwater Vehicle Workshop*, pp. 1–3, 2018.

- [71] Y. Zhang, C. Rueda, B. Kieft, J. P. Ryan, C. Wahl, T. C. O'Reilly, T. Maughan, and F. P. Chavez, "Autonomous tracking of an oceanic thermal front by a wave glider," *Journal of Field Robotics*, vol. 36, no. 5, pp. 940–954, 2019.
- [72] W. Li, J. A. Farrell, S. Pang, and R. M. Arrieta, "Moth-inspired chemical plume tracing on an autonomous underwater vehicle," *IEEE Transactions on Robotics*, vol. 22, no. 2, pp. 292–307, 2006.
- [73] J. Hwang, N. Bose, H. Nguyen, and G. Williams, "AUV adaptive sampling to delineate subsurface oil plumes," *IEEE/OES Autonomous Underwater Vehicles Symposium*, pp. 1–2, 2020.
- [74] Y. Zhang, B. Kieft, B. W. Hobson, J. P. Ryan, B. Barone, C. M. Preston, B. Roman, B.-Y. Raanan, R. Marin III, T. C. O'Reilly, C. A. Rueda, D. Pargett, K. M. Yamahara, S. Poulos, A. Romano, G. Foreman, H. Ramm, S. T. Wilson, E. F. DeLong, D. M. Karl, J. M. Birch, J. G. Bellingham, and C. A. Scholin, "Autonomous tracking and sampling of the deep chlorophyll maximum layer in an open-ocean eddy by a long-range autonomous underwater vehicle," *IEEE Journal of Oceanic Engineering*, vol. 45, no. 4, pp. 1308–1321, 2020.
- [75] A. Branch, M. M. Flexas, B. Claus, A. F. Thompson, Y. Zhang, E. B. Clark, S. Chien, D. M. Fratantoni, J. C. Kinsey, B. Hobson, B. Kieft, and F. P. Chavez, "Front delineation and tracking with multiple underwater vehicles," *Journal of Field Robotics*, vol. 36, no. 3, pp. 568–586, 2019.
- [76] Y. Zhang, M. Godin, J. Bellingham, and J. Ryan, "Using an autonomous underwater vehicle to track a coastal upwelling front," *IEEE Journal of Oceanic Engineering*, vol. 37, pp. 338–347, 2012.
- [77] R. N. Smith, P. Cooksey, F. Py, G. S. Sukhatme, and K. Rajan, "Adaptive path planning for tracking ocean fronts with an autonomous underwater vehicle," *14th International Symposium on Experimental Robotics*, pp. 761–775, 2016.
- [78] Y. Zhang, J. G. Bellingham, J. P. Ryan, B. Kieft, and M. J. Stanway, "Autonomous four-dimensional mapping and tracking of a coastal upwelling front by an autonomous underwater vehicle," *Journal of Field Robotics*, vol. 33, no. 1, pp. 67–81, 2016.

- [79] D. Kularatne, R. N. Smith, and M. A. Hsieh, "Zig-zag wanderer: Towards adaptive tracking of time-varying coherent structures in the ocean," *IEEE International Conference on Robotics and Automation*, pp. 3253–3258, 2015.
- [80] J. Hwang, N. Bose, H. D. Nguyen, and G. Williams, "Oil plume mapping: Adaptive tracking and adaptive sampling from an autonomous underwater vehicle," *IEEE Access*, vol. 8, pp. 198 021–198 034, 2020.
- [81] Y. Zhang, R. S. McEwen, J. P. Ryan, J. G. Bellingham, H. Thomas, C. H. Thompson, and E. Rienecker, "A peak-capture algorithm used on an autonomous underwater vehicle in the 2010 gulf of mexico oil spill response scientific survey," *Journal of Field Robotics*, vol. 28, no. 4, pp. 484–496, 2011.
- [82] I. Belkin, J. B. D. Sousa, J. Pinto, R. Mendes, and F. Lopez-Castejon, "A new front-tracking algorithm for marine robots," *IEEE/OES Autonomous Underwater Vehicle Workshop*, 2018.
- [83] Y. Zhang, M. A. Godin, J. G. Bellingham, and J. P. Ryan, "Using an autonomous underwater vehicle to track a coastal upwelling front," *IEEE Journal of Oceanic Engineering*, vol. 37, no. 3, pp. 338–347, 2012.
- [84] C. J. Cannell, A. S. Gadre, and D. J. Stilwell, "Boundary tracking and rapid mapping of a thermal plume using an autonomous vehicle," *IEEE OCEANS*, pp. 1–6, 2006.
- [85] C. J. Cannell and D. J. Stilwell, "A comparison of two approaches for adaptive sampling of environmental processes using autonomous underwater vehicles," *IEEE OCEANS*, pp. 1514–1521, 2005.
- [86] J. A. Farrell, S. Pang, and W. Li, "Chemical plume tracing via an autonomous underwater vehicle," *IEEE Journal of Oceanic Engineering*, vol. 30, no. 2, pp. 428–442, 2005.
- [87] J. Harvey, J. Ryan, R. III, C. Preston, N. Alvarado, C. Scholin, and R. Vrijenhoek, "Robotic sampling, in situ monitoring and molecular detection of marine zooplankton," *Journal of Experimental Marine Biology and Ecology*, vol. 413, pp. 1–11, 2012.
- [88] S. Petillo, H. Schmidt, P. Lermusiaux, D. Yoerger, and A. Balasuriya, "Autonomous & adaptive oceanographic front tracking on board autonomous underwater vehicles," *IEEE OCEANS*, pp. 1–10, 2015.

- [89] H. Feng, J. Yu, Y. Huang, J. Qiao, Z. Wang, Z. Xie, and K. Liu, "Adaptive coverage sampling of thermocline with an autonomous underwater vehicle," *Ocean Engineering*, vol. 233, no. 109151, 2021.
- [90] P. Ögren, E. Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment," *IEEE Transactions on Automatic Control*, vol. 49, no. 8, pp. 1292–1302, 2004.
- [91] E. Fiorelli, N. E. Leonard, P. Bhatta, D. A. Paley, R. Bachmayer, and D. M. Fratantoni, "Multi-AUV control and adaptive sampling in Monterey Bay," *IEEE Journal of Oceanic Engineering*, vol. 31, no. 4, pp. 935–948, 2006.
- [92] T. O. Fossum, P. Norgren, I. Fer, F. Nilsen, Z. C. Koenig, and M. Ludvigsen, "Adaptive sampling of surface fronts in the arctic using an autonomous underwater vehicle," *IEEE Journal of Oceanic Engineering*, vol. 46, no. 4, pp. 1155–1164, 2021.
- [93] N. K. Yilmaz, C. Evangelinos, P. F. J. Lermusiaux, and N. M. Patrikalakis, "Path planning of autonomous underwater vehicles for adaptive sampling using mixed integer linear programming," *IEEE Journal of Oceanic Engineering*, vol. 33, no. 4, pp. 522–537, 2008.
- [94] A. A. Bennett and J. J. Leonard, "A behavior-based approach to adaptive feature detection and following with autonomous underwater vehicles," *IEEE Journal of Oceanic Engineering*, vol. 25, no. 2, pp. 213–226, 2000.
- [95] S. McCammon, G. Marcon dos Santos, M. Frantz, T. P. Welch, G. Best, R. K. Shearman, J. D. Nash, J. A. Barth, J. A. Adams, and G. A. Hollinger, "Ocean front detection and tracking using a team of heterogeneous marine vehicles," *Journal of Field Robotics*, vol. 38, no. 6, pp. 854–881, 2021.
- [96] C. Mellucci, P. P. Menon, C. Edwards, and P. Challenor, "Predictive oceanic features tracking with formations of autonomous vehicles," *IEEE Conference on Decision and Control*, pp. 6446–6451, 2016.
- [97] M. Mateus, R. Canelas, L. Pinto, and N. Vaz, "When tragedy strikes: Potential contributions from ocean observation to search and rescue operations after drowning accidents," *Frontiers in Marine Science*, vol. 7, 2020.

- [98] R. N. Smith, Y. Chao, P. P. Li, D. A. Caron, B. H. Jones, and G. S. Sukhatme, "Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean model," *The International Journal of Robotics Research*, vol. 29, no. 12, pp. 1475–1497, 2010.
- [99] J. Das, F. Py, J. B. Harvey, J. P. Ryan, A. Gellene, R. Graham, D. A. Caron, K. Rajan, and G. S. Sukhatme, "Data-driven robotic sampling for marine ecosystem monitoring," *The International Journal of Robotics Research*, vol. 34, no. 12, pp. 1435–1452, 2015.
- [100] T. O. Fossum, J. Eidsvik, I. Ellingsen, M. O. Alver, G. M. Fragoso, G. Johnsen, R. Mendes, M. Ludvigsen, and K. Rajan, "Information-driven robotic sampling in the coastal ocean," *Journal of Field Robotics*, vol. 35, no. 7, pp. 1101–1121, 2018.
- [101] T. D. Dickey, "Emerging ocean observations for interdisciplinary data assimilation systems," *Journal of Marine Systems*, vol. 40, pp. 5–48, 2003.
- [102] O. Schofield, T. Bergmann, P. Bissett, J. F. Grassle, D. B. Haidvogel, J. Kohut, M. Moline, and S. M. Glenn, "The long-term ecosystem observatory: An integrated coastal observatory," *IEEE Journal of Oceanic Engineering*, vol. 27, no. 2, pp. 146–154, 2002.
- [103] T. O. Fossum, G. M. Fragoso, E. J. Davies, J. E. Ullgren, R. Mendes, G. Johnsen, I. Ellingsen, J. Eidsvik, M. Ludvigsen, and K. Rajan, "Toward adaptive robotic sampling of phytoplankton in the coastal ocean," *Science Robotics*, vol. 4, no. 27, 2019.
- [104] K. D. Heaney, G. Gawarkiewicz, T. F. Duda, and P. F. Lermusiaux, "Nonlinear optimization of autonomous undersea vehicle sampling strategies for oceanographic data-assimilation," *Journal of Field Robotics*, vol. 24, no. 6, pp. 437–448, 2007.
- [105] D. Chang, C. R. Edwards, F. Zhang, and J. Sun, "A data assimilation framework for data-driven flow models enabled by motion tomography," *International Journal of Intelligent Robotics and Applications*, vol. 3, no. 2, pp. 158–177, 2019.
- [106] B. Gips, "Texture-based seafloor characterization using Gaussian process classification," *IEEE Journal of Oceanic Engineering*, vol. 47, no. 4, pp. 1058–1068, 2022.

- [107] J. P. Ryan, A. M. Fischer, R. M. Kudela, M. A. McManus, J. S. Myers, J. D. Paduan, C. M. Ruhsam, C. B. Woodson, and Y. Zhang, "Recurrent frontal slicks of a coastal ocean upwelling shadow," *Journal of Geophysical Research: Oceans*, vol. 115, no. C12, 2010.
- [108] R. N. Smith, M. Schwager, S. L. Smith, B. H. Jones, D. Rus, and G. S. Sukhatme, "Persistent ocean monitoring with underwater gliders: Adapting sampling resolution," *Journal of Field Robotics*, vol. 28, no. 5, pp. 714–741, 2011.
- [109] R. Ferrari, "A frontal challenge for climate models," *Science*, vol. 332, no. 6027, pp. 316–317, 2011.
- [110] R. Millet, F. Plumet, and J.-C. Dern, "Autonomous surface vehicle for oceanographic survey," *International Autonomous Surface Ship Symposium*, 2008.
- [111] Z. Liu, Y. Zhang, X. Yu, and C. Yuan, "Unmanned surface vehicles: An overview of developments and challenges," *Annual Reviews in Control*, vol. 41, pp. 71–93, 2016.
- [112] B. D. O. Anderson, "Exponential stability of linear equations arising in adaptive identification," *IEEE Transactions on Automatic Control*, vol. 22, no. 1, pp. 83–88, 1977.
- [113] N. Shimkin and A. Feuer, "Persistency of excitation in continuous-time systems," *Systems & Control Letters*, vol. 9, no. 3, pp. 225–233, 1987.
- [114] L. Farina and S. Rinaldi, *Positive Linear Systems: Theory and Applications*. A Wiley-Interscience publication, 2000.
- [115] J. Wei, X. Yi, H. Sandberg, and K. H. Johansson, "Nonlinear consensus protocols with applications to quantized communication and actuation," *IEEE Transactions on Control of Network Systems*, 2018.
- [116] K. J. Åström, "Adaptive control," in *Mathematical System Theory: The Influence of R. E. Kalman*, Springer Berlin Heidelberg, 1991, pp. 437–450.
- [117] *SINMOD simulation data*, <https://www.sintef.no/en/ocean/initiatives/sinmod/>.
- [118] J. Jurisa, *Columbia river plume*. [Online]. Available: <https://joejurisa.wordpress.com/research/columbia-river-plume/>.

- [119] J. Pinto, P. S. Dias, R. Martins, J. Fortuna, E. Marques, and J. Sousa, "The LSTS toolchain for networked vehicle systems," *IEEE OCEANS*, 2013.
- [120] R. Martins, P. S. Dias, E. R. Marques, J. Pinto, J. B. Sousa, and F. L. Pereira, "IMC: A communication protocol for networked vehicles and sensors," *IEEE OCEANS*, pp. 1–6, 2009.
- [121] F. L. Pereira, J. Pinto, J. Sousa, R. Gomes, G. M. Gonçalves, and P. S. Dias, "Mission planning and specification in the Neptus framework," *IEEE International Conference on Robotics and Automation*, pp. 3220–3225, 2006.
- [122] M. C. Sousa, A. S. Ribeiro, M. Des, R. Mendes, I. Alvarez, M. Gomez-Gesteira, and J. M. Dias, "Integrated high-resolution numerical model for the NW iberian peninsula coast and main estuarine systems," *Journal of Coastal Research*, pp. 66–70, 2018.
- [123] *DUNE unified navigation environment*, <https://lsts.fe.up.pt/software/64>.
- [124] *Communications protocol associated with the LSTS software framework*, <https://www.lsts.pt/docs/imc/master/>.
- [125] *NEPTUS command and control software developed by LSTS*, <https://lsts.fe.up.pt/toolchain/neptus>.
- [126] *Copernicus marine service*, <https://marine.copernicus.eu/about-us/about-eu-copernicus/>.
- [127] *Swedish maritime robotics centre*, <https://smarc.se>.
- [128] *YSI sensors*, <https://www.ysi.com/pro/talpc>.
- [129] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, Ltd, 2011.
- [130] N. Wasmund, "Harmful algal blooms in coastal waters of the south-eastern Baltic sea," in *Baltic Coastal Ecosystems: Structure, Function and Coastal Zone Management*. Springer Berlin Heidelberg, 2002, pp. 93–116.
- [131] E.U. Copernicus Marine Service Information (CMEMS), "Baltic sea biogeochemistry analysis and forecast," [Online]. Available: https://resources.marine.copernicus.eu/product-detail/BALTICSEA_ANALYSISFORECAST_BGC_003_007.

- [132] Ä. Bilaletdin, H. Kaipainen, and T. Frisk, “Dynamic nutrient modelling of a large river basin in Finland,” *WIT Transactions on Ecology and the Environment*, vol. 111, pp. 53–61, 2008.
- [133] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.
- [134] P. Erickson, M. Cline, N. Tirpankar, and T. Henderson, “Gaussian processes for multi-sensor environmental monitoring,” *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pp. 208–213, 2015.
- [135] Scipy implementation of the L-BFGS-B algorithm., <https://docs.scipy.org/doc/scipy/reference/optimize.minimize-lbfgsb.html>.
- [136] S. Bhat, I. Stenius, and T. Miao, “Real-time flight simulation of hydro-batic AUVs over the full 0° – 360° envelope,” *IEEE Journal of Oceanic Engineering*, vol. 46, no. 4, pp. 1114–1131, 2021.
- [137] J. Fonseca, M. Aguiar, J. a. B. d. Sousa, and K. H. Johansson, “Algal bloom front tracking using an unmanned surface vehicle: Numerical experiments based on Baltic sea data,” *IEEE OCEANS*, pp. 1–7, 2021.
- [138] J. Fonseca, J. Wei, T. A. Johansen, and K. H. Johansson, “Cooperative circumnavigation for a mobile target using adaptive estimation,” *CON-TROLO*, pp. 33–48, 2020.
- [139] E.U. Copernicus Marine Service Information (CMEMS), “Baltic sea ocean colour plankton, reflectances, transparency and optics L3 NRT daily observations,” [Online]. Available: https://resources.marine.copernicus.eu/product-detail/OCEANCOLOUR_BAL_CHL_L3_NRT_OBSERVATIONS_009_049.
- [140] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, “ROS: An open-source robot operating system,” *ICRA workshop on open source software*, vol. 3, p. 5, 2009.
- [141] J. Aiken, “A chlorophyll sensor for automatic, remote, operation in the marine environment,” *Marine Ecology Progress Series*, vol. 4, no. 2, pp. 235–239, 1981.
- [142] P. Cieślak, “Stonefish: An advanced open-source simulation tool designed for marine robotics, with a ROS interface,” *IEEE OCEANS*, 2019.

- [143] S. Bhat, I. Stenius, N. Bore, J. Severholt, C. Ljung, and I. Torroba Balmori, "Towards a cyber-physical system for hydrobatic AUVs," *IEEE OCEANS*, pp. 1–7, 2019.
- [144] S. Bhat, I. Torroba, Ö. Özkahraman, N. Bore, C. I. Sprague, Y. Xie, I. Stenius, J. Severholt, C. Ljung, J. Folkesson, and P. Ögren, "A cyber-physical system for hydrobatic AUVs: System integration and field demonstration," *IEEE OCEANS*, 2020.
- [145] C. I. Sprague, Ö. Özkahraman, A. Munafo, R. Marlow, A. Phillips, and P. Ögren, "Improving the modularity of AUV control systems using behaviour trees," *IEEE/OES Autonomous Underwater Vehicle Workshop*, pp. 1–6, 2018.
- [146] Ö. Özkahraman and P. Ögren, "Combining control barrier functions and behavior trees for multi-agent underwater coverage missions," *IEEE Conference on Decision and Control*, 2020.
- [147] I. Stenius, J. Folkesson, S. Bhat, C. I. Sprague, L. Ling, Ö. Özkahraman, N. Bore, Z. Cong, J. Severholt, C. Ljung, A. Arnwald, I. Torroba, F. Gröndahl, and J.-B. Thomas, "A system for autonomous seaweed farm inspection with an underwater robot," *Sensors*, vol. 22, no. 13, 2022.
- [148] M. Jafarian, E. Vos, C. De Persis, J. Scherpen, and A. van der Schaft, "Disturbance rejection in formation keeping control of nonholonomic wheeled robots," *International Journal of Robust and Nonlinear Control*, vol. 26, no. 15, pp. 3344–3362, 2016.
- [149] M. Jafarian, "Robust consensus of unicycles using ternary and hybrid controllers," *International Journal of Robust and Nonlinear Control*, vol. 27, no. 17, pp. 4013–4034, 2017.
- [150] H. Xu and J. Pan, "AUV motion planning in uncertain flow fields using bayes adaptive MDPs," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5575–5582, 2022.
- [151] H. L. Kwa, J. Leong Kit, and R. Bouffanais, "Balancing collective exploration and exploitation in multi-agent and multi-robot systems: A review," *Frontiers in Robotics and AI*, vol. 8, 2022.
- [152] C. Wang, D. Mei, Y. Wang, X. Yu, W. Sun, D. Wang, and J. Chen, "Task allocation for multi-AUV system: A review," *Ocean Engineering*, vol. 266, p. 112911, 2022.

- [153] Q. Cheng, D. Yin, J. Yang, and L. Shen, "An auction-based multiple constraints task allocation algorithm for multi-UAV system," *IEEE International Conference on Cybernetics, Robotics and Control*, pp. 1–5, 2016.
- [154] I. Lončar, A. Babić, B. Arbanas, G. Vasiljević, T. Petrović, S. Bogdan, and N. Mišković, "A heterogeneous robotic swarm for long-term monitoring of marine environments," *Applied Sciences*, vol. 9, no. 7, 2019.
- [155] Y. Zhang, R. S. McEwen, J. P. Ryan, and J. G. Bellingham, "Design and tests of an adaptive triggering method for capturing peak samples in a thin phytoplankton layer by an autonomous underwater vehicle," *IEEE Journal of Oceanic Engineering*, vol. 35, no. 4, pp. 785–796, 2010.
- [156] C. Scholin, G. Doucette, S. Jensen, B. Roman, D. Pargett, R. Marin, C. Preston, W. Jones, J. Feldman, C. Everlove, A. Harris, N. Alvarado, E. Massion, J. Birch, D. Greenfield, R. Vrijenhoek, C. Mikulski, and K. Jones, "Remote detection of marine microbes, small invertebrates, harmful algae, and biotoxins using the environmental sample processor (ESP)," *Oceanography*, vol. 22, no. 2, pp. 158–167, 2009.
- [157] K. Law, A. Stuart, and K. Zygalakis, "Data assimilation," *Cham, Switzerland: Springer*, vol. 214, p. 52, 2015.
- [158] A. M. Moore, M. J. Martin, S. Akella, H. G. Arango, M. Balmaseda, L. Bertino, S. Ciavatta, B. Cornuelle, J. Cummings, S. Frolov, P. Lermusiaux, P. Oddo, P. R. Oke, A. Storto, A. Teruzzi, A. Vidard, and A. T. Weaver, "Synthesis of ocean observations using data assimilation for operational, real-time and reanalysis systems: A more complete picture of the state of the ocean," *Frontiers in Marine Science*, vol. 6, p. 90, 2019.
- [159] J. Anderson and R. N. Smith, "Predicting water properties with Markov random fields for augmented terrain-based navigation in autonomous underwater vehicles," *IEEE OCEANS*, pp. 1–5, 2018.
- [160] M. Sarkar, S. Nandy, S. Vadali, S. Roy, and S. N. Shome, "Modelling and simulation of a robust energy efficient AUV controller," *Mathematics and Computers in Simulation*, vol. 121, pp. 34–47, 2016.

- [161] D. M. Crimmins, C. T. Patty, M. A. Beliard, J. Baker, J. C. Jalbert, R. J. Komerska, S. G. Chappell, and D. R. Blidberg, “Long-endurance test results of the solar-powered AUV system,” *IEEE OCEANS*, pp. 1–5, 2006.
- [162] B. R. Page, R. Lambert, J. Chavez-Galaviz, and N. Mahmoudian, “Underwater docking approach and homing to enable persistent operation,” *Frontiers in Robotics and AI*, vol. 8, 2021.
- [163] J. Pinto, M. Costa, K. Lima, P. Dias, J. a. Pereira, M. Ribeiro, R. Campos, Z. Mirmalek, R. Mendes, F. Castejón, J. Gilabert, M. P. Tomasino, C. Magalhães, J. da Silva, P. Relvas, T. Lukaczyk, K. Skarpnes, M. Ludvigsen, A. Chekalyuk, and K. Rajan, “To boldly dive where no one has gone before: Experiments in coordinated robotic ocean exploration,” *Experimental Robotics*, vol. 19, pp. 472–487, 2021.
- [164] Y. Zhang, J. P. Ryan, B. Kieft, B. W. Hobson, R. S. McEwen, M. A. Godin, J. B. Harvey, B. Barone, J. G. Bellingham, J. M. Birch, C. A. Scholin, and F. P. Chavez, “Targeted sampling by autonomous underwater vehicles,” *Frontiers in Marine Science*, vol. 6, p. 415, 2019.
- [165] G. Schirripa Spagnolo, L. Cozzella, and F. Leccese, “Underwater optical wireless communications: Overview,” *Sensors*, vol. 20, no. 8, p. 2261, 2020.
- [166] A. Annaswamy, “Women in the IEEE control systems society [president’s message],” *IEEE Control Systems Magazine*, vol. 40, no. 2, pp. 8–11, 2020.
- [167] National Academies of Sciences, Engineering, and Medicine, *Sexual Harassment of Women: Climate, Culture, and Consequences in Academic Sciences, Engineering, and Medicine*, P. A. Johnson, S. E. Widnall, and F. F. Benya, Eds. The National Academies Press, 2018.
- [168] Forsknings-och samverkansprogrammet, *Enkätstudie om genusbaserad utsatthet och sexuella trakasserier i svensk högskolesektor*. Stockholm: Karolinska Institutet, KTH, Malmö Universitet, Nationella sekretariatet för genusforskning, 2022.
- [169] S. Knobloch-Westerwick and C. Glynn, “The matilda effect — role congruity effects on scholarly communication,” *Communication Research*, vol. 40, pp. 3–26, 2013.

- [170] K. M. Elsesser and J. Lever, “Does gender bias against female leaders persist? Quantitative and qualitative data from a large-scale survey,” *Human Relations*, vol. 64, no. 12, pp. 1555–1578, 2011.
- [171] A. E. Budden, T. Tregenza, L. W. Aarssen, J. Koricheva, R. Leimu, and C. J. Lortie, “Double-blind review favours increased representation of female authors,” *Trends in Ecology & Evolution*, vol. 23, no. 1, pp. 4–6, 2008.
- [172] S. G. Roberts and T. Verhoef, “Double-blind reviewing at EvoLang 11 reveals gender bias,” *Journal of Language Evolution*, vol. 1, no. 2, pp. 163–167, 2016.
- [173] D. Maliniak, R. Powers, and B. F. Walter, “The gender citation gap in international relations,” *Cambridge University Press*, vol. 67, no. 4, pp. 889–922, 2013.
- [174] J. Jansson and B. Tyrefors, “The genius is a male: Stereotypes and same-sex bias in exam grading in economics at Stockholm University,” Research Institute of Industrial Economics, Working Paper Series 1226, 2020.
- [175] S. Laohaprapanon and S. G. Appeler, “Ethnicolr: Predict race and ethnicity based on the sequence of characters in a name,” *GitHub*, vol. 9, p. 27, 2021.
- [176] D. Zhou, J. Stiso, E. Cornblath, E. Teich, A. Blevins, K. Oudyk, and C. Michael, “Virtualmario, & camp. dalejn/cleanbib: V1. 1.1.3,” *Zenodo*, vol. 10, 2020.

