Doctoral theses at NTNU, 2023:347

Josef Matous

Cooperative Control of Formations of Underwater Vehicles

Doctoral thesis

NTNU
Norwegian University of Science and Technology
Thesis for the Degree of Philosophiae Doctor
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of Science and Technology

Josef Matous

Cooperative Control of Formations of Underwater Vehicles

Thesis for the Degree of Philosophiae Doctor

Trondheim, November 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU

Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the Degree of Philosophiae Doctor

Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

ISBN 978-82-326-7394-0 (printed ver.)
ISBN 978-82-326-7393-3 (electronic ver.)
ISSN 1503-8181 (printed ver.)
ISSN 2703-8084 (online ver.)

IMT-report 2023-20-W

Doctoral theses at NTNU, 2023:347

Printed by NTNU Grafisk senter

Summary

This thesis investigates various control algorithms for marine vehicles. Most of the algorithms proposed in the thesis address the formation path-following problem for a fleet of underactuated autonomous underwater vehicles, although other types of vehicles, such as autonomous surface vehicles and differential drive robots, and other types of control problems, such as collision avoidance, trajectory tracking, and path following, are also considered. The thesis is divided into three parts.

In the first part, we develop a collision avoidance algorithm for overactuated vehicles. The vehicles must reach a desired position while maintaining some minimum safety distance from each other. To solve this problem, we propose an optimization-based control allocation scheme augmented with control barrier functions. Control allocation is a collection of methods for finding an actuator configuration that satisfies a given goal (*e.g.*, reaching a desired position), while control barrier functions allow us to enforce constraints on dynamical systems (*e.g.*, keeping a minimum safety distance). By combining control allocation with control barrier functions, we can create a controller that satisfies a given goal while avoiding collisions. The proposed controller is tested in numerical simulations on two types of autonomous surface vehicles: the milliAmpere ferry, and the Inocean Cat I drillship.

The second part addresses the formation path-following problem. We propose to solve the problem using the null-space-behavioral method. This method allows us to decompose the problem into several tasks. Then, by combining these tasks in a hierarchic manner, we can achieve the desired behavior. To solve the formation path-following problem, we define three tasks: collision avoidance, formation keeping, and path following. In this thesis, we develop and analyze three different null-space-behavioral algorithms for the formation path-following problem. The first algorithm uses a model of an autonomous underwater vehicle with five degrees of freedom. Using Lyapunov analysis, we show that the path-following task is uniformly semiglobally exponentially stable. Numerical simulations then validate this result. The second algorithm uses a six-degree-of-freedom model. Compared to the previous method, this algorithm does not suffer from numerical singularities. This algorithm also contains additional tasks, namely obstacle avoidance and depth limiting. Moreover, we prove that both the path-following and the formation-keeping tasks are uniformly semiglobally exponentially stable. These theoretical results are then validated in numerical simulations. One issue with null-space-behavioral algorithms is that they are centralized. In many applications, centralized algorithms are difficult to implement, as they require a central node or an agent that can communicate and coordinate with other agents in real-time. To solve this issue,

the third algorithm combines the null-space-behavioral method with consensus, resulting in a fully distributed controller. We propose two types of consensus algorithms. First, we propose a continuous-time consensus algorithm and prove its stability using Lyapunov analysis. Then, we present a modified discrete-time version of the algorithm based on event-triggered control. The effectiveness of both the continuous- and discrete-time algorithms is demonstrated in numerical simulations. Furthermore, the discrete-time version is also tested in field experiments.

The third part of the thesis extends the hand position approach to underactuated underwater vehicles moving in three dimensions. This approach was originally developed to stabilize nonholonomic vehicles. By treating the hand position of the vehicle as the output of the system, we can use input-output feedback linearization to transform the underactuated highly nonlinear vehicle model into a system with linear external dynamics and nonlinear internal dynamics. We analyze the closed-loop behavior of a generic hand position-based controller and present four applications of the hand position approach. First, we use this approach to solve the trajectory-tracking and path-following problems. We propose simple PID-based controllers to solve these problems and show that using these controllers renders the external states globally exponentially stable, while the internal states remain bounded. The theoretical results are validated in numerical simulations as well as field experiments. Next, we present a spline-based model predictive control method for solving the formation path-following problem. The proposed method is not restricted to the hand position approach only. In fact, the method is applicable to any vehicle with a differentially flat model. To demonstrate this, we present two case studies: underwater vehicles with the hand position controller, and differential drive robots. Next, we use the hand position concept to solve the tracking-in-formation problem for a fleet of autonomous underwater vehicles. The proposed method combines consensus with barrier Lyapunov functions, allowing the fleet to reach the desired formation while avoiding collisions and maintaining connectivity. We show that the closed-loop system is almost-everywhere uniformly asymptotically stable and that the output error dynamics converge to the origin exponentially fast while satisfying the constraints. The theoretical results are verified in numerical simulations. Finally, we combine the hand position approach with null-space-behavioral control. Specifically, we extend the null-space-behavioral algorithm, which was originally developed for first-order kinematic systems, to second-order systems. Similarly to our previous work, we then design the path-following, formation-keeping, and collision-avoidance tasks, so that the fleet can follow a given path in a formation while avoiding collisions. We prove the stability of the control scheme using Lyapunov analysis and verify its effectiveness in simulations.

Preface

This thesis is submitted in partial fulfillment of the requirements for the degree of Philosophiae Doctor (PhD) at the Norwegian University of Science and Technology (NTNU), Trondheim.

The work presented has been conducted at the Department of Engineering Cybernetics (ITK), NTNU, as part of the Autonomous Underwater Fleets (AUF) project. Professor Kristin Ytterstad Pettersen from the ITK has been the main supervisor. Professor Damiano Varagnolo from the ITK and Department of Information Engineering, University of Padova, Italy, and Dr. Claudio Paliotta from SINTEF Digital and Aker Solutions have been the co-supervisors. This work was partly supported by the Research Council of Norway through project No. 302435 and the Centres of Excellence funding scheme, project No. 223254.

Acknowledgments

First and foremost, I would like to thank my supervisor, Kristin. Not only did she give me this opportunity, but she has also been an excellent supervisor, always positive and optimistic, gently nudging me towards the right direction. Furthermore, she possesses an incredible ability to refine my very coarse ideas into something presentable.

I would also like to thank my co-supervisors, Damiano and Claudio, for taking the time off their very busy schedules to guide and help me. I remember that on multiple occasions, Damiano took a half-day off just to give me detailed feedback on my work, and Claudio has continued to work with me even after leaving SINTEF and getting a new job at Aker Solutions. Thank you both for that.

I also must not forget to thank my former supervisors from the Czech Technical University in Prague, Dr. Tomáš Michálek and Dr. Martin Gurtner, as well as the head of the research group, Professor Zdeňek Hurák, for helping me to find the position at NTNU and encouraging me to apply.

The results presented in this thesis are not a work of a single person. To this end, I would like to thank everyone who contributed to the presented results. This includes my supervisors, as well as Erlend Basso and Emil Thyri, who helped me with my very first scientific paper, Esteban Restrepo, who visited NTNU for two months in 2022 and with whom we have had a fruitful collaboration ever since, and Erling Lie, whose brilliant Master's project became a conference paper.

Demonstrating that the presented theory actually works typically requires some nice plots and results. While numerical simulations are nice, there is no substitute

for field experiments with very expensive and fragile hardware placed in a hostile environment. To this end, I would like to thank Professor João Tasso de Figueiredo Borges de Sousa and the engineering team at Laboratório de Sistemas e Tecnologias Subaquáticas (LSTS), Faculty of Engineering, University of Porto, and Professor Martin Ludvigsen, Dr. Antonio Vasilijevic, and the engineering team at the Applied Underwater Robotics Laboratory (AUR-lab), NTNU, for providing access to their underwater vehicles and facilities, and invaluable support before and during the experiments. In addition, I wish to thank Tore Mo-Bjørkelund from the AUR-lab and Renato Campos from OceanScan-MST for their help with deploying the control algorithms to the vehicles.

Furthermore, I would like to thank all my friends and colleagues at NTNU for helping me to survive these incredible three years. Last but not least, I would like to thank my family, my father Josef, my mother Vladislava, and my sister Vladimíra, for their support and the relaxing summer and Christmas holidays.

Contents

Abbreviations

ADMM alternating direction method of multipliers.
ASV autonomous surface vehicle.
AUF Autonomous Underwater Fleets.
AUV autonomous underwater vehicle.
  LAUV light autonomous underwater vehicle.

BLF barrier Lyapunov function.

CBF control barrier function.
CLF control Lyapunov function.
CLIK closed-loop inverse kinematics.
  SOCLIK second-order closed-loop inverse kinematics.
COLAV collision avoidance.

DOF degree of freedom.
DUNE Unified Navigation Environment.

FOV field-of-view.

GES global exponential stability.

ITK Department of Engineering Cybernetics.

LOS line-of-sight.

MPC model predictive control.

NED north-east-down.
NSB null-space-behavioral.
NTNU Norwegian University of Science and Technology.

ODE ordinary differential equation.

PhD Philosophiae Doctor.

QP quadratic program.

ROV remotely operated vehicle.

UGAS uniform global asymptotic stability.
USGES uniform semiglobal exponential stability.

List of Symbols

$\mathbb{R}$	The set of real numbers.
$\mathbb{Z}_{\geq 0}$	The set of nonnegative integers.
$\mathbb{R}^n$	The set of n-dimensional real vectors.
$\mathbb{R}^{n\times m}$	The set of n-by-m real matrices.
$SO(3)$	The special orthogonal group of dimension 3.
$\mathfrak{so}(3)$	The space of 3-by-3 skew-symmetric matrices, and the Lie algebra of $SO(3)$.
$\mathfrak{B}_r^n$	The closed ball of radius r ($\mathfrak{B}_r^n = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{x}\| \leq r\}$).
$\partial\mathcal{M}$	The boundary of an open set $\mathcal{M} \subset \mathbb{R}^n$.
$\mathbf{I}_n$	The n-by-n identity matrix.
$\mathbf{O}_{n\times m}$	The n-by-m null matrix.
$\mathbf{0}_n$	The n-dimensional null vector.
$\mathbf{1}_n$	The n-dimensional vector of ones.
$\mathbf{1}_{n\times m}$	The n-by-m matrix of ones.
$\|\mathbf{x}\|$	The Euclidean norm of a vector $\mathbf{x} \in \mathbb{R}^n$.
$\|\mathbf{x}\|_{\mathcal{M}}$	The Euclidean distance between a point $\mathbf{x}$ and a set $\mathcal{M}$ ($\|\mathbf{x}\|_{\mathcal{M}} = \inf_{\mathbf{y}\in\mathcal{M}} \|\mathbf{x} - \mathbf{y}\|$).
$\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$	The standard basis of $\mathbb{R}^3$ ($\mathbf{e}_1 = [1,0,0]^{\mathrm{T}}$, $\mathbf{e}_2 = [0,1,0]^{\mathrm{T}}$, $\mathbf{e}_3 = [0,0,1]^{\mathrm{T}}$).
$\mathbf{A}^\dagger$	The Moore-Penrose pseudoinverse of a matrix $\mathbf{A} \in \mathbb{R}^{n\times m}$.
$\mathbf{A} \otimes \mathbf{B}$	The Kronecker tensor product of matrices $\mathbf{A} \in \mathbb{R}^{n_a \times m_a}$ and $\mathbf{B} \in \mathbb{R}^{n_b \times m_b}$.
$\arctan_2$	Four-quadrant inverse tan; $\arctan_2(y, x)$ is equivalent to the phase of the complex number $x + iy$.

Chapter 1

Introduction

In the beginning the Universe was created.
This has made a lot of people very angry
and been widely regarded as a bad move.

— Douglas Adams, "The Restaurant at the End of the Universe," 1980.

This thesis presents and studies control algorithms for autonomous marine vehicles. The thesis is mainly focused on formation path-following control of underactuated autonomous underwater vehicles (AUVs), although some chapters also consider autonomous surface vehicles (ASVs) and additional control problems, such as reactive collision avoidance, path following, and trajectory tracking. The thesis studies three main topics:

- the unification of reactive collision avoidance and control allocation (Chapter 3),
- formation path following using the null-space-behavioral algorithm (Part I, Chapters 4–6),
- the hand position concept and its applications (Part II, Chapters 7–11).

This chapter presents the motivation and the contributions of the thesis.

1.1 Literature Review and Motivation

In this section, we present a general introduction to the problems studied in the thesis, as well as an overview of the existing literature.

1.1.1 Marine Robots

In this section, we briefly introduce the three main types of marine robots and their use. Marine robots (*i.e.,* unmanned marine vehicles) are being increasingly used in a variety of applications such as transportation [1, 2], inspection, maintenance and repair [3, 4], mapping of underwater structures, *e.g.,* shipwrecks [5], and various oceanographic and environmental missions such as tracking of oil spills [6] and

harmful algal blooms [7]. These vehicles often operate in environments that are inaccessible or dangerous to humans, such as the deep sea or the Arctic.

Marine robots can generally be split into three categories: autonomous surface vehicles (ASVs), remotely operated vehicles (ROVs), and autonomous underwater vehicles (AUVs). ASVs are also referred to as *autonomous ships*, since their design is commonly based on surface vessels such as kayaks [8], catamarans [9, 10], and miniature ferries [2]. ASVs are used in scientific missions as well as transportation. ROVs, also referred to as *underwater drones*, are small, box-shaped vehicles with thrusters. ROVs are typically *fully actuated*, meaning that the configuration of thrusters allows the vehicle to move and rotate in any direction. ROVs are designed for low speeds, often less than 1.5 meters per second, and due to the high power demands of the thrusters, the vehicles can only operate for a few hours [11]. Consequently, ROVs are used for short-term inspection, maintenance and repair missions. To complete these missions, ROVs are connected to an operator via a series of cables, referred to as the *tether*.

Conversely, AUVs are able to operate independently and without any connecting cables. There are various types of AUVs. This thesis studies slender, torpedo-shaped AUVs with a propeller that provides forward (surge) thrust, and fins that provide torque. This configuration of actuators means that these AUVs are *underactuated*, as we cannot directly control the lateral (sway and heave) velocities. Compared to ROVs, AUVs can reach higher speeds and operate longer [12, 13], making them suitable for long-term oceanographic missions.

1.1.2 Control Problems for AUVs

This section presents the various control problems for AUVs studied in this thesis.

The Trajectory-Tracking and Path-Following Problems

Arguably, the trajectory-tracking and path-following problems are the most interesting and significant ones, since many high-level mission planners assume that the vehicle is able to follow a given path or trajectory.

For the purposes of this thesis, a path is a curve (*i.e.,* a one-dimensional object in two- or three-dimensional Euclidean space), while a trajectory is a time-varying reference position. In the literature, it is often stated that a trajectory is a "path with temporal constraints" [14]. Consequently, in trajectory tracking, the desired position of the vehicle for a given time is fixed, while in path following, we have some freedom in choosing which point on the desired path should be followed at a given time. A more detailed discussion on the differences between trajectory tracking and path following is presented in Section 2.3.

To solve the trajectory-tracking problem, numerous methods based on backstepping [15, 16], sliding-mode control [17], control Lyapunov functions (CLFs) [18], and model predictive control (MPC) [19] have been proposed.

To solve the path-following problem, most controllers utilize line-of-sight (LOS) guidance. In [20], an integral LOS guidance scheme is used to counteract the sea loads, [21] combines LOS guidance with an adaptively tuned PID controller, and [22] uses LOS with active disturbance rejection control.

The Formation Path-Following Problem

In many applications, it is advantageous to perform the tasks with a group of cooperating AUVs. Compared to a single vehicle, a group of AUVs can cover a larger area (*e.g.,* in inspection and oceanographic tasks). A group is also more flexible, able to reconfigure if the parameters of the mission change, and able to complete the task even if one or more AUVs fail. Many of the aforementioned applications can be formulated as a formation path-following problem, *i.e.,* a problem of steering a group of AUVs along a predefined path in a given formation.

As presented in [23], there exists a plethora of formation path-following methods, most of them based on two concepts: coordinated path-following [24, 25] and leader-follower [26, 27]. In the *coordinated path-following* approach, each vehicle follows a predefined path separately. Formation is then achieved by coordinating the motion of the vehicles along these paths. In this approach, the formation-keeping error (*i.e.,* the difference between the actual and desired relative position of the vehicles) may initially grow as the vehicles converge to their predefined paths. In the *leader-follower* approach, one leading vehicle tracks the given path while the followers adjust their speed and position to obtain the desired formation shape, relative to the leader.

Both the coordinated path-following and the leader-follower method can be solved using model predictive control [28, 29]. MPC is a model-based optimal control method that allows us to enforce constraints on the inputs and states of the vehicles. However, most MPC methods are based on sampling. Consequently, any constraints on the inputs or states can only be enforced at discrete-time instances. In other words, we have no control over the behavior of the system between the samples. We can mitigate this issue by decreasing the sampling time. However, by decreasing the sampling time, we increase the number of optimized variables, thus increasing the computational requirements.

In recent years, researchers have focused on computationally tractable MPC schemes. One possibility of reducing the computational requirements is to parametrize the vehicles' trajectories using splines. A spline-based path-planning MPC algorithm for first-order nonholonomic vehicles was proposed [30]. The algorithm solves the point-to-point formation tracking problem with static obstacles. Another spline-based MPC algorithm was proposed in [31]. This algorithm is applicable to a wider range of systems compared to [30], and it has been demonstrated on point-to-point and trajectory-tracking problems.

Another method that can be applied to the formation path-following problem is the so-called null-space-behavioral (NSB) algorithm [32–35]. In the NSB framework, the control objective is expressed using multiple tasks. By combining several simple tasks, the vehicles can exhibit the desired complex behavior. In the literature, there are many examples of NSB algorithms applied to kinematic vehicles [32] and marine vehicles moving in the horizontal plane [33–35].

However, the standard NSB algorithm is centralized, meaning that in a real-life application, there must be a central node that communicates and coordinates with all the vehicles. While underwater, the AUVs typically communicate via acoustic modems. These modems have low bandwidth and significant delays, making them unsuitable for real-time control. A distributed NSB algorithm was proposed in

[36]. In this algorithm, the group of vehicles is split into smaller, fully connected subgroups. Each subgroup performs the standard, centralized NSB algorithm. The proposed method is limited to static formations. A similar scheme was proposed in [37], where a group of heterogenous surface vehicles is split into homogenous subgroups. Each subgroup has one leading vehicle that exchanges information with the leaders of other subgroups.

1.1.3 Safety Constraints

This section presents the various constraints that need to be considered when deploying autonomous vehicles.

Collision Avoidance

Autonomous vehicles are often used in cluttered and unpredictable environments where considerations to other vehicles and obstacles need to be made. Therefore, the control system of autonomous vehicles should include some form of collision avoidance (COLAV).

Reviews of various COLAV concepts are presented in [38–40]. In general, algorithms for COLAV can be split into two categories: motion planning and reactive algorithms.

Motion planning algorithms include, among others, various types of path planning algorithms [41–44], the dynamic window algorithm [45], and MPC. MPC can be used both for a single vehicle [46, 47] and for multi-agent systems in a distributed form [48, 49]. Some motion planning algorithms also include consideration of relevant traffic protocols that apply in the given domain, *e.g.,* the regulations for marine vehicles known as COLREGs [41–43].

Reactive algorithms for COLAV include, among others, virtual potential fields [50], geometric guidance [51], and control barrier functions (CBFs) [52–56]. Reactive algorithms are often used together with motion planning algorithms in a hybrid controller. In such a controller, the reactive algorithm ensures the safety of the vehicle in unexpected situations. Such an algorithm is proposed in [57], where a collision-free velocity reference is obtained through numerical optimization.

CBFs offer a COLAV method that is applicable to a wide range of systems [58]. In the literature, there are typically two ways in which CBFs are applied for COLAV. They are either applied to a simplified model of the vehicle (*e.g.,* a unicycle model [52, 53]) to provide safe velocity references, or they are used together with CLFs [54–56] on the complete model.

Reactive COLAV methods that work with a simplified model do not take into account the physical limitations of the vehicle, such as acceleration or actuator constraints. Consequently, these methods may output reference signals that the underlying controllers cannot track. To mitigate this, reactive COLAV methods should be included at the lowest-possible control level.

Connectivity Maintenance

In addition to COLAV, autonomous vehicles often need to maintain a sufficiently close distance to each other to guarantee the reliability of the communication and the connectivity of the multi-agent system. In special cases when the vehicles use optical sensors or communications, the vehicles are also limited by field-of-view (FOV) constraints.

Many works in the literature address the coordination problem of multiple marine vehicles under such inter-agent constraints. In [59, 60] planning-based methods are developed to generate trajectories that satisfy the constraints. However, planning algorithms usually require *a priori* knowledge of the environment, which might be unrealistic in highly dynamical environments, such as under water. Reactive algorithms are based, *e.g.*, on artificial potential fields [61, 62] and barrier Lyapunov functions (BLFs) [63, 64].

1.2 Outline and Contributions

Chapter 3: Unifying Reactive Collision Avoidance and Control Allocation

This chapter differs from the rest of the thesis in the considered vehicle model. In this chapter, we consider overactuated vehicles, *i.e.,* vehicles with more actuators than degrees of freedom (DOFs). As mentioned in Section 1.1, reactive collision avoidance should be included at the lowest-possible control level. Overactuated vehicles often use control allocation in their lowest-level controller [65]. Most control allocation methods are based on numerical optimization [66–68] which makes them ideal for augmenting with control barrier functions (CBFs).

The main contribution of Chapter 3 is a reactive collision avoidance (COLAV) algorithm that is included at the lowest level in the control pipeline, *i.e.* in the control allocation, to ensure the safety of the vehicle regarding collision avoidance. Since it is included at the lowest-possible control level, it also ensures the "baseline" safety of any other higher level (long term/deliberate) planners of the vehicle guidance, navigation and control system. The algorithm can easily be implemented on vehicles that apply a numerical optimization-based method to control allocation. Moreover, the algorithm does not rely on any communication between the vehicles; the only required information is the position and velocity of other vehicles. The chapter extends the results in [69], which only considers autonomous surface vehicles (ASVs) and simple encounters between one ASV and a vessel moving at a constant course and speed, making the method applicable to a wider range of vehicles and scenarios with multiple autonomous vehicles.

Chapter 4: Formation Path-Following Control of 5DOF Underactuated AUVs

As mentioned in Section 1.1, the NSB algorithm has been applied to kinematic vehicles (*i.e.,* vehicles with single-integrator dynamics), as well as ASVs and AUVs moving in the horizontal plane. Chapter 4 extends the NSB algorithm to AUVs moving in three dimensions.

Specifically, the chapter extends the results of [35], where an NSB algorithm is used to guide two ASVs, by proposing an algorithm that works with an arbitrary number of AUVs with five degrees of freedom (DOFs) moving in 3D. We solve the formation path-following problem by defining three tasks: collision avoidance, formation keeping, and path following. The tasks are combined using the NSB algorithm to achieve the desired behavior. Similarly to [35], we solve the path-following task using LOS guidance. Using the cascaded systems theory results of [70], we prove that the closed-loop system consisting of a 3D LOS guidance law, combined with surge, pitch, and yaw autopilots based on [71], is uniformly semiglobally exponentially stable (USGES) and uniformly globally asymptotically stable (UGAS). The theoretical results are verified through numerical simulations.

Chapter 5: Formation Path-Following Control of 6DOF Underactuated AUVs

This chapter further extends the NSB algorithm proposed in Chapter 4. The algorithm in Chapter 4 uses a 5DOF AUV model, considers only inter-vehicle collision avoidance, and proves only the stability of the path-following task. Furthermore, the orientation of the 5DOF model was expressed using Euler angles, which causes singularities for a pitch angle of ± 90 degrees.

This work applies the NSB algorithm to a full 6DOF model, uses rotation matrices to describe the attitude of the vehicles to avoid singularities, modifies and extends the tasks, and proves the stability of the combined path-following and formation-keeping tasks. We also add a scheme for obstacle avoidance and a scheme that keeps the vehicles within a given range of depths. As opposed to the previous work, we do not limit the analysis to a specific low-level attitude controller. Consequently, the new algorithm can be integrated into existing on-board controllers. Assuming that the existing low-level controller allows exponential tracking, we use results from cascaded systems theory [70] to prove that the closed-loop system composed by the NSB algorithm and the low-level controller is uniformly semiglobally exponentially stable (USGES). We verify the results in numerical simulations.

Chapter 6: A Distributed NSB Algorithm for Formation Path Following

The algorithms presented in Chapters 4 and 5 are centralized, making them difficult to use in real-life applications. As mentioned in Section 1.1, there are distributed NSB algorithms [36, 37]. However, these algorithms work by dividing the fleet into smaller, fully connected subgroups, or by using leading vehicles. In both approaches, there is still a requirement for fast and reliable communications within the subgroups and between the leaders. Furthermore, the leader-follower scheme is vulnerable to failures of the leading vehicles.

Chapter 6 presents an approach that is fully distributed, so that the fleet does not need to decompose into subgroups nor requires leading vehicles. To do so, the proposed algorithm combines the centralized schemes presented in Chapters 4 and 5 with a consensus algorithm. Specifically, we first propose a continuous-time consensus algorithm and prove its stability using Lyapunov analysis. Then, we present a modified discrete-time version of the algorithm based on event-triggered

control. The effectiveness of both the continuous- and discrete-time algorithms is demonstrated in numerical simulation. Furthermore, the discrete-time version is also tested in field experiments.

Chapter 7: Hand Position for Underactuated Underwater Vehicles

This thesis studies slender torpedo-shaped autonomous underwater vehicles (AUVs) with a propeller that provides forward (surge) thrust, and fins that provide torque. The configuration of actuators means that AUVs are *underactuated*, as we cannot directly control the lateral (sway and heave) velocities. Most control algorithms use the so-called *neutral point* of the vehicle as the output of the system (see Figure 1.1a). The neutral point is a location on the x-axis (the stern-bow axis) of the vehicle such that, if chosen as the origin of the vehicle's body-fixed coordinate frame, the lateral motion of the vehicle is not affected by its control inputs. Due to the underactuated nature of the AUV, controlling the neutral point requires specialized algorithms, such as line-of-sight guidance [20–22, 72]. In this chapter, we propose to use the *hand position* concept to control the AUV. The hand position is a point located a given distance in front of the neutral point along the vehicle's x-axis (see Figure 1.1b for an illustration). The concept was first introduced in [73] to stabilize nonholonomic vehicles with unicycle dynamics. Later, it was applied to control formations of unicycles [74]. The concept was then extended to marine vehicles moving in the horizontal plane [14], and two- and three-dimensional Euler-Lagrange-like systems [75, 76].

There are two main advantages to using the hand position concept. The first advantage stems from the applications of AUVs. The aim of many scientific missions is to scan a given area using a sensor attached to the AUV. Since the position of the sensor typically does not coincide with the neutral point, there may be a significant offset between the sensor and the desired trajectory, caused by the sea loads (see Figure 1.1a). In some cases, the hand position can be chosen such that it coincides with the position of the sensor, allowing to scan the area more accurately. The second advantage is that if we choose the hand position as the output of our system, it is possible to transform the nonlinear underactuated vehicle model to a double integrator, using output feedback linearization. This allows us to apply advanced control strategies, *e.g.,* various consensus algorithms [74–77] that cannot be directly used on nonholonomic or underactuated vehicles.

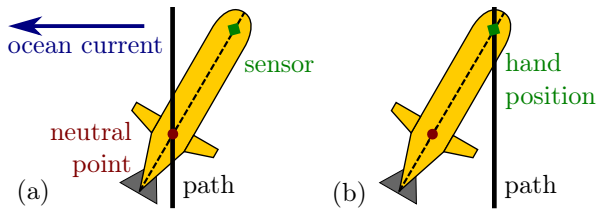


Figure 1.1: Illustration of (a) the traditional approach where the aim is to control the neutral point of the vehicle, and (b) the proposed hand position-based approach. The dashed line represents the body-fixed x-axis.

Note that the three-dimensional Euler-Lagrange-like system used in [76] does not accurately represent AUVs, since it does not consider the Coriolis and centripetal effects or the restoring forces (gravity and buoyancy). Furthermore, the model in [76] has five degrees of freedom (DOFs): three position coordinates, pitch angle, and yaw angle. The use of Euler angles inherently introduces singularities into the system.

The goal of this chapter is thus to further extend the hand position concept to AUVs moving in three dimensions. We employ a more realistic AUV model than in [76]. We model the full 6DOF motion and use rotation matrices to describe the orientation of the vehicle, thus avoiding singularities. Using Lyapunov analysis, we derive the sufficient conditions for boundedness of the internal states, *i.e.,* the orientation and the angular velocities, for a generic hand position-based controller.

Chapter 8: Trajectory Tracking and Path Following using the Hand Position Concept

In this chapter, we use the hand position concept to solve the trajectory-tracking and path-following problems. Specifically, we show that these two problems can be solved using the hand position transformation and a simple PID controller. We analyze the closed-loop behavior of the system and prove that the proposed controllers exponentially track the desired trajectory or path, while the angular velocities of the vehicle remain bounded. Moreover, we prove that in the special case when the desired trajectory or path is a straight line, the whole closed-loop system is exponentially stable. The theoretical results are verified both in numerical simulations and experiments.

Chapter 9: Distributed MPC for Formation Path-Following of Multi-Vehicle Systems

In this chapter, we employ the spline-based MPC presented in [30, 31] to solve the formation path-following problem. The proposed scheme is applicable to a wide range of vehicles. The only restriction is that the model of the vehicle must be differentially flat. The spline-based MPC scheme can thus be seen as a trade-off between lower computational requirements and more restrictive assumptions on the model.

We present the results of two numerical case studies. The first case study considers a fleet of AUVs. To make the AUV model differentially flat, we employ the hand position transformation. The second case study considers a group of differential drive robots modeled as first-order unicycles.

Chapter 10: Control of AUVs Under Hard and Soft Constraints

This chapter investigates the tracking-in-formation problem for a fleet of AUVs. This problem is similar to the formation path-following problem, except that the fleet should track a desired trajectory instead of following a desired path. We assume that the AUVs communicate over a directed topology and are subject to the COLAV and connectivity constraints discussed in Section 1.1.

Under the control designs proposed in the literature, in many instances, in order to guarantee the satisfaction of the inter-agent constraints, the vehicles are forced to move backwards, oftentimes during a prolonged period of time and at relatively high speeds (for backwards motion of a marine vehicle). However, although marine vehicles are able to move backwards, they are not well-suited to do so due to their shapes and their propulsion system. This issue, however, has not been addressed in the literature of multi-AUV systems.

In Chapter 10, we propose a distributed control law that solves the tracking-in-formation problem for multiple marine vehicles interacting over a directed communication graph and that guarantees, simultaneously, connectivity preservation and inter-agent collision avoidance. Moreover, we address the issue of backwards motion by imposing a non-negativity constraint on the surge velocity of the vehicles. More precisely, on one hand we encode via BLFs the proximity and safety constraints as *hard constraints* that need to be always satisfied. On the other hand, we encode the non-negativity of the surge velocity as a *soft constraint*, so that it is imposed on the vehicles as long as it does not interfere with the hard constraints, in which case it is dynamically relaxed. The proposed controller is based on the hand-position-based input-output feedback linearization method presented in Chapter 7 and on the so-called *edge-agreement* representation of multi-agent systems [78], in which the relative states of the connected agents are used instead of the absolute ones, making it well adapted to practical applications where, usually, only relative measurements are available. With regards to the stability analysis, differing from most of the existing works in the literature, where only non-uniform convergence to the formation and to the target vehicle is guaranteed, we establish *almost-everywhere* uniform asymptotic stability of the tracking-in-formation objective and we show that the output error dynamics converge to the origin exponentially fast, while satisfying the constraints.

Chapter 11: Combining NSB with the Hand Position Approach

This chapter presents an extended null-space-behavioral (NSB) algorithm for vehicles with second-order dynamics. The NSB algorithm, as presented in the existing literature, is developed for kinematic single-integrator systems [33, 35, 79]. Although existing NSB methods are developed for first-order systems, AUV dynamics are inherently second-order. Therefore, any first-order solution is necessarily perturbed by the dynamics of the maneuvering controller. In Chapter 11, we extend the NSB method to vehicles with double integrator dynamics and propose an algorithm that uses the second-order closed-loop inverse kinematics equation to control the task variables through desired acceleration. The procedure is inspired by robotic manipulators, where second-order methods are more common, due to the inherent second-order dynamics of mechanical systems [80, 81]. Compared to the existing methods, our formulation handles the second-order dynamics directly in the task space as interpretable mass-spring-damper systems.

We apply the proposed NSB method to a fleet of AUVs. To make the proposed method applicable to AUVs, we use the hand position transformation. Subsequently, through the design of specific path-following, formation-keeping, and collision-avoidance tasks, we can control the fleet to follow a given path in a for-

mation while avoiding collisions both within the fleet and with external obstacles. We prove the stability of the control scheme using Lyapunov analysis and verify its effectiveness in simulations. Because our reformulated NSB method works directly with the second-order system, there is no need to transform desired velocities or accelerations into surge and orientation references, as has been done in previous works. This reduces one level of complexity in the controller design.

1.3 Publications

Conference Publications

[82] J. Matouš, E. A. Basso, E. H. Thyri, and K. Y. Pettersen, "Unifying reactive collision avoidance and control allocation for multi-vehicle systems," in *Proc. 2021 IEEE Conference on Control Technology and Applications*, 2021.

[83] J. Matouš, K. Y. Pettersen, and C. Paliotta, "Formation path following control of underactuated AUVs," in *Proc. 2022 European Control Conference*, 2022.

[79] J. Matouš, K. Y. Pettersen, D. Varagnolo, and C. Paliotta, "Singularity-free formation path following of underactuated AUVs," in *Proc. 2023 IFAC World Congress*, 2023.

[84] J. Matouš, D. Varagnolo, K. Y. Pettersen, and C. Paliotta, "Distributed mpc for formation path-following of multi-vehicle systems," in *Proc. 9th IFAC Conference on Networked Systems*, Jul. 2022.

[77] E. Restrepo, J. Matouš, and K. Y. Pettersen, "Tracking-in-formation of multiple autonomous marine vehicles under proximity and collision-avoidance constraints," in *Proc. 2022 European Control Conference*, Jul. 2022.

[85] E. S. Lie, J. Matouš, and K. Y. Pettersen, "Formation control of underactuated AUVs using the hand position concept," in *To appear in Proc. 2023 IEEE Conference on Decision and Control*, 2023.

Journal Submissions

[86] J. Matouš, K. Y. Pettersen, D. Varagnolo, and C. Paliotta, "A distributed NSB algorithm for formation path following," *Submitted to IEEE Transactions on Control Systems Technology*, 2023.

[87] J. Matouš, C. Paliotta, K. Y. Pettersen, and D. Varagnolo, "Trajectory tracking and path following of underactuated AUVs using the hand position concept," *Submitted to IEEE Transactions on Control Systems Technology*, 2023.

[88] E. Restrepo, J. Matouš, and K. Y. Pettersen, "Tracking control of cooperative marine vehicles under hard and soft constraints," *Submitted to IEEE Transactions on Control of Network Systems*, 2023.

Chapter 2

Background

The latter consisted simply of six hydrocoptic marzlevanes, so fitted to the ambifacient lunar waneshaft that side fumbling was effectively prevented.

— John Hellins Quick, "The turbo-encabulator in industry," Students' Quarterly Journal, 1944.

This chapter presents some of the background theory that is used in the thesis. The theory presented here is relevant to multiple chapters throughout the thesis. In Section 2.1, we present a control-oriented model of marine vehicles. Section 2.2 presents the theory behind paths, that is then used in Section 2.3 to define the formation path-following problem. In Section 2.4, we then present the line-of-sight guidance algorithm as a method for solving the path-following problem. Section 2.5 presents the NSB algorithm as a method for solving the combined formation path-following problem. Finally, Section 2.6 presents the concept of uniform semiglobal exponential stability.

2.1 Mathematical Models of Marine Vehicles

Mathematical models are vital to the design, analysis, and verification of control algorithms. For complex systems, such as marine vehicles, there typically exist different types of models. These models often represent a trade-off between simplicity and fidelity. On one end of the spectrum, there are high-fidelity models. These models are an accurate representation of the system. Consequently, high-fidelity models are used in ship-handling simulators, as well as in some simulation environments, *e.g.,* the Unified Navigation Environment (DUNE) [89], to train the operators and to verify the effectiveness of control algorithms. However, due to their complexity, these models are not suitable for designing or analyzing control algorithms.

On the other end of the spectrum, there are control-oriented models. Compared to high-fidelity models, the structure of control-oriented models is simpler. This simplicity means that we can use these models to design control algorithms and

analyze their closed-loop properties. The control-oriented models are thus designed to capture the system properties that are the most significant and relevant for the design of the control system. The inaccuracies of control-oriented models can be compensated by designing robust controllers.

This section presents control-oriented models of autonomous surface vehicles (ASVs) and autonomous underwater vehicles (AUVs). The presented models are similar to those used in [72] and [90]. The models are based on the matrix-vector models of marine vehicles [91] and some simplifying assumptions.

2.1.1 State Variables and Degrees of Freedom

Marine vehicles are typically modeled as rigid bodies. A rigid body moving in three-dimensional space has six DOFs, three for position and three for orientation.

The position of a marine vehicle is commonly expressed in a local north-east-down (NED) coordinate frame. Although the NED frame is not inertial, it is often used as an approximation of an inertial coordinate frame for short-term and short-distance missions, since the effect of Earth's rotation on the vehicles is negligible. In general, we will denote the position of the vehicles as $\mathbf{p} = [x, y, z]^{\mathrm{T}}$.

The orientation of a vehicle can be expressed using *Euler angles*, $\boldsymbol{\Theta} = [\phi, \theta, \psi]^{\mathrm{T}}$, where ϕ is the roll angle, θ is the pitch angle, and ψ is the yaw angle. The complete position and orientation vector of the vehicle is then given by $\boldsymbol{\eta}^{\mathrm{T}} = \left[\mathbf{p}^{\mathrm{T}}, \boldsymbol{\Theta}^{\mathrm{T}}\right]$.

Although Euler angles can represent any orientation, in some cases, this representation is not unique. For example, the following two sets of Euler angles represent the same attitude

$$\boldsymbol{\Theta}_1 = \left[\frac{\pi}{2}, \frac{\pi}{2}, 0\right]^{\mathrm{T}}, \qquad \boldsymbol{\Theta}_2 = \left[0, \frac{\pi}{2}, -\frac{\pi}{2}\right]^{\mathrm{T}}. \tag{2.1}$$

At these attitudes, there exist mathematical singularities called *gimbal locks* [92]. Furthermore, the use of Euler angles in control may lead to a phenomenon called *unwinding* [93], in which the vehicle performs an unnecessary rotation to reach the desired attitude.

The orientation of a vehicle can also be described using a *rotation matrix*. Rotation matrices are members of the special orthogonal group $SO(3)$. Unlike Euler angles, rotation matrices do not suffer from singularities. For a given set of Euler angles, the corresponding rotation matrix is given by [91]

$$\mathbf{R}(\phi, \theta, \psi) = \begin{bmatrix} \mathrm{c}_\psi \, \mathrm{c}_\theta & \mathrm{c}_\psi \, \mathrm{s}_\phi \, \mathrm{s}_\theta - \mathrm{c}_\phi \, \mathrm{s}_\psi & \mathrm{s}_\phi \, \mathrm{s}_\psi + \mathrm{c}_\phi \, \mathrm{c}_\psi \, \mathrm{s}_\theta \\ \mathrm{c}_\theta \, \mathrm{s}_\psi & \mathrm{c}_\phi \, \mathrm{c}_\psi + \mathrm{s}_\phi \, \mathrm{s}_\psi \, \mathrm{s}_\theta & \mathrm{c}_\phi \, \mathrm{s}_\psi \, \mathrm{s}_\theta - \mathrm{c}_\psi \, \mathrm{s}_\phi \\ -\mathrm{s}_\theta & \mathrm{c}_\theta \, \mathrm{s}_\phi & \mathrm{c}_\phi \, \mathrm{c}_\theta \end{bmatrix}, \tag{2.2}$$

where c_x and s_x represent the cosine and sine of the corresponding angle.

Next, let us discuss the representation of velocities. The velocities of the vehicle are expressed in the *body-fixed* frame, a non-inertial coordinate frame attached to the vehicle, with the x-axis pointing towards the bow (front) side, the y-axis pointing to the starboard (right) side, and the z-axis pointing to the bottom side of the vehicle. The *linear velocities* of the vehicle $\boldsymbol{\upsilon} = [u, v, w]^{\mathrm{T}}$ consist of the surge, sway, and heave velocities. The *angular velocities* of the vehicle $\boldsymbol{\omega} = [p, q, r]^{\mathrm{T}}$

consist of the roll, pitch, and yaw rates. The full velocity vector is then given by $\boldsymbol{\nu}^{\mathrm{T}} = \left[\boldsymbol{\upsilon}^{\mathrm{T}}, \boldsymbol{\omega}^{\mathrm{T}}\right]$.

Throughout the thesis, we denote velocities in the body-fixed frame as $\boldsymbol{\upsilon}$, while velocities in the inertial frame are denoted as $\mathbf{v}$. The difference between these two types of velocities is illustrated in Figure 2.1.

Finally, let us discuss simplified 3DOF and 5DOF models. In the case of ASVs or AUVs moving in the horizontal plane, we often assume that the roll and pitch angles are zero, and that the depth is constant. Consequently, we can disregard the roll, pitch, and heave motion of the vehicle, and derive a simplified 3DOF model with $\boldsymbol{\eta} = [x, y, \psi]^{\mathrm{T}}$ and $\boldsymbol{\nu} = [u, v, r]^{\mathrm{T}}$. In the case of slender, torpedo-shaped AUVs, the roll motion is assumed to be small and self-stabilizing by the design of the vehicle. Consequently, we can disregard the roll motion and derive a simplified 5DOF model with $\boldsymbol{\eta} = [x, y, z, \theta, \psi]^{\mathrm{T}}$ and $\boldsymbol{\nu} = [u, v, w, q, r]^{\mathrm{T}}$.

2.1.2 Kinematics

First, let us discuss the kinematics of the vehicles, starting with the 6DOF model. The time-derivative of the position is

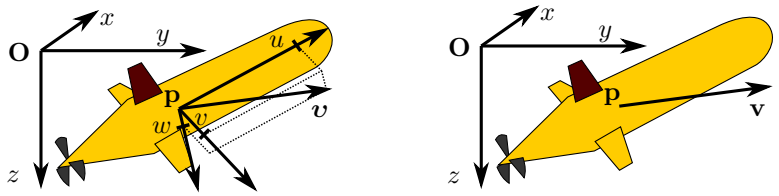
$$\dot{\mathbf{p}} = \mathbf{R}\boldsymbol{\upsilon}. \tag{2.3}$$

The time-derivative of the Euler angles is given by [91]

$$\dot{\boldsymbol{\Theta}} = \mathbf{T}(\boldsymbol{\Theta})\boldsymbol{\omega}, \qquad \mathbf{T}(\boldsymbol{\Theta}) = \begin{bmatrix} 1 & \mathrm{s}_\phi \mathrm{t}_\theta & \mathrm{c}_\phi \mathrm{t}_\theta \\ 0 & \mathrm{c}_\phi & -\mathrm{s}_\phi \\ 0 & \mathrm{s}_\phi / \mathrm{c}_\theta & \mathrm{c}_\phi / \mathrm{c}_\theta \end{bmatrix}, \tag{2.4}$$

where $\mathrm{t}_\theta = \tan(\theta)$. Due to the aforementioned singularities, $\dot{\boldsymbol{\Theta}}$ is not defined for $\theta = \pm\pi/2$. The time-derivative of a rotation matrix is given by

$$\dot{\mathbf{R}} = \mathbf{R}\mathbf{S}(\omega), \qquad \mathbf{S}(\omega) = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}. \tag{2.5}$$



(a) Illustration of body-fixed velocities. The kinematics of the vehicle are $\dot{\mathbf{p}} = \mathbf{R}\boldsymbol{\upsilon}$.

(b) Illustration of inertial velocities. The kinematics of the vehicle are $\dot{\mathbf{p}} = \mathbf{v}$.

Figure 2.1: Illustration of the difference between body-fixed and inertial velocities.

To derive the kinematics of the 5DOF model, we simply substitute $\phi = 0$ and $p = 0$ into (2.3) and (2.4), and get

$$\dot{\mathbf{p}} = \mathbf{R}(0, \theta, \psi)\boldsymbol{\upsilon}, \tag{2.6a}$$

$$\dot{\theta} = q, \tag{2.6b}$$

$$\dot{\psi} = \frac{r}{\cos(\theta)}. \tag{2.6c}$$

Similarly, we can derive the kinematics of the 3DOF model by substituting $z = \phi = \theta = w = p = q = 0$ into (2.3) and (2.4)

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\psi)\boldsymbol{\nu}, \qquad \mathbf{J}(\psi) = \begin{bmatrix} \mathrm{c}_\psi & -\mathrm{s}_\psi & 0 \\ \mathrm{s}_\psi & \mathrm{c}_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.7}$$

2.1.3 Dynamics

When modeling the dynamics of marine vehicles, we often need to consider the effect of *sea loads* such as waves, wind, and ocean currents. Let $\mathbf{V}_c \in \mathbb{R}^3$ be a vector that represents the velocity of the ocean current in the inertial coordinate frame. Since the dynamics of ocean currents are typically much slower than the dynamics of the vehicle, the ocean current can be considered constant and irrotational. Let $\boldsymbol{\upsilon}_c = \mathbf{R}^{\mathrm{T}}\mathbf{V}_c$ denote the velocity of the ocean current expressed in the vehicle's body-fixed frame. Furthermore, let $\boldsymbol{\upsilon}_r = \boldsymbol{\upsilon} - \boldsymbol{\upsilon}_c \triangleq [u_r, v_r, w_r]^{\mathrm{T}}$ denote the relative surge, sway, and heave velocity of the vehicle, and let $\boldsymbol{\nu}_r^{\mathrm{T}} = \left[\boldsymbol{\upsilon}_r^{\mathrm{T}}, \boldsymbol{\omega}^{\mathrm{T}}\right]$ denote the full relative velocity vector. The dynamics of the vehicle can then be expressed using the following matrix-vector model [91]

$$\mathbf{M}\dot{\boldsymbol{\nu}}_r + \mathbf{C}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \mathbf{D}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \mathbf{g}(\mathbf{R}) = \boldsymbol{\tau}, \tag{2.8}$$

where $\mathbf{M}$ is the mass and inertia matrix, including the added mass effects, $\mathbf{C}(\boldsymbol{\nu}_r)$ is the Coriolis and centripetal matrix, also including the added mass, $\mathbf{D}(\boldsymbol{\nu}_r)$ is the hydrodynamic damping matrix, $\mathbf{g}(\mathbf{R})$ represents the effects of gravity and buoyancy, and $\boldsymbol{\tau}$ represents additional forces and torques such as the effects of actuators and external disturbances.

The model in (2.8) can also be expressed in terms of absolute velocities

$$\mathbf{M}\left(\dot{\boldsymbol{\nu}} - \dot{\boldsymbol{\nu}}_c\right) + \mathbf{C}(\boldsymbol{\nu} - \boldsymbol{\nu}_c)(\boldsymbol{\nu} - \boldsymbol{\nu}_c) + \mathbf{D}(\boldsymbol{\nu} - \boldsymbol{\nu}_c)(\boldsymbol{\nu} - \boldsymbol{\nu}_c) + \mathbf{g}(\mathbf{R}) = \boldsymbol{\tau}, \tag{2.9}$$

where $\boldsymbol{\nu}_c^{\mathrm{T}} = \left[\boldsymbol{\upsilon}_c^{\mathrm{T}}, \mathbf{0}^{\mathrm{T}}\right]$.

The inertia matrix $\mathbf{M}$ is symmetric positive definite, the damping matrix $\mathbf{D}$ is positive definite, and the Coriolis matrix $\mathbf{C}$ is parametrized so that it is skew-symmetric. There exist multiple expressions for the Coriolis matrix, *e.g.,*

$$\mathbf{C}(\boldsymbol{\nu}_r) = \begin{bmatrix} \mathbf{O}_{3\times 3} & -\mathbf{S}(\mathbf{M}_{11}\boldsymbol{\upsilon}_r + \mathbf{M}_{12}\boldsymbol{\omega}) \\ -\mathbf{S}(\mathbf{M}_{11}\boldsymbol{\upsilon}_r + \mathbf{M}_{12}\boldsymbol{\omega}) & -\mathbf{S}(\mathbf{M}_{21}\boldsymbol{\upsilon}_r + \mathbf{M}_{22}\boldsymbol{\omega}) \end{bmatrix}, \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix} = \mathbf{M}. \tag{2.10}$$

The gravity and buoyancy vector is given by [91]

$$\mathbf{g}(\mathbf{R}) = -\begin{bmatrix} (W - B)\mathbf{R}^{\mathrm{T}}\mathbf{e}_3 \\ (W\mathbf{r}_g - B\mathbf{r}_b) \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3 \end{bmatrix}, \tag{2.11}$$

where $W \in \mathbb{R}_{>0}$ is the gravitational force, $B \in \mathbb{R}_{>0}$ is the buoyant force, $\mathbf{r}_g$ is the position of the center of gravity, $\mathbf{r}_b$ is the position of the center of buoyancy, and $\mathbf{e}_3 = [0, 0, 1]^{\mathrm{T}}$.

2.1.4 Control-Oriented Model of Underactuated AUVs

Before deriving the control-oriented model, we need to present the assumptions that allow us to simplify the matrix-vector model in (2.8).

Assumption 2.1. *The vehicle is slender, torpedo-shaped, with port-starboard symmetry.*

Assumption 2.2. *The vehicle is maneuvering at low speeds. Consequently, the hydrodynamic damping can be considered linear.*

Under Assumption 2.2, the hydrodynamic damping matrix is constant. Under Assumption 2.1, the inertia and damping matrices have the following structure

$$\mathbf{M} = \begin{bmatrix} m_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & m_{22} & 0 & 0 & 0 & m_{26} \\ 0 & 0 & m_{33} & 0 & m_{35} & 0 \\ 0 & 0 & 0 & m_{44} & 0 & 0 \\ 0 & 0 & m_{35} & 0 & m_{55} & 0 \\ 0 & m_{26} & 0 & 0 & 0 & m_{66} \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} d_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & d_{22} & 0 & 0 & 0 & d_{26} \\ 0 & 0 & d_{33} & 0 & d_{35} & 0 \\ 0 & 0 & 0 & d_{44} & 0 & 0 \\ 0 & 0 & d_{53} & 0 & d_{55} & 0 \\ 0 & d_{62} & 0 & 0 & 0 & d_{66} \end{bmatrix}. \tag{2.12}$$

Assumption 2.3. *The vehicle is equipped with a propeller and fins. Consequently, the vehicle is capable of generating a force in the surge direction and torques around all three axes.*

Under this assumption, the external forces acting on the vehicle are given by

$$\boldsymbol{\tau} = \mathbf{B}\mathbf{f}, \qquad \mathbf{B} = \begin{bmatrix} b_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & b_{24} \\ 0 & 0 & b_{33} & 0 \\ 0 & b_{42} & 0 & 0 \\ 0 & 0 & b_{53} & 0 \\ 0 & 0 & 0 & b_{64} \end{bmatrix}, \tag{2.13}$$

where $\mathbf{f} = [T_u, T_p, T_q, T_r]^{\mathrm{T}}$ is the control input consisting of the surge thrust and the forces produced by the fins.

If the vehicle model (2.8) satisfies Assumptions 2.1–2.3, then we can perform a change of coordinates such that the actuators produce no sway or heave acceleration. In other words, for all inputs $\mathbf{f}$, there exist $f_u, t_p, t_q, t_r \in \mathbb{R}$ such that

$$\mathbf{M}^{-1}\mathbf{B}\mathbf{f} = [f_u, 0, 0, t_p, t_q, t_r]^{\mathrm{T}}. \tag{2.14}$$

This change of coordinates was demonstrated for 5DOF vehicles in [72], and can be trivially extended to 6DOFs. The transformed body-fixed velocities, $\boldsymbol{\nu}_r'$, are

$$\boldsymbol{\nu}_r' = [u_r, v_r + \varepsilon_1 r, w_r + \varepsilon_2 q, p, q, r]^{\mathrm{T}}, \tag{2.15}$$

where $\varepsilon_1, \varepsilon_2 \in \mathbb{R}$. This transformation can also be written as

$$\boldsymbol{\nu}_r = \mathbf{H}\boldsymbol{\nu}_r', \qquad \mathbf{H} = \begin{bmatrix} & 0 & 0 & 0 \\ \mathbf{I}_3 & 0 & 0 & -\varepsilon_1 \\ & 0 & -\varepsilon_2 & 0 \\ \mathbf{O}_{3\times 3} & & \mathbf{I}_3 & \end{bmatrix}. \tag{2.16}$$

The transformed dynamics are then given by

$$\mathbf{M}'\dot{\boldsymbol{\nu}}_r' + \mathbf{C}'(\boldsymbol{\nu}_r')\boldsymbol{\nu}_r' + \mathbf{D}'\boldsymbol{\nu}_r' + \mathbf{H}^{\mathrm{T}}\mathbf{g}(\mathbf{R}) = \mathbf{H}^{\mathrm{T}}\boldsymbol{\tau}, \tag{2.17}$$

where $\mathbf{M}' = \mathbf{H}^{\mathrm{T}}\mathbf{M}\mathbf{H}$, $\mathbf{D}' = \mathbf{H}^{\mathrm{T}}\mathbf{D}\mathbf{H}$ are the transformed inertia and damping matrices, and $\mathbf{C}'(\boldsymbol{\nu}_r') = \mathbf{H}^{\mathrm{T}}\mathbf{C}(\boldsymbol{\nu}_r)\mathbf{H}$.

If we choose

$$\varepsilon_1 = \frac{b_{24}\, m_{66} - b_{64}\, m_{26}}{b_{24}\, m_{26} - b_{64}\, m_{22}}, \qquad \varepsilon_2 = \frac{b_{33}\, m_{55} - b_{53}\, m_{35}}{b_{33}\, m_{35} - b_{53}\, m_{33}}, \tag{2.18}$$

the effect of the actuators on the dynamics of $\boldsymbol{\nu}'$ is given by

$$\mathbf{M}'^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{B}\mathbf{f} = \begin{bmatrix} \frac{b_{11}}{m_{11}} T_u \\ 0 \\ 0 \\ \frac{b_{42}}{m_{44}} T_p \\ \frac{(b_{53}\, m_{33} - b_{33}\, m_{35})}{m_{33}\, m_{55} - {m_{35}}^2} T_q \\ \frac{(b_{64}\, m_{22} - b_{24}\, m_{26})}{m_{22}\, m_{66} - {m_{26}}^2} T_r \end{bmatrix}. \tag{2.19}$$

We have thus shown that (2.14) is satisfied with

$$t_u = \frac{b_{11}}{m_{11}} T_u, \qquad t_p = \frac{b_{42}}{m_{44}} T_p, \tag{2.20a}$$

$$t_q = \frac{(b_{53}\, m_{33} - b_{33}\, m_{35})}{m_{33}\, m_{55} - {m_{35}}^2} T_q, \qquad t_r = \frac{(b_{64}\, m_{22} - b_{24}\, m_{26})}{m_{22}\, m_{66} - {m_{26}}^2} T_r. \tag{2.20b}$$

Moreover, if all the numerators in (2.20) are nonzero, then the converse holds as well, *i.e.,* for all t_u, t_p, t_q, t_r, there exists an input $\mathbf{f}$ such that (2.14) holds. Consequently, we can treat t_u, t_p, t_q, t_r as the new inputs to the system.

Remark. *If the vehicle is rotationally symmetric around the x-axis (*i.e., *if $m_{22} = m_{33}$, $m_{35} = -m_{26}$, $m_{55} = m_{66}$, $b_{33} = -b_{24}$, and $b_{53} = b_{64}$), then we have $\varepsilon_1 = -\varepsilon_2$, and the transformation* (2.16) *corresponds to moving the body-fixed coordinate frame a distance ε_1 along the body-fixed x-axis.*

Assumption 2.4. *The vehicle is neutrally buoyant, with the centers of gravity and buoyancy located on one vertical axis.*

Under this assumption, $\mathbf{g}(\mathbf{R})$ has the following shape

$$\mathbf{g}(\mathbf{R}) = \begin{bmatrix} \mathbf{0}_3 \\ W z_{gb}\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3 \end{bmatrix}, \tag{2.21}$$

where z_{gb} is the distance between the centers of gravity and buoyancy.

2.1.5 The Component Form

In this section, we express the matrix-vector model (2.8) in a *component form* by writing out the equations of motion for the individual state variables. The purpose of this model is to gain a better understanding of the AUV dynamics, which allows us to design low-level controllers and analyze their closed-loop behavior.

Consider an AUV model that satisfies Assumptions 2.1–2.4. Then, the model in the component form is given by

$$\dot{u}_r = F_u(\boldsymbol{\nu}_r) + f_u, \tag{2.22a}$$
$$\dot{v}_r = X_v(u_r)r + Y_v(u_r)v_r + Z_v(p)w_r, \tag{2.22b}$$
$$\dot{w}_r = X_w(u_r)q + Y_w(u_r)w_r + Z_w(p)v_r + G(\mathbf{R}), \tag{2.22c}$$
$$\dot{p} = F_p(\boldsymbol{\nu}_r) + t_p, \tag{2.22d}$$
$$\dot{q} = F_q(\boldsymbol{\nu}_r) + t_q, \tag{2.22e}$$
$$\dot{r} = F_r(\boldsymbol{\nu}_r) + t_r. \tag{2.22f}$$

The expressions for F_u, X_v, Y_v, Z_v, X_w, Y_w, Z_w, G, F_p, F_q, and F_r are shown in Appendix A.1. Note that $\dot{u}_r$, $\dot{p}$, $\dot{q}$, and $\dot{r}$ depend on the control inputs. Consequently, it is possible to stabilize these states using feedback control. The states v and w are commonly referred to as the *underactuated* dynamics of the vehicle, as these states cannot be controlled directly. The terms Y_v and Y_w represent the effects of hydrodynamic damping. Because hydrodynamic damping is dissipative, the terms Y_v and Y_w are negative. The term G represents the effects of gravity and buoyancy on the heave velocity. The remaining terms represent the Coriolis and centripetal forces.

In the remainder of this section, we derive a component form for the absolute velocities. From (2.22), we have

$$\dot{u} = F_u(\boldsymbol{\nu} - \boldsymbol{\nu}_c) + \dot{u}_c + f_u, \tag{2.23a}$$
$$\dot{v} = X_v(u - u_c)r + Y_v(u - u_c)(v - v_c) + Z_v(p)(w - w_c) + \dot{v}_c, \tag{2.23b}$$
$$\dot{w} = X_w(u - u_c)q + Y_w(u - u_c)(w - w_c) + Z_w(p)(v - v_c) + G(\mathbf{R}) + \dot{w}_c, \tag{2.23c}$$
$$\dot{p} = F_p(\boldsymbol{\nu} - \boldsymbol{\nu}_c) + t_p, \tag{2.23d}$$
$$\dot{q} = F_q(\boldsymbol{\nu} - \boldsymbol{\nu}_c) + t_q, \tag{2.23e}$$
$$\dot{r} = F_r(\boldsymbol{\nu} - \boldsymbol{\nu}_c) + t_r. \tag{2.23f}$$

From the expressions in Appendix A.1, we conclude that all terms in (2.22) that contain the relative velocities are either linear or quadratic. Since the relative velocities are affine in the ocean current ($\boldsymbol{\upsilon}_r = \boldsymbol{\upsilon} - \mathbf{R}^{\mathrm{T}}\mathbf{V}_c$), we can conclude that all terms in (2.23) that contain the ocean current are also linear or quadratic. Consequently, if we denote the components of the ocean current velocity as $\mathbf{V}_c =$

$[V_x, V_y, V_z]^{\mathrm{T}}$, the model (2.23) can be written as

$$\dot{u} = F_u(\boldsymbol{\nu}) + f_u + \Phi_u(\boldsymbol{\nu}, \mathbf{R})^{\mathrm{T}} \mathbb{V}_c, \tag{2.24a}$$

$$\dot{v} = X_v(u)r + Y_v(u)v + Z_v(p)w + \Phi_v(\boldsymbol{\nu}, \mathbf{R})^{\mathrm{T}} \mathbb{V}_c, \tag{2.24b}$$

$$\dot{w} = X_w(u)q + Y_w(u)w + Z_w(p)v + G(\mathbf{R}) + \Phi_w(\boldsymbol{\nu}, \mathbf{R})^{\mathrm{T}} \mathbb{V}_c, \tag{2.24c}$$

$$\dot{p} = F_p(\boldsymbol{\nu}, \mathbf{R}) + t_p + \Phi_p(\boldsymbol{\nu}, \mathbf{R})^{\mathrm{T}} \mathbb{V}_c, \tag{2.24d}$$

$$\dot{q} = F_q(\boldsymbol{\nu}, \mathbf{R}) + t_q + \Phi_q(\boldsymbol{\nu}, \mathbf{R})^{\mathrm{T}} \mathbb{V}_c, \tag{2.24e}$$

$$\dot{r} = F_r(\boldsymbol{\nu}) + t_r + \Phi_r(\boldsymbol{\nu}, \mathbf{R})^{\mathrm{T}} \mathbb{V}_c, \tag{2.24f}$$

where $\mathbb{V}_c = \left[V_x, V_y, V_z, V_x^2, V_y^2, V_z^2, V_xV_y, V_xV_z, V_yV_z\right]^{\mathrm{T}}$. The expressions for Φ_u, Φ_v, Φ_w, Φ_p, Φ_q, and Φ_r are omitted in this thesis. Instead, let us present a method for finding them.

Let $\mathbf{r}_1$, $\mathbf{r}_2$, and $\mathbf{r}_3$ denote the columns of the rotation matrix $\mathbf{R}$. The ocean current velocities in the body-fixed frame are given by

$$\boldsymbol{\upsilon}_c = \mathbf{R}^{\mathrm{T}} \mathbf{V}_c \quad \Longrightarrow \quad u_c = \mathbf{r}_1^{\mathrm{T}} \mathbf{V}_c,\ v_c = \mathbf{r}_2^{\mathrm{T}} \mathbf{V}_c,\ w_c = \mathbf{r}_3^{\mathrm{T}} \mathbf{V}_c. \tag{2.25}$$

Suppose then that the right-hand side of (2.22) contains a linear term, *e.g.* ku_r, where $k \in \mathbb{R}$. This term can be expressed as

$$ku_r = k(u - u_c) = ku - k\mathbf{r}_1^{\mathrm{T}} \mathbf{V}_c = ku + k\varphi_u^{\mathrm{T}} \mathbb{V}_c, \tag{2.26}$$

where $\varphi_u^{\mathrm{T}} = \left[-\mathbf{r}_1^{\mathrm{T}}, \mathbf{0}_6^{\mathrm{T}}\right]$. We can derive similar equations for linear terms containing v_r and w_r.

Next, consider a quadratic term, *e.g.,* ku_rv_r, where $k \in \mathbb{R}$. This term can be expressed as

$$ku_rv_r = k(uv - uv_c - u_cv + u_cv_c) = kuv + k\varphi_{uv}(u, v)^{\mathrm{T}} \mathbb{V}_c, \tag{2.27}$$

where

$$\varphi_{uv}(u, v)^{\mathrm{T}} = \big[- u\mathbf{r}_2^{\mathrm{T}} - v\mathbf{r}_1^{\mathrm{T}}, r_{11}r_{21}, r_{12}r_{22}, r_{13}r_{23}, r_{11}r_{22} + r_{12}r_{21}, \\ r_{11}r_{23} + r_{13}r_{21}, r_{12}r_{23} + r_{13}r_{22}\big], \tag{2.28}$$

where

$$\mathbf{r}_1 = [r_{11}, r_{12}, r_{13}]^{\mathrm{T}}, \qquad \mathbf{r}_2 = [r_{21}, r_{22}, r_{23}]^{\mathrm{T}}. \tag{2.29}$$

We can derive similar equations for all the other quadratic terms.

Finally, let us investigate the derivatives of the ocean current velocities. The derivative of $\boldsymbol{\upsilon}_c$ is

$$\dot{\boldsymbol{\upsilon}}_c = \left(\mathbf{R}\mathbf{S}(\boldsymbol{\omega})\right)^{\mathrm{T}} \mathbf{V}_c \triangleq \left[\dot{\mathbf{r}}_1, \dot{\mathbf{r}}_2, \dot{\mathbf{r}}_3\right]^{\mathrm{T}} \mathbf{V}_c, \tag{2.30}$$

where $\dot{\mathbf{r}}_1$, $\dot{\mathbf{r}}_2$, and $\dot{\mathbf{r}}_3$ denote the columns of $\mathbf{R}\mathbf{S}(\boldsymbol{\omega})$. The components of $\dot{\boldsymbol{\upsilon}}_c$ are thus given by

$$\dot{u}_c = \dot{\mathbf{r}}_1^{\mathrm{T}} \mathbf{V}_c, \qquad \dot{v}_c = \dot{\mathbf{r}}_2^{\mathrm{T}} \mathbf{V}_c, \qquad \dot{w}_c = \dot{\mathbf{r}}_3^{\mathrm{T}} \mathbf{V}_c. \tag{2.31}$$

We have thus shown that the AUV dynamics can be split into an ocean current-independent and an ocean current-dependent part.

In the remainder of this section, we derive a component form for the 5DOF and 3DOF models.

5DOF Component Form

In the 5DOF model, the roll dynamics are disregarded. Consequently, the inertia, damping, and Coriolis matrices of the 5DOF model are obtained by removing the fourth row and fourth column from the inertia, damping, and Coriolis matrices of the 6DOF model from Section 2.1.4.

The assumptions for deriving the 5DOF control-oriented model are analogous to the assumptions in Section 2.1.4, with one exception. Due to fewer degrees of freedom, Assumption 2.3 must be modified.

Assumption 2.3 (5DOF). *The vehicle is equipped with a propeller and fins. Consequently, the vehicle is capable of generating a force in the surge direction and torques in pitch and yaw.*

Under this assumption, the external forces acting on the vehicle are

$$\boldsymbol{\tau} = \mathbf{B}\mathbf{f}, \qquad \mathbf{B} = \begin{bmatrix} b_{11} & 0 & 0 \\ 0 & 0 & b_{23} \\ 0 & b_{32} & 0 \\ 0 & b_{42} & 0 \\ 0 & 0 & b_{53} \end{bmatrix}, \tag{2.32}$$

where $\mathbf{f} = [T_u, T_q, T_r]^{\mathrm{T}}$ is the control input.

Similarly to Section 2.1.4, we can perform a change of coordinates so that the actuators produce no sway or heave acceleration. In other words, for all inputs $\mathbf{f}$, there exist $f_u, t_q, t_r \in \mathbb{R}$ such that

$$\mathbf{M}^{-1}\mathbf{B}\mathbf{f} = [f_u, 0, 0, t_q, t_r]^{\mathrm{T}}. \tag{2.33}$$

The component form of the 5DOF model is then simply obtained by substituting $p = 0$ into (2.24)

$$\dot{u} = F_u(\boldsymbol{\nu}) + f_u + \Phi_u(\boldsymbol{\nu}, \theta, \psi)^{\mathrm{T}}\mathbb{V}_c, \tag{2.34a}$$
$$\dot{v} = X_v(u)r + Y_v(u)v + \Phi_v(\boldsymbol{\nu}, \theta, \psi)^{\mathrm{T}}\mathbb{V}_c, \tag{2.34b}$$
$$\dot{w} = X_w(u)q + Y_w(u)w + G(\theta) + \Phi_w(\boldsymbol{\nu}, \theta, \psi)^{\mathrm{T}}\mathbb{V}_c, \tag{2.34c}$$
$$\dot{q} = F_q(\boldsymbol{\nu}) + t_q + \Phi_q(\boldsymbol{\nu}, \theta, \psi)^{\mathrm{T}}\mathbb{V}_c, \tag{2.34d}$$
$$\dot{r} = F_r(\boldsymbol{\nu}) + t_r + \Phi_r(\boldsymbol{\nu}, \theta, \psi)^{\mathrm{T}}\mathbb{V}_c. \tag{2.34e}$$

3DOF Component Form

First, let us discuss the 3DOF control-oriented model. Similarly to the previous section, the model is derived using assumptions that are analogous to the ones in Section 2.1.4, with some modifications. Namely, due to fewer degrees of freedom, Assumption 2.3 needs to be modified.

Assumption 2.3 (3DOF). *The vehicle is equipped with a propeller and a rudder. Consequently, the vehicle is capable of generating a force in the surge direction and a torque in yaw.*

Under this assumption, the external forces acting on the vehicle are

$$\boldsymbol{\tau} = \mathbf{B}\mathbf{f}, \qquad \mathbf{B} = \begin{bmatrix} b_{11} & 0 \\ 0 & b_{22} \\ 0 & b_{32} \end{bmatrix}, \tag{2.35}$$

where $\mathbf{f} = [T_u, T_r]^{\mathrm{T}}$ is the control input.

The inertia, damping, and Coriolis matrices of the 3DOF model are [90]

$$\mathbf{M} = \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & m_{22} & m_{23} \\ 0 & m_{23} & m_{33} \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} d_{11} & 0 & 0 \\ 0 & d_{22} & d_{23} \\ 0 & d_{32} & d_{33} \end{bmatrix}, \tag{2.36}$$

$$\mathbf{C}(\boldsymbol{\nu}_r) = \begin{bmatrix} 0 & 0 & -m_{22}v_r - m_{23}r \\ 0 & 0 & m_{11}u_r \\ m_{22}v_r + m_{23}r & -m_{11}u_r & 0 \end{bmatrix}. \tag{2.37}$$

In [90], it is shown that the origin of the body-fixed coordinate frame can be chosen such that the actuators produce no sway acceleration. In other words, for all inputs $\mathbf{f}$, there exist $f_u, t_r \in \mathbb{R}$ such that

$$\mathbf{M}^{-1}\mathbf{B}\mathbf{f} = [f_u, 0, t_r]^{\mathrm{T}}. \tag{2.38}$$

The change of coordinates is done by translating the origin of the body-fixed frame by ε along the x-axis. The transformed velocities are given by

$$\boldsymbol{\nu}_r' = \mathbf{H}\boldsymbol{\nu}_r, \qquad \mathbf{H} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \varepsilon \\ 0 & 0 & 1 \end{bmatrix}. \tag{2.39}$$

Similarly to the procedure in Section 2.1.4, we define the transformed inertia matrix $\mathbf{M}' = \mathbf{H}^{\mathrm{T}}\mathbf{M}\mathbf{H}$. If we choose

$$\varepsilon = -\frac{b_{22}\, m_{33} - b_{32}\, m_{23}}{b_{22}\, m_{23} - b_{32}\, m_{22}}, \tag{2.40}$$

then the effect of actuators in the new coordinate frame is given by

$$\mathbf{M}'^{-1}\mathbf{H}^{\mathrm{T}}\mathbf{B}\mathbf{f} = \begin{bmatrix} \frac{b_{11}}{m_{11}} T_u \\ 0 \\ \frac{b_{32}\, m_{22} - b_{22}\, m_{23}}{m_{22}\, m_{33} - {m_{23}}^2} T_r \end{bmatrix}. \tag{2.41}$$

We can then perform a change of inputs

$$f_u = \frac{b_{11}}{m_{11}} T_u, \qquad t_r = \frac{b_{32}\, m_{22} - b_{22}\, m_{23}}{m_{22}\, m_{33} - {m_{23}}^2} T_r, \tag{2.42}$$

and express the 3DOF model in the following component form

$$\dot{u}_r = F_u(\boldsymbol{\nu}_r) + f_u, \tag{2.43a}$$

$$\dot{v}_r = X_v(u_r)r + Y_v(u_r)v_r, \tag{2.43b}$$

$$\dot{r} = F_r(\boldsymbol{\nu}_r) + t_r. \tag{2.43c}$$

The expressions for F_u, X_v, Y_v, and F_r are shown in Appendix A.1.

2.2 Geometric Paths

This section presents the definitions of paths in the context of guidance. The theory presented in this section applies to two- and three-dimensional Euclidean spaces. Let $n_d \in \{2, 3\}$ denote the number of dimensions.

2.2.1 Paths and Their Parametrizations

A *path* is a curve in n_d-dimensional space. A path can be expressed as a subset of Euclidean space $P \subset \mathbb{R}^{n_d}$. A *parametrization* of a path is a function $\mathbf{p}_p : \mathbb{R} \mapsto \mathbb{R}^{n_d}$ whose image space represents the given path, *i.e.,* $\{\mathbf{p}_p(s) | s \in \mathbb{R}\} = P$. Note that for a given path, there exist infinitely many parametrizations. For example, the following two functions

$$\mathbf{p}_{p,1}(s) = [s, 0, 0]^{\mathrm{T}}, \qquad \mathbf{p}_{p,1}(s) = \left[s^3, 0, 0\right]^{\mathrm{T}}, \tag{2.44}$$

represent the same path; a straight line going through the origin, parallel to the x-axis. Furthermore, if we multiply these parametrizations by a positive scalar, we also get a valid parametrization. In general, if $\mathbf{p}_p(s)$ is a path parametrization that is defined for all $s \in \mathbb{R}$, and $\rho(s)$ is a monotonically increasing function that is also defined for all $s \in \mathbb{R}$, then $\mathbf{p}_p(s)$ and $\mathbf{p}_p(\rho(s))$ parametrize the same path. We will refer to $\mathbf{p}_p(\rho(s))$ as a *reparametrization* of $\mathbf{p}_p(s)$.

2.2.2 Continuity and Regularity

Continuity of paths, also referred to as smoothness, is an important property, as some vehicles are not able to follow a path that has discontinuities or sharp turns. There are two types of continuity; *parametric* and *geometric*. Parametric continuity is related to a specific parametrization of a path, while geometric continuity is related to the curve itself. Here, we will only present the definition of parametric continuity, as this will be used further in the thesis. For details on geometric continuity, the reader is referred to [94]. Parametric continuity is denoted $\mathcal{C}^n$, where $n \in \mathbb{Z}_{\geq 0}$ is the order. A parametrization $\mathbf{p}_p(s)$ is $\mathcal{C}^n$ if it is n-times continuously differentiable.

A parametrization is *regular* if

$$\left\| \frac{\partial \mathbf{p}_p}{\partial s} \right\| \neq 0. \tag{2.45}$$

A regular parametrization means that there are no "stops" along the path. Recalling the two examples in (2.44), both $\mathbf{p}_{p,1}$ and $\mathbf{p}_{p,2}$ are $\mathcal{C}^\infty$, but only $\mathbf{p}_{p,1}$ is regular since the derivative of $\mathbf{p}_{p,2}(s)$ at $s = 0$ is zero. Regularity is an important property when defining the path-tangential vector and the path-tangential coordinate frame, as we discuss next.

2.2.3 Path-Tangential Vector and Coordinate Frame

If a parametrization is $\mathcal{C}^1$ and regular, then the *path-tangential vector* is simply the first partial derivative of $\mathbf{p}_p(s)$ with respect to s.

A *path-tangential coordinate frame* has its origin at $\mathbf{p}_p(s)$, and is oriented such that the path-tangential vector is aligned with its x-axis.

In the case of two-dimensional paths, this frame is uniquely defined. Let $\mathbf{R}_p(s) \in SO(2)$ be the rotation matrix between the path-tangential and the inertial frame. This matrix is given by

$$\mathbf{R}_p(s) = \begin{bmatrix} \cos(\psi_p(s)) & -\sin(\psi_p(s)) \\ \sin(\psi_p(s)) & \cos(\psi_p(s)) \end{bmatrix}, \ \psi_p(s) = \arctan_2\left(\frac{\partial y_p(s)}{\partial s}, \frac{\partial x_p(s)}{\partial s}\right), \tag{2.46}$$

where $x_p(s)$ and $y_p(s)$ are the components of $\mathbf{p}_p(s)$.

In the case of three-dimensional paths, the path-tangential frame is not unique. To make the x-axis of the coordinate frame aligned with the path-tangential vector, the rotation matrix $\mathbf{R}_p(s) \in SO(3)$ must satisfy

$$\mathbf{R}_p(s)\,[1,0,0]^{\mathrm{T}} = \left\| \frac{\partial \mathbf{p}_p(s)}{\partial s} \right\|^{-1} \frac{\partial \mathbf{p}_p(s)}{\partial s}. \tag{2.47}$$

There exists a subspace of rotation matrices $\mathbf{R}_p(s)$ that satisfy (2.47). For the purpose of this thesis, the choice of $\mathbf{R}_p(s)$ is not important as long as it is smooth (*i.e.,* the partial derivative of $\mathbf{R}_p(s)$ with respect to s exists and is continuous).

One potential method for choosing $\mathbf{R}_p(s)$ is to use Euler angles and enforce a zero roll angle. The rotation matrix is then given by

$$\mathbf{R}_p(s) = \begin{bmatrix} \cos(\psi_p(s))\cos(\theta_p(s)) & -\sin(\psi_p(s)) & \cos(\psi_p(s))\sin(\theta_p(s)) \\ \cos(\theta_p(s))\sin(\psi_p(s)) & \cos(\psi_p(s)) & \sin(\psi_p(s))\sin(\theta_p(s)) \\ -\sin(\theta_p(s)) & 0 & \cos(\theta_p(s)) \end{bmatrix}, \tag{2.48}$$

where

$$\theta_p(s) = -\arcsin\left(\frac{\partial z_p(s)/\partial s}{\|\partial \mathbf{p}_p(s)/\partial s\|}\right), \quad \psi_p(s) = \arctan_2\left(\frac{\partial y_p(s)}{\partial s}, \frac{\partial x_p(s)}{\partial s}\right). \tag{2.49}$$

An illustration is shown in Figure 2.2. Note that the yaw angle $\psi_p(s)$ is not defined if the desired path is vertical (*i.e.,* if $\theta_p(s) = \pm\frac{\pi}{2}$). However, we also note that most commercial AUVs can only reach a limited range of pitch angles, making them unable to move vertically. Consequently, we should avoid designing vertical paths, where the singularities of Euler angles become an issue.

2.2.4 Curvature

As previously mentioned, some vehicles are unable to follow paths with "sharp turns". For the purpose of this thesis, we define *curvature* as a measure of change of the path-tangential coordinate frame.

In the two-dimensional case, the curvature, $\kappa(s)$, is defined as

$$\kappa(s) = \frac{\partial \psi_p(s)}{\partial s}. \tag{2.50}$$

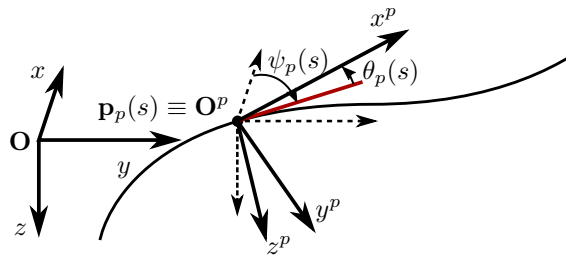


Figure 2.2: Illustration of the path-tangential coordinate frame. $\mathbf{O}$ denotes the origin of the inertial coordinate frame, $\mathbf{O}^p$ denotes the origin of the path-tangential coordinate frame. The red line represents the projection of the path-tangential vector onto the xy-plane of the inertial coordinate frame.

In the three-dimensional case, the curvature is not defined as a scalar, but rather as a vector $\boldsymbol{\omega}_p(s) \in \mathbb{R}^3$ such that

$$\frac{\partial \mathbf{R}_p(s)}{\partial s} = \mathbf{R}_p(s)\mathbf{S}\left(\boldsymbol{\omega}_p(s)\right). \tag{2.51}$$

If the rotation matrix $\mathbf{R}_p(s)$ was chosen according to (2.48), then we can also define curvature in pitch and yaw, $\kappa(s)$ and $\iota(s)$, as

$$\kappa(s) = \frac{\partial \theta_p(s)}{\partial s}, \qquad \iota(s) = \frac{\partial \psi_p(s)}{\partial s}. \tag{2.52}$$

The vector $\boldsymbol{\omega}_p(s)$ is then given by

$$\boldsymbol{\omega}_p(s) = \left[-\iota(s)\sin\left(\theta_p(s)\right), \kappa(s), \iota(s)\cos\left(\theta_p(s)\right)\right]^{\mathrm{T}}. \tag{2.53}$$

2.2.5 Parametrization by Arc Length

A path parametrization $\mathbf{p}_p(s)$ is said to be a *parametrization by arc length* if, for all $s_1, s_2 \in \mathbb{R}$, we have

$$\int_{s_1}^{s_2} \left\| \frac{\partial \mathbf{p}_p(s)}{\partial s} \right\| \mathrm{d}s = s_2 - s_1. \tag{2.54}$$

This condition is equivalent to

$$\left\| \frac{\partial \mathbf{p}_p(s)}{\partial s} \right\| = 1. \tag{2.55}$$

A convenient property of parametrizations by arc length is that the path parameter s can be interpreted as distance. Consequently, parametrizations by arc length are useful when we want the vehicles to follow the path at a constant speed. For example, choosing the corresponding path parameter $s(t)$ such that $\dot{s}(t) = 1$ means that the vehicles should follow the path at a constant speed of 1 meter per second.

Now, let us discuss how to find a parametrization by arc length. Let $\mathbf{p}_p(s)$ be an arbitrary parametrization that is $\mathcal{C}^1$ and regular. Then, we can find a parametrization by arc length by reparametrizing $\mathbf{p}_p(s)$, *i.e.,* by finding a monotonically increasing function $\rho(s) : \mathbb{R} \mapsto \mathbb{R}$ such that

$$\left\| \frac{\partial \mathbf{p}_p(\rho(s))}{\partial s} \right\| = 1. \tag{2.56}$$

The function $\rho(s)$ can be found by solving the following differential equation

$$\frac{\partial \rho(s)}{\partial s} = \left\| \frac{\partial \mathbf{p}_p(\rho)}{\partial \rho} \right\|^{-1}, \qquad \rho(0) = \rho_0, \tag{2.57}$$

where $\rho_0 \in \mathbb{R}$ is the initial condition. Although the initial condition is arbitrary, it is convenient to choose $\rho_0 = 0$ so that the new parametrization starts at the same point.

2.3 Formation Path Following

This section formally defines the formation path-following problem. Throughout the section, we consider a fleet of N vehicles. Let $\mathbf{p}_1, \ldots, \mathbf{p}_N$ denote the positions of the vehicles.

2.3.1 The Path-Following Problem

To formulate the path-following problem, we first need to define the *barycenter* of the fleet. The barycenter, $\mathbf{p}_b$, is given by the mean position of the vehicles, *i.e.,*

$$\mathbf{p}_b = \frac{1}{N} \sum_{i=1}^{N} \mathbf{p}_i. \tag{2.58}$$

To solve the path-following problem, we need to control the vehicles such that the barycenter coincides with the desired path. Let $\mathbf{p}_p(s)$ be the parametrization of the desired path. Then, the goal of path following is to control the vehicles such that

$$\mathbf{p}_b \to \mathbf{p}_p(s). \tag{2.59}$$

Let $\mathbf{p}_b^p$ denote the position of the barycenter in the path-tangential coordinate frame (see Section 2.2.3).

$$\mathbf{p}_b^p = \mathbf{R}_p^{\mathrm{T}} \left(\mathbf{p}_b - \mathbf{p}_p(s) \right). \tag{2.60}$$

Note that $\mathbf{p}_b^p$ can be interpreted as the path-following error. Indeed, $\mathbf{p}_b$ is equal to $\mathbf{p}_p(s)$ if and only if $\mathbf{p}_b^p$ is zero.

Furthermore, there is a geometric interpretation of the components of $\mathbf{p}_b^p$. Let us define $[x_b^p, y_b^p, z_b^p]^{\mathrm{T}} = \mathbf{p}_b^p$. The component x_b^p is commonly referred to as the *along-track error*, since the value of x_b^p indicates whether the barycenter is "in front of" or "behind" the desired path. The components y_b^p and z_b^p are referred to as the *cross-track errors*, since they indicate the lateral deviation from the desired path.

Path-Following versus Trajectory-Tracking

In the remainder of this section, we discuss the difference between the trajectory-tracking and the path-following problem.

The goal of trajectory tracking is to control the vehicles such that the barycenter follows a given trajectory $\mathbf{p}_d(t)$. Note that the desired trajectory is a function of time. Consequently, in trajectory tracking, the desired position of the barycenter for a given time t is fixed. In path following, the desired position of the barycenter depends on the path parameter s. The path parameter can thus be treated as an additional degree of freedom when designing the path-following controller.

2.3.2 The Formation-Keeping Problem

The *formation* is defined by the relative positions of the vehicles. Let

$$\mathbf{p}_{\mathrm{rel},i} = \mathbf{p}_i - \mathbf{p}_b, \tag{2.61}$$

denote the position of vehicle i, relative to the barycenter. The goal of formation keeping is to control the vehicles such that

$$\mathbf{p}_{\mathrm{rel},i} \to \mathbf{p}_{f,i}, \qquad \forall i = 1, \ldots, N, \tag{2.62}$$

where $\mathbf{p}_{f,1}, \ldots, \mathbf{p}_{f,N}$ are vectors that represent the desired formation.

From (2.58) and (2.61), the sum of the relative positions is

$$\sum_{i=1}^{N} \mathbf{p}_{\mathrm{rel},i} = \sum_{i=1}^{N} \left(\mathbf{p}_i - \frac{1}{N} \sum_{j=1}^{N} \mathbf{p}_j \right) = \mathbf{0}. \tag{2.63}$$

Consequently, to make the formation-keeping problem feasible, $\mathbf{p}_{f,1}, \ldots, \mathbf{p}_{f,N}$ must be chosen such that

$$\sum_{i=1}^{N} \mathbf{p}_{f,i} = \mathbf{0}. \tag{2.64}$$

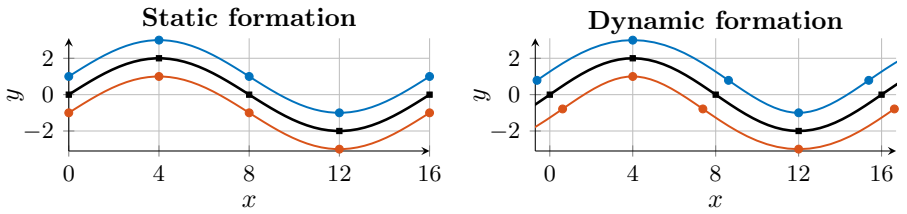
Formations can be split into two categories: static and dynamic. In *static* formations, the vectors $\mathbf{p}_{f,i}$ are constant. In *dynamic* formations, the vectors $\mathbf{p}_{f,i}$ are time-varying. In this thesis, we investigate a specific type of dynamic formations: formations that rotate with the desired path. In this type of formation, the desired relative positions are given by

$$\mathbf{p}_{f,i} = \mathbf{R}_p(s)\mathbf{p}^f_{f,i}, \qquad i = 1, \ldots, N, \tag{2.65}$$

where $\mathbf{p}^f_{f,i}$ is a constant vector.

An example of a static and a dynamic formation is shown in Figure 2.3. In both cases, the barycenter should follow a sine-wave path parametrized by

$$\mathbf{p}_p(s) = \left[s, 2\sin\left(\frac{\pi}{8}s\right) \right]^{\mathrm{T}}. \tag{2.66}$$



(a) Example of a static formation. **(b)** Example of a dynamic formation.

Figure 2.3: Examples of a static and a dynamic formation. The black line represents the desired path, the blue and red lines represent the desired positions of the vehicles. The markers represent the desired positions for $s = 0, 4, \ldots, 16$.

Figure 2.3a shows a static formation consisting of two vehicles, with the desired relative positions given by

$$\mathbf{p}_{f,1} = [0, 1]^{\mathrm{T}}, \qquad \mathbf{p}_{f,2} = [0, -1]^{\mathrm{T}}. \tag{2.67}$$

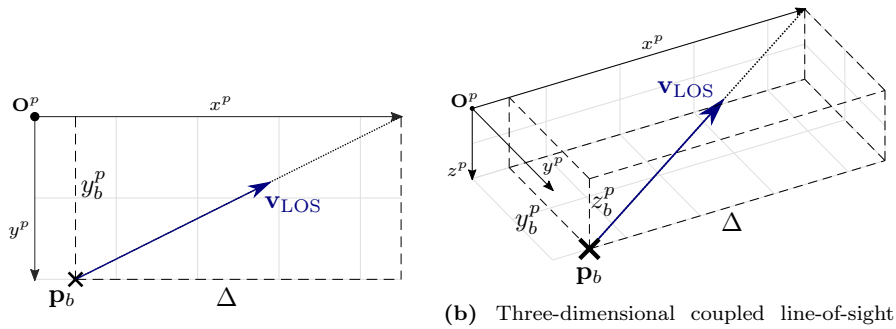
Figure 2.3a shows a dynamic formation that rotates with the desired path. The formation consists of two vehicles with the desired relative positions given by

$$\mathbf{p}_{f,i} = \mathbf{R}_p(s)\mathbf{p}^f_{f,i}, \qquad \mathbf{p}^f_{f,1} = [0, 1]^{\mathrm{T}}, \qquad \mathbf{p}^f_{f,2} = [0, -1]^{\mathrm{T}}. \tag{2.68}$$

2.4 Line-of-Sight Guidance

This section describes the LOS guidance algorithm. LOS is an intuitive method for steering vehicles towards the desired path. A review of LOS guidance methods for marine vehicles is presented in [95].

First, let us discuss LOS guidance for vehicles moving in the horizontal plane. Let $\mathbf{p}^p_b = [x^p_b, y^p_b]^{\mathrm{T}}$ denote the path-following error of the barycenter. Let $\mathbf{v}_{\mathrm{LOS}}$ denote the desired (inertial) line-of-sight velocity that steers the barycenter towards



(a) Two-dimensional line-of-sight guidance. **(b)** Three-dimensional coupled line-of-sight guidance.

Figure 2.4: Illustrations of line-of-sight guidance methods.

the desired path. This velocity is given by [91, 96]

$$\mathbf{v}_{\mathrm{LOS}} = U_{\mathrm{LOS}} \left[\cos(\chi_{\mathrm{LOS}}), \sin(\chi_{\mathrm{LOS}})\right]^{\mathrm{T}}, \qquad \chi_{\mathrm{LOS}} = \psi_p - \arctan\left(\frac{y_b^p}{\Delta}\right), \tag{2.69}$$

where $U_{\mathrm{LOS}} > 0$ is the desired path-following speed, ψ_p is the path-tangential angle, as defined in (2.46), and $\Delta > 0$ is the so-called *lookahead* distance. An illustration of LOS guidance in the horizontal plane is shown in Figure 2.4a.

For vehicles moving in three dimensions, there exist two types of LOS guidance algorithms: *decoupled* [97, 98] and *coupled* [99–101]. A decoupled LOS algorithm consists of two separate guidance schemes that steer the vehicle in the horizontal and vertical plane. Let $\mathbf{p}_b^p = [x_b^p, y_b^p, z_b^p]^{\mathrm{T}}$ denote the path-following error of the barycenter. Let us assume that the path-tangential coordinate frame is chosen according to (2.48), so that the rotation matrix $\mathbf{R}_p(s)$ has a zero roll angle. The desired line-of-sight velocity, $\mathbf{v}_{\mathrm{LOS}}$, is then given by

$$\mathbf{v}_{\mathrm{LOS}} = U_{\mathrm{LOS}} \begin{bmatrix} \cos(\gamma_{\mathrm{LOS}})\cos(\chi_{\mathrm{LOS}}) \\ \cos(\gamma_{\mathrm{LOS}})\sin(\chi_{\mathrm{LOS}}) \\ -\sin(\gamma_{\mathrm{LOS}}) \end{bmatrix}, \qquad \begin{aligned} \gamma_{\mathrm{LOS}} &= \theta_p + \arctan\left(\frac{z_b^p}{\Delta_z}\right), \\ \chi_{\mathrm{LOS}} &= \psi_p - \arctan\left(\frac{y_b^p}{\Delta_y}\right), \end{aligned} \tag{2.70}$$

where $U_{\mathrm{LOS}} > 0$ is the desired path-following speed, θ_p and ψ_p are the path-tangential angles, as defined in (2.49), and $\Delta_y, \Delta_z > 0$ are the lookahead distances of the horizontal and vertical guidance scheme, respectively. Comparing the decoupled guidance scheme in (2.70) to the two-dimensional LOS algorithm in (2.69), we can see that the decoupled guidance scheme effectively consists of two separate two-dimensional LOS guidance algorithms.

In a *coupled* LOS guidance scheme, the desired velocity is given by

$$\mathbf{v}_{\mathrm{LOS}} = \frac{U_{\mathrm{LOS}}}{\sqrt{\Delta^2 + y_b^{p2} + z_b^{p2}}} \mathbf{R}_p \left[\Delta, -y_b^p, -z_b^p\right]^{\mathrm{T}}, \tag{2.71}$$

where $U_{\mathrm{LOS}} > 0$ is the desired path-following speed, $\mathbf{R}_p$ is the rotation matrix between the path-tangential and the inertial coordinate frame, and $\Delta > 0$ is the lookahead distance. An illustration of this scheme is shown in Figure 2.4b. The coupled scheme can be seen as an extension of the horizontal LOS guidance scheme to three dimensions. Indeed, the two-dimensional guidance scheme (2.69) can also be written as

$$\mathbf{v}_{\mathrm{LOS}} = \frac{U_{\mathrm{LOS}}}{\sqrt{\Delta^2 + y_b^{p2}}} \begin{bmatrix} \Delta\cos(\psi_p) + y_b^p \sin(\psi_p) \\ \Delta\sin(\psi_p) - y_b^p \cos(\psi_p) \end{bmatrix} = \frac{U_{\mathrm{LOS}}}{\sqrt{\Delta^2 + y_b^{p2}}} \mathbf{R}_p \begin{bmatrix} \Delta \\ -y_b^p \end{bmatrix}. \tag{2.72}$$

Comparing (2.72) to (2.71), we can see that both equations have a similar form.

2.5 Null Space Behavioral Algorithm

This section describes the null-space-behavioral (NSB) algorithm. The NSB algorithm is a method that allows us to combine several tasks in a hierarchic manner.

The algorithm was originally developed for first-order systems

$$\dot{\mathbf{p}} = \mathbf{v}, \tag{2.73}$$

where $\mathbf{p} \in \mathbb{R}^n$ are the generalized coordinates, and $\mathbf{v} \in \mathbb{R}^n$ are the input velocities.

In NSB algorithms, the desired behavior of the system is divided into several tasks. Let there be M tasks arranged by priority in descending order (*i.e.,* task 1 has the highest priority, task M has the lowest priority). Let $\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_M$ denote the so-called *task variables*. Each variable is a function of the system coordinates

$$\boldsymbol{\sigma}_m = f_m(\mathbf{p}), \qquad f_m : \mathbb{R}^n \mapsto \mathbb{R}^{n_m}, \qquad \forall m = 1, \ldots, M, \tag{2.74}$$

where $n_m \leq n$ is the dimensionality of task m. Applying the chain rule, the time-derivative of $\boldsymbol{\sigma}_m$ is

$$\dot{\boldsymbol{\sigma}}_m = \frac{\partial f_m(\mathbf{p})}{\partial \mathbf{p}} \mathbf{v} \triangleq \mathbf{J}_m(\mathbf{p})\mathbf{v}. \tag{2.75}$$

Let $\dot{\boldsymbol{\sigma}}^*_m$ be the desired closed-loop behavior of the task variable. Then, the smallest input (in terms of Euclidean norm) that guarantees the desired behavior is

$$\mathbf{v}_m = \mathbf{J}^\dagger_m \dot{\boldsymbol{\sigma}}^*_m, \tag{2.76}$$

where $\mathbf{J}^\dagger_m$ is the Moore-Penrose pseudoinverse of the task Jacobian.

Remark. *In many applications, the task variable should track some pre-defined desired value, $\boldsymbol{\sigma}_{d,m}$. In such cases, we typically choose*

$$\dot{\boldsymbol{\sigma}}^*_m = \dot{\boldsymbol{\sigma}}_{d,m} - \boldsymbol{\Lambda}_m \left(\boldsymbol{\sigma}_m - \boldsymbol{\sigma}_{d,m} \right), \tag{2.77}$$

where $\boldsymbol{\Lambda}_m$ is a positive definite gain matrix. In the literature, (2.77) *is commonly referred to as the closed-loop inverse kinematics (CLIK) equation [102].*

If the task is *redundant*, *i.e.*, if the inequality $n_m < n$ strictly holds, then there exists a subspace of control inputs that do not interfere with the task. Let $\mathbf{v}_{\text{add}}$ be an additional input. Then, the following control input

$$\mathbf{v} = \mathbf{v}_m + \mathbf{N}_m \mathbf{v}_{\text{add}}, \tag{2.78}$$

where $\mathbf{N}_m = \mathbf{I}_N - \mathbf{J}^\dagger_m \mathbf{J}_m$ is the null-space projector of $\mathbf{J}_m$, guarantees the desired behavior of the task. The additional control input is satisfied only if it does not interfere with the task.

In the NSB algorithm, the control inputs from the individual tasks are composed by projecting the inputs from the lower-priority tasks onto the null space of the higher-priority tasks. In the literature, there exist two variants of the algorithm. The first variant calculates the control input $\mathbf{v}$ using the following equation

$$\mathbf{v} = \mathbf{v}_1 + \mathbf{N}_1 \Big(\mathbf{v}_2 + \mathbf{N}_2 \Big(\mathbf{v}_3 \cdots + \mathbf{N}_{M-2} \big(\mathbf{v}_{M-1} + \mathbf{N}_{M-1} \mathbf{v}_M \big) \Big) \Big), \tag{2.79}$$

with $\mathbf{v}_m$ given by (2.76).

The second variant uses the so-called *augmented* Jacobians

$$\bar{\mathbf{J}}_m = \left[\mathbf{J}_1^{\mathrm{T}}, \ldots, \mathbf{J}_m^{\mathrm{T}}\right]^{\mathrm{T}}. \tag{2.80}$$

Let $\bar{\mathbf{N}}_m$ denote the null-space projector of $\bar{\mathbf{J}}_m$. Then, the control input $\mathbf{v}$ is given by

$$\mathbf{v} = \mathbf{v}_1 + \sum_{m=2}^{M} \bar{\mathbf{N}}_{m-1}\mathbf{v}_m. \tag{2.81}$$

The advantages and disadvantages of both approaches are discussed in [103]. In this thesis, we will mostly use the first variant.

2.5.1 Independence and Orthogonality

The concepts of independence and orthogonality are important when analyzing the interactions between the tasks. Specifically, these concepts determine whether the tasks can be executed simultaneously, and how the null-space projector affects the lower-priority tasks.

Two tasks are *independent* if the pseudoinverses of their Jacobians are linearly independent. Let $\mathbf{J}_a$ and $\mathbf{J}_b$ denote the Jacobians of task a and b, respectively. These tasks are independent if

$$\operatorname{rank}\left(\mathbf{J}_a^{\dagger}\right) + \operatorname{rank}\left(\mathbf{J}_b^{\dagger}\right) = \operatorname{rank}\left(\left[\mathbf{J}_a^{\dagger}, \mathbf{J}_b^{\dagger}\right]\right), \tag{2.82}$$

Antonelli *et al.* [103] remark that the pseudoinverse and the transpose of the task Jacobian share the same span. Consequently, (2.82) is equivalent to

$$\operatorname{rank}\left(\mathbf{J}_a^{\mathrm{T}}\right) + \operatorname{rank}\left(\mathbf{J}_b^{\mathrm{T}}\right) = \operatorname{rank}\left(\left[\mathbf{J}_a^{\mathrm{T}}, \mathbf{J}_b^{\mathrm{T}}\right]\right). \tag{2.83}$$

Two tasks are *orthogonal* if the subspaces spanned by their Jacobians tasks are orthogonal, *i.e.,* if

$$\mathbf{J}_a\,\mathbf{J}_b^{\dagger} = \mathbf{O}_{n_a \times n_b}. \tag{2.84}$$

Now, let us consider two consecutive tasks that are independent and orthogonal. Without loss of generality, let us denote these tasks 1 and 2. The control input produced by combining these two tasks is

$$\mathbf{v} = \mathbf{J}_1^{\dagger}\dot{\boldsymbol{\sigma}}_1^{*} + \mathbf{N}_1\mathbf{J}_2^{\dagger}\dot{\boldsymbol{\sigma}}_2^{*} = \mathbf{J}_1^{\dagger}\dot{\boldsymbol{\sigma}}_1^{*} + \left(\mathbf{I} - \mathbf{J}_1^{\dagger}\mathbf{J}_1\right)\mathbf{J}_2^{\dagger}\dot{\boldsymbol{\sigma}}_2^{*} = \mathbf{J}_1^{\dagger}\dot{\boldsymbol{\sigma}}_1^{*} + \mathbf{J}_2^{\dagger}\dot{\boldsymbol{\sigma}}_2^{*}. \tag{2.85}$$

We have thus shown that if two consecutive tasks are independent and orthogonal, they can be executed simultaneously. Moreover, the null-space projector does not affect the lower-priority task.

2.5.2 NSB Algorithm for the Formation Path-Following Problem

In the remainder of this section, we demonstrate how the NSB algorithm can be used to solve the formation path-following problem. A similar scheme was proposed

in [33] for static formations, in [102, 103] for circular formations, and in [35] for dynamic formations.

Let $\mathbf{p}^{\mathrm{T}} = \left[\mathbf{p}_1^{\mathrm{T}}, \ldots, \mathbf{p}_N^{\mathrm{T}}\right]$ be the concatenated position vector of N vehicles. To solve the problem, we define two tasks: formation-keeping and path-following. The formation-keeping task has the highest priority. The task variable, $\boldsymbol{\sigma}_1$, is given by

$$\boldsymbol{\sigma}_1 = \left[\boldsymbol{\sigma}_{1,1}^{\mathrm{T}}, \boldsymbol{\sigma}_{1,2}^{\mathrm{T}}, \ldots, \boldsymbol{\sigma}_{1,N-1}^{\mathrm{T}}\right]^{\mathrm{T}}, \qquad \boldsymbol{\sigma}_{1,i} = \mathbf{p}_i - \mathbf{p}_b, \tag{2.86}$$

where $\mathbf{p}_b = \frac{1}{N}\sum_{i=1}^{N} \mathbf{p}_i$ is the barycenter of the formation (see Section 2.3).

Remark. *The formation-keeping task contains the relative positions of the first $N-1$ vehicles. The relative position of the last vehicle is omitted because it can be expressed as a linear combination of the remaining relative positions. Indeed, from* (2.86)*, we get*

$$\boldsymbol{\sigma}_{1,N} = \mathbf{p}_N - \mathbf{p}_b = -\sum_{i=1}^{N-1} \boldsymbol{\sigma}_{1,i}. \tag{2.87}$$

By omitting the last relative position vector, the task Jacobian has full row rank. Indeed, the Jacobian of the formation-keeping task is

$$\begin{aligned} \mathbf{J}_1 = \frac{\partial \boldsymbol{\sigma}_1}{\partial \mathbf{p}} &= \begin{bmatrix} \frac{N-1}{N}\mathbf{I}_3 & -\frac{1}{N}\mathbf{I}_3 & \cdots & -\frac{1}{N}\mathbf{I}_3 & -\frac{1}{N}\mathbf{I}_3 \\ -\frac{1}{N}\mathbf{I}_3 & \frac{N-1}{N}\mathbf{I}_3 & & -\frac{1}{N}\mathbf{I}_3 & -\frac{1}{N}\mathbf{I}_3 \\ \vdots & & \ddots & & \vdots \\ -\frac{1}{N}\mathbf{I}_3 & -\frac{1}{N}\mathbf{I}_3 & \cdots & \frac{N-1}{N}\mathbf{I}_3 & -\frac{1}{N}\mathbf{I}_3 \end{bmatrix} \\ &= \left([\mathbf{I}_{N-1},\, \mathbf{0}_{N-1}] - \frac{1}{N}\mathbf{1}_{(N-1)\times N}\right) \otimes \mathbf{I}_3. \end{aligned} \tag{2.88}$$

One can verify that the rank of $\mathbf{J}_1$ is $3(N-1)$, and the Jacobian thus has full row rank.

The desired value of the formation-keeping task variable is

$$\boldsymbol{\sigma}_{d,1} = \left[\mathbf{p}_{f,1}^{\mathrm{T}}, \mathbf{p}_{f,2}^{\mathrm{T}}, \ldots, \mathbf{p}_{f,N-1}^{\mathrm{T}}\right]^{\mathrm{T}}, \tag{2.89}$$

where $\mathbf{p}_{f,i}$ is the desired position of vehicle i within the formation (see Section 2.3).

The formation-keeping control input $\mathbf{v}_1$ can then be found, *e.g.*, using the CLIK equation (2.77)

$$\mathbf{v}_1 = \mathbf{J}_1^{\dagger}\left(\dot{\boldsymbol{\sigma}}_{d,1} - \boldsymbol{\Lambda}_1(\boldsymbol{\sigma}_1 - \boldsymbol{\sigma}_{d,1})\right), \tag{2.90}$$

where $\boldsymbol{\Lambda}_1$ is a positive definite gain matrix.

For the path-following task, the task variable is given by the position of the barycenter *i.e.,*

$$\boldsymbol{\sigma}_2 = \mathbf{p}_b = \frac{1}{N}\sum_{i=1}^{N} \mathbf{p}_i. \tag{2.91}$$

Proposition 2.1. *The formation-keeping and path-following tasks are independent and orthogonal.*

Proof. The Jacobian of the path-following task is given by

$$\mathbf{J}_2 = \frac{\partial \boldsymbol{\sigma}_2}{\partial \mathbf{p}} = \frac{1}{N}\mathbf{1}_{1\times N} \otimes \mathbf{I}_3. \tag{2.92}$$

The matrices $\mathbf{J}_1$ and $\mathbf{J}_2$ satisfy

$$\operatorname{rank}\left(\left[\mathbf{J}_1^{\mathrm{T}},\, \mathbf{J}_2^{\mathrm{T}}\right]\right) = 3N = \operatorname{rank}\left(\mathbf{J}_1^{\mathrm{T}}\right) + \operatorname{rank}\left(\mathbf{J}_2^{\mathrm{T}}\right), \tag{2.93}$$

and the tasks are thus independent. Moreover, the pseudoinverse of $\mathbf{J}_2$ is given by

$$\mathbf{J}_2^{\dagger} = \mathbf{1}_{N\times 1} \otimes \mathbf{I}_3. \tag{2.94}$$

One can then verify that the Jacobians satisfy

$$\mathbf{J}_1\mathbf{J}_2^{\dagger} = \mathbf{O}_{3(N-1)\times 3}, \tag{2.95}$$

and the tasks are thus orthogonal. □

The desired value of the path-following task is given by

$$\boldsymbol{\sigma}_{d,2} = \mathbf{p}_p(s), \tag{2.96}$$

where s is the value of the path parameter.

We propose to solve the path-following problem using line-of-sight guidance. The desired behavior of the path-following task is thus given by

$$\dot{\boldsymbol{\sigma}}_2^* = \mathbf{v}_{\mathrm{LOS}}, \tag{2.97}$$

where $\mathbf{v}_{\mathrm{LOS}}$ is either the coupled (2.71) or decoupled (2.70) LOS guidance law. The path-following control input $\mathbf{v}_2$ is then given by

$$\mathbf{v}_2 = \mathbf{J}_2^{\dagger}\dot{\boldsymbol{\sigma}}_2^* = \mathbf{1}_{N\times 1} \otimes \mathbf{v}_{\mathrm{LOS}}. \tag{2.98}$$

Thanks to the independence and orthogonality of the tasks, the combined control input $\mathbf{v}$ is given by

$$\mathbf{v} = \mathbf{v}_1 + \mathbf{N}_1\mathbf{v}_2 = \mathbf{v}_1 + \mathbf{v}_2. \tag{2.99}$$

Finally, let us investigate the closed-loop behavior of the tasks. First, we need to define the error variables. The formation-keeping error is defined as

$$\widetilde{\boldsymbol{\sigma}}_1 = \boldsymbol{\sigma}_1 - \boldsymbol{\sigma}_{1,d}. \tag{2.100}$$

The path-following error is given by the position of the barycenter in the path-tangential coordinate frame, *i.e.,*

$$\widetilde{\boldsymbol{\sigma}}_2 = \mathbf{p}_b^p = \mathbf{R}_p(s)^{\mathrm{T}}\left(\mathbf{p}_b - \mathbf{p}_p(s)\right). \tag{2.101}$$

Now, let us analyze the closed-loop behavior of the formation-keeping error. Differentiating (2.100) with respect to time, we get

$$\dot{\widetilde{\boldsymbol{\sigma}}}_1 = \mathbf{J}_1\mathbf{v} - \dot{\boldsymbol{\sigma}}_{1,d} = -\Lambda_1\widetilde{\boldsymbol{\sigma}}_1. \tag{2.102}$$

Since Λ_1 is positive definite by design, the closed-loop system (2.102) is globally exponentially stable (GES).

From (2.101), the dynamics of the path-following error are given by

$$\begin{aligned}\dot{\mathbf{p}}_b^p &= \mathbf{R}_p(s)^{\mathrm{T}}\left(\mathbf{J}_2\mathbf{v} - \dot{\mathbf{p}}_p(s)\right) - \mathbf{S}(\boldsymbol{\omega}_p(s))\mathbf{R}_p(s)^{\mathrm{T}}\left(\mathbf{p}_b - \mathbf{p}_p(s)\right) \\ &= \mathbf{R}_p(s)^{\mathrm{T}}\left(\mathbf{v}_{\mathrm{LOS}} - \dot{\mathbf{p}}_p(s)\right) - \mathbf{S}(\boldsymbol{\omega}_p(s))\mathbf{p}_b^p.\end{aligned} \tag{2.103}$$

The stability of the path-following task depends on the choice of the LOS guidance law. The stability of controllers that utilize decoupled and coupled LOS guidance will be discussed in Chapters 4 and 5, respectively.

2.6 Uniform Semiglobal Exponential Stability

This section discusses the concept of USGES. In some cases, dynamical systems cannot attain global stability due to, for instance, high-order nonlinearities, the choice of the control law, or actuator saturations. An example of such a system is marine vehicles controlled by line-of-sight guidance laws. In [104], it has been shown that the structure of proportional LOS guidance laws prevents the system from having global exponential convergence.

Uniform semiglobal exponential stability has been studied, *e.g.,* in [70, 105]. In these works, USGES is defined as follows.

Definition 2.1 (USGES)**.** *Consider a nonlinear system given by the following set of ordinary differential equations (ODEs)*

$$\dot{\mathbf{x}} = f(t, \mathbf{x}), \qquad \mathbf{x}(0) = \mathbf{x}_0, \tag{2.104}$$

with the origin $\mathbf{x} = \mathbf{0}$ being the equilibrium point of the system.

Let $\mathbf{x}(t|\mathbf{x}_0)$ be a solution to (2.104) that is defined for all $t \geq 0$. The origin $\mathbf{x} = \mathbf{0}$ is a USGES equilibrium point of (2.104) if for all $\Delta > 0$, there exist positive constants k_Δ and λ_Δ such that $\forall \mathbf{x}_0 \in \mathfrak{B}_\Delta^n$

$$\|\mathbf{x}(t|\mathbf{x}_0)\| \leq k_\Delta \|\mathbf{x}_0\| \mathrm{e}^{-\lambda_\Delta t}, \qquad \forall t \geq 0. \tag{2.105}$$

Remark. *The work in [70] studies parametric systems, i.e., systems with ODEs in the following form*

$$\dot{\mathbf{x}} = f(\mathbf{x}, t, \theta), \tag{2.106}$$

where $\theta \in \Theta \subset \mathbb{R}^m$ is a constant parameter. However, since this thesis does not consider parametric systems, and since the parameter θ is assumed constant, we can omit the parametric dependence for the sake of simplicity.

2.6.1 Lyapunov Sufficient Conditions for Uniform Semiglobal Exponential Stability

In this section, we restate Theorem 5 and Proposition 9 from [70].

Theorem 5 introduces sufficient conditions for uniform semiglobal exponential stability (USGES) of nonlinear systems.

Theorem 2.1 (Theorem 5. [70]). *Consider the nonlinear system given in* (2.104). *If for any* $\Delta > 0$, *there exist a continuously differentiable Lyapunov function* $V_\Delta : \mathbb{R}_{\geq 0} \times \mathfrak{B}_\Delta^n \mapsto \mathbb{R}_{\geq 0}$ *and positive constants* k_{1_Δ}, k_{2_Δ}, k_{3_Δ}, *and* a, *such that* $\forall \mathbf{x} \in \mathfrak{B}_\Delta^n$, $\forall t \geq 0$

$$k_{1_\Delta} \|\mathbf{x}\|^a \leq V_\Delta(t, \mathbf{x}) \leq k_{2_\Delta} \|\mathbf{x}\|^a, \tag{2.107a}$$

$$\lim_{\Delta \to \infty} \left(\frac{k_{1_\Delta}}{k_{2_\Delta}} \right)^{1/a} \Delta = \infty, \tag{2.107b}$$

$$\frac{\partial V_\Delta}{\partial t} + \frac{\partial V_\Delta}{\partial \mathbf{x}} f(t, \mathbf{x}) \leq -k_{3_\Delta} \|\mathbf{x}\|^a, \tag{2.107c}$$

then the origin of (2.104) *is USGES.*

Proposition 9 then introduces sufficient conditions for uniform semiglobal exponential stability (USGES) of nonlinear cascaded systems.

Proposition 2.2 (Proposition 9. [70]). *Consider the following cascaded nonlinear time-varying system*

$$\dot{\mathbf{x}}_1 = f_1(t, \mathbf{x}_1) + g(t, \mathbf{x}_1)\mathbf{x}_2, \tag{2.108a}$$

$$\dot{\mathbf{x}}_2 = f_2(t, \mathbf{x}_2), \tag{2.108b}$$

where $t \in \mathbb{R}_{\geq 0}$, $\mathbf{x}_1 \in \mathbb{R}^{n_1}$, $\mathbf{x}_2 \in \mathbb{R}^{n_2}$. *The functions* f_1, f_2, *and* g *are continuous in* t *and locally Lipschitz in* $\mathbf{x}_1$ *and* $\mathbf{x}_2$. *Furthermore,* f_1 *is assumed* $\mathcal{C}^1$ *in* t *and* $\mathbf{x}_1$, *and the origin* $\left[\mathbf{x}_1^{\mathrm{T}}, \mathbf{x}_2^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$ *is an equilibrium point of* (2.108).

Let each of the systems

$$\dot{\mathbf{x}}_1 = f_1(t, \mathbf{x}_1), \tag{2.109}$$

$$\dot{\mathbf{x}}_2 = f_2(t, \mathbf{x}_2), \tag{2.110}$$

be UGAS and satisfy the conditions of Theorem 2.1. Then, the origin of the cascaded system (2.108) *is USGES and UGAS if the following two assumptions hold*

1. *There exist constants* $c_1, c_2, \eta > 0$ *and a positive definite, radially unbounded Lyapunov function* $V : \mathbb{R}_{\geq 0} \times \mathbb{R}^{n_1}$ *of* (2.109) *such that* $\dot{V}(t, \mathbf{x}_1) \leq 0$ *and*

$$\left\| \frac{\partial V}{\partial \mathbf{x}_1} \right\| \|\mathbf{x}_1\| \leq c_1 V, \qquad \forall \|\mathbf{x}_1\| \geq \eta, \tag{2.111a}$$

$$\left\| \frac{\partial V}{\partial \mathbf{x}_1} \right\| \leq c_2, \qquad \forall \|\mathbf{x}_1\| \leq \eta. \tag{2.111b}$$

2. *There exist two continuous functions* $\alpha_1, \alpha_2 : \mathbb{R}_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$ *such that*

$$\|g(t, \mathbf{x}_1, \mathbf{x}_2)\| \leq \alpha_1(\|\mathbf{x}_2\|) + \alpha_2(\|\mathbf{x}_2\|)\|\mathbf{x}_1\|. \tag{2.112}$$

Chapter 3

Unifying Reactive Collision Avoidance and Control Allocation

To enable autonomous vehicles to operate in cluttered and unpredictable environments with numerous obstacles, such vehicles need a collision avoidance system that can react to and handle sudden changes in the environment. This chapter discusses an optimization-based reactive collision avoidance system that uses control barrier functions integrated into the control allocation. We demonstrate the effectiveness of this method through numerical simulations of autonomous surface vehicles. The simulated vehicles track their reference waypoints while maintaining safe distances. The proposed method can be readily implemented on vehicles that already use an optimization-based control allocation method. The contents of this chapter are based on [82].

The chapter is organized as follows. Section 3.1 defines the notation and describes the model of the vehicle. Section 3.2 defines the combined control allocation/collision avoidance problem. The proposed control allocation method and CBFs for solving this problem are then introduced in Sections 3.3 and 3.4. Section 3.5 describes the resulting combined COLAV and control allocation optimization problem. Finally, Section 3.6 presents the results of numerical simulations using models of ASVs.

3.1 Vehicle Model

In this chapter, we consider overactuated vehicles, *i.e.,* vehicles with more actuators than degrees of freedom (DOFs), with a control system consisting of blocks shown in Figure 3.1. The control system contains a long-term, deliberate planner, a high-level controller that outputs desired forces and torques ($\boldsymbol{\tau}_d$), and a control allocation block. The goal of control allocation is to find actuator control inputs ($\mathbf{u}$) that generate the desired forces and torques.

3.1.1 Notation

Let $\mathbf{p}$ denote the position and $\boldsymbol{\Theta}$ the orientation (expressed using the Euler angles) of the vehicle in a north-east-down (NED) reference frame. Let $\boldsymbol{\eta}$ be the pose of

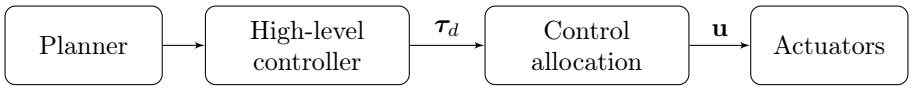


Figure 3.1: Control system of overactuated vehicles considered in this chapter

the vehicle

$$\boldsymbol{\eta} = \left[\mathbf{p}^{\mathrm{T}},\, \boldsymbol{\Theta}^{\mathrm{T}}\right]^{\mathrm{T}}. \tag{3.1}$$

Let $\boldsymbol{\nu}$ be the velocities of the vehicle in the body-fixed frame. The complete state of the vehicle, $\mathbf{x}$, is defined as

$$\mathbf{x} = \left[\boldsymbol{\eta}^{\mathrm{T}}\,, \boldsymbol{\nu}^{\mathrm{T}}\right]^{\mathrm{T}}. \tag{3.2}$$

Let $\boldsymbol{\tau}$ be the vector of generalized forces acting on the vehicle. Let K be the number of actuator parameters and $\mathbf{u} \in \mathbb{R}^K$ the vector of inputs. Furthermore, let $b : \mathbb{R}^K \to \mathbb{R}^{n_{\mathrm{DOF}}}$ be a nonlinear function that maps the inputs to the generalized forces (n_{DOF} is the number of DOFs).

3.1.2 Equations of Motion

The time-derivative of the pose can be obtained by transforming the velocities. In addition, we assume that the time-derivatives of the velocities are affine in the generalized forces. We thus consider vehicles described by the following dynamical equations

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\boldsymbol{\eta}} \\ \dot{\boldsymbol{\nu}} \end{bmatrix} = \begin{bmatrix} \mathbf{J}(\boldsymbol{\Theta})\,\boldsymbol{\nu} \\ f(\mathbf{x}) + g(\mathbf{x})\,\boldsymbol{\tau} \end{bmatrix} = \begin{bmatrix} \mathbf{J}(\boldsymbol{\Theta})\,\boldsymbol{\nu} \\ f(\mathbf{x}) + g(\mathbf{x})\,b(\mathbf{u}) \end{bmatrix}, \tag{3.3}$$

where $\mathbf{J}(\boldsymbol{\Theta})$ is the transformation matrix. This equation describes a large class of systems, including the matrix-vector model of marine vessels [91]

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\Theta})\,\boldsymbol{\nu}, \tag{3.4a}$$

$$\mathbf{M}\,\dot{\boldsymbol{\nu}} + (\mathbf{C}(\boldsymbol{\nu}) + \mathbf{D}(\boldsymbol{\nu}))\;\boldsymbol{\nu} + \mathbf{g}(\boldsymbol{\eta}) = b(\mathbf{u}), \tag{3.4b}$$

This model can be converted to the form in (3.3) since the matrix $\mathbf{M}$ is invertible.

3.2 Problem Definition

We consider a scenario with N vehicles. We shall denote the variables that belong to a given vehicle by a lower index (*e.g.,* $\mathbf{x}_i$ is the state of the i^{th} vehicle). Let us assume that each vehicle has access to the position ($\mathbf{p}_j$) and the inertial velocity ($\dot{\mathbf{p}}_j$) of all other vehicles.

Furthermore, let $\boldsymbol{\tau}_{d,i}$ be the desired forces and torques obtained from the high-level controller of vehicle i (see Figure 3.1). The goal of this chapter is to design a control allocation block that incorporates safety constraints. This block outputs actuator configuration $\mathbf{u}_i$ that produces the desired forces and torques as closely as

possible (*i.e.,* that minimizes the difference between $\boldsymbol{\tau}_{d,i}$ and $b(\mathbf{u}_i)$) while avoiding collisions with other vehicles. To avoid collisions, we want the vehicle i to satisfy

$$\|\mathbf{p}_i - \mathbf{p}_j\| \geq d_{\min}, \qquad \forall j \in \{1, \dots, N\} \setminus \{i\}, \tag{3.5}$$

where $d_{\min} > 0$ is some minimum safety distance.

3.3 Control Allocation

As stated in the Introduction, the goal of the control allocation is to find the inputs that generate the desired forces given by the high-level controller. For details on control allocation techniques for both linear and nonlinear systems, the reader is referred to [65]. Since control allocation is done individually for each vehicle, we can omit the lower index i in this section.

In this chapter, we consider systems where the function b can be nonlinear. In the literature, nonlinear control allocation is commonly solved by linearizing the function b [67, 68]

$$b(\mathbf{u}_0 + \Delta\mathbf{u}) \approx b(\mathbf{u}_0) + \mathbf{B}(\mathbf{u}_0)\,\Delta\mathbf{u}, \tag{3.6}$$

where $\mathbf{u}_0$ are the inputs around which we linearize, $\Delta\mathbf{u}$ is the increment, and

$$\mathbf{B}(\mathbf{u}_0) = \left.\frac{\partial b(\mathbf{u})}{\partial \mathbf{u}}\right|_{\mathbf{u}_0}, \tag{3.7}$$

is the Jacobian of b evaluated at $\mathbf{u}_0$. Let $\boldsymbol{\tau}_d$ be the desired forces. The goal of our control allocation scheme is to find optimal inputs $\mathbf{u}^*$ that satisfy

$$\mathbf{u}^* = \underset{\mathbf{u}\in\mathbb{R}^K}{\arg\min} \left\|b(\mathbf{u}) - \boldsymbol{\tau}_d\right\|^2. \tag{3.8}$$

Using the approximation (3.6), we can formulate the control allocation problem as a quadratic program (QP)

$$\mathbf{u}^* = \mathbf{u}_0 + \Delta\mathbf{u}^*, \tag{3.9}$$

$$\Delta\mathbf{u}^* = \underset{\Delta\mathbf{u}\in\mathbb{R}^K}{\arg\min} \left\|b(\mathbf{u}_0) + \mathbf{B}(\mathbf{u}_0)\,\Delta\mathbf{u} - \boldsymbol{\tau}_d\right\|^2. \tag{3.10}$$

3.4 Control Barrier Functions

In this section, we will briefly present the theory behind control barrier functions (CBFs). For more details, the reader is referred to [58]. After presenting the notation for multiple vehicles, we define the CBF for COLAV.

3.4.1 Introduction to CBFs

Consider a nonlinear control-affine system

$$\dot{\mathbf{x}} = \widetilde{f}(\mathbf{x}) + \widetilde{g}(\mathbf{x})\,\mathbf{u}, \tag{3.11}$$

where $\mathbf{x} \in \mathbb{R}^n$. Suppose that the system must satisfy a safety constraint $h(\mathbf{x}) \geq 0$, where $h : \mathbb{R}^n \to \mathbb{R}$ is the so-called *barrier function*. Then, we can define the so-called *safe set*, a set of all states that satisfy the safety constraint, as

$$\mathcal{C} = \{\mathbf{x} \,|\, h(\mathbf{x}) \geq 0\}\,. \tag{3.12}$$

If the initial condition of the system (3.11) lies in the safe set, the system trajectory will stay within $\mathcal{C}$ if the following inequality holds [58]

$$\frac{\mathrm{d}}{\mathrm{d}t}h(\mathbf{x}) = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \left(\widetilde{f}(\mathbf{x}) + \widetilde{g}(\mathbf{x})\,\mathbf{u}\right) \geq -\gamma\big(h(\mathbf{x})\big), \tag{3.13}$$

where γ is an extended class-$\mathcal{K}_\infty$ function. If there exists an input $\mathbf{u}$ such that (3.13) is satisfied, then h is a valid CBF for the system (3.11).

3.4.2 CBFs for Reactive Collision Avoidance

Let us define the relative position of vehicles i and j as

$$\mathbf{p}_{ij} = \mathbf{p}_i - \mathbf{p}_j. \tag{3.14}$$

To ensure safety, we need a collection of CBFs that enforce safe distances between each pair of vehicles. In the literature, vehicles described by the model (3.3) frequently use CBFs in the following form [55, 69]

$$h_{ij}(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{p}_{ij}\| - d_{\min} + k_v \frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{p}_{ij}\|, \tag{3.15}$$

where $d_{\min}$ is a minimum safe distance, and k_v is a coefficient that penalizes the relative speed of the vehicles.

To use h_{ij} as a control barrier function, we need to calculate its time-derivative. Differentiating (3.15) with respect to time yields

$$\frac{\mathrm{d}}{\mathrm{d}t}h_{ij}(\mathbf{x}_i, \mathbf{x}_j) = \frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{p}_{ij}\| + k_v \frac{\mathrm{d}^2}{\mathrm{d}t^2}\|\mathbf{p}_{ij}\|. \tag{3.16}$$

To calculate the first and second time-derivative of the relative distance, we need to find the first and second time-derivatives of the relative position. For $\dot{\mathbf{p}}_{ij}$, we split the derivative of $\boldsymbol{\eta}$ from (3.3) into the derivatives of position and orientation

$$\dot{\boldsymbol{\eta}}_i = \begin{bmatrix} \dot{\mathbf{p}}_i \\ \dot{\boldsymbol{\Theta}}_i \end{bmatrix} = \begin{bmatrix} \mathbf{J}_\mathbf{p}(\boldsymbol{\Theta}_i) \\ \mathbf{J}_\boldsymbol{\Theta}(\boldsymbol{\Theta}_i) \end{bmatrix} \boldsymbol{\nu}_i. \tag{3.17}$$

Substituting this into the time-derivative of (3.14) yields

$$\dot{\mathbf{p}}_{ij} = \mathbf{J}_\mathbf{p}(\boldsymbol{\Theta}_i)\,\boldsymbol{\nu}_i - \dot{\mathbf{p}}_j. \tag{3.18}$$

For $\ddot{\mathbf{p}}_{ij}$, we assume that the other vehicle maintains its velocity, *i.e.*,

$$\ddot{\mathbf{p}}_{ij} \approx \ddot{\mathbf{p}}_i, \tag{3.19}$$

when calculating the time-derivative for the i^{th} vehicle. As discussed in [69], this is a "mild worst-case" assumption, since maneuvers of the target vehicle tend to aid to resolving the situation. Thus, taking the time-derivative of (3.18) yields

$$\ddot{\mathbf{p}}_i = \dot{\mathbf{J}}_{\mathbf{p}}(\boldsymbol{\Theta}_i)\,\boldsymbol{\nu}_i + \mathbf{J}_{\mathbf{p}}(\boldsymbol{\Theta}_i)\,\dot{\boldsymbol{\nu}}_i. \tag{3.20}$$

Finally, we substitute the approximation of forces from (3.6) into the equation for $\dot{\boldsymbol{\nu}}$ in (3.3) to get

$$\dot{\boldsymbol{\nu}}_i = f(\mathbf{x}_i) + g(\mathbf{x}_i)\ \left(b(\mathbf{u}_{0,i}) + \mathbf{B}(\mathbf{u}_{0,i})\,\Delta\mathbf{u}_i\right), \tag{3.21}$$

which we can substitute into (3.20) to calculate $\ddot{\mathbf{p}}_i$.

3.5 Formulating the Optimization Problem

Now we can combine the definitions from Sections 3.3 and 3.4 to formulate the proposed optimization problem for control allocation with multi-vehicle COLAV.

3.5.1 The Basic Optimization Problem

Let $\mathbf{u}_{0,i}$ be the inputs of vehicle i from the previous control period. The new inputs are calculated as

$$\mathbf{u}_i = \mathbf{u}_{0,i} + \Delta\mathbf{u}_i^*, \tag{3.22}$$

where $\Delta\mathbf{u}_i^*$ is obtained by solving the following QP

$$\Delta\mathbf{u}_i^* = \underset{\Delta\mathbf{u}_i \in \mathbb{R}^K}{\arg\min} \left\| b(\mathbf{u}_{0,i}) + \mathbf{B}(\mathbf{u}_{0,i})\,\Delta\mathbf{u}_i - \boldsymbol{\tau}_{d,i} \right\|^2, \tag{3.23a}$$

$$\text{s.t.}\ \frac{\mathrm{d}}{\mathrm{d}t} h_{ij}(\mathbf{x}_i, \mathbf{x}_j) \geq -\gamma\left(h_{ij}(\mathbf{x}_i, \mathbf{x}_j)\right), \quad j \in \{1, \ldots, N\} \setminus \{i\}, \tag{3.23b}$$

$$\mathbf{u}_{i,\min} \leq \mathbf{u}_{0,i} + \Delta\mathbf{u}_i \leq \mathbf{u}_{i,\max}, \tag{3.23c}$$

$$\Delta\mathbf{u}_{i,\min} \leq \Delta\mathbf{u}_i \leq \Delta\mathbf{u}_{i,\max}, \tag{3.23d}$$

where $\mathbf{u}_{i,\min}$ and $\mathbf{u}_{i,\max}$ are the absolute actuator limits, and $\Delta\mathbf{u}_{i,\min}$ and $\Delta\mathbf{u}_{i,\max}$ are the actuator rate limits. The absolute limits are usually given by the physical limitations of the vehicle (*e.g.,* the thrust of a propeller or the deflection of control surfaces) whereas the rate limits are user-defined to reduce the rapid changes that wear out the actuators.

Simulation results using this control allocation algorithm are presented in Section 3.6.

3.5.2 Modified Optimization Problem

The algorithm in (3.23) is suitable for vehicles where the number of actuators is equivalent to the number of DOFs. Applying the algorithm to vehicles where the number of actuators is much greater than the number of DOFs results in inefficient usage of the available actuators, as can be seen in Section 3.6.

To reduce this effect, we add penalty terms on the actuator usage, similar to those proposed in [68], in the cost function. To simplify the notation, let $\|\mathbf{x}\|^2_{\mathbf{Q}}$ be the squared norm of a vector $\mathbf{x}$ weighted by a matrix $\mathbf{Q}$, *i.e.,*

$$\|\mathbf{x}\|^2_{\mathbf{Q}} = \mathbf{x}^{\mathrm{T}}\,\mathbf{Q}\,\mathbf{x}. \tag{3.24}$$

The modified optimization problem is defined as follows

$$\begin{aligned}\Delta\mathbf{u}_i^* = \underset{\Delta\mathbf{u}_i\in\mathbb{R}^K}{\arg\min}\ & \|b(\mathbf{u}_{i,0}) + \mathbf{B}(\mathbf{u}_{i,0})\,\Delta\mathbf{u}_i - \boldsymbol{\tau}_{d,i}\|^2_{\mathbf{Q}} \\ & + \|\mathbf{u}_{0,i} + \Delta\mathbf{u}_i\|^2_{\mathbf{R}_{\text{abs}}} + \|\Delta\mathbf{u}_i\|^2_{\mathbf{R}_{\text{rel}}}\,,\end{aligned} \tag{3.25a}$$

$$\text{s.t. constraints (3.23b)–(3.23d)}, \tag{3.25b}$$

where $\mathbf{Q}$ is a positive definite matrix that penalizes the difference between the desired and actual forces, and $\mathbf{R}_{\text{abs}}$ and $\mathbf{R}_{\text{rel}}$ are positive semidefinite matrices that penalize the absolute and incremental usage of actuators, respectively.

Note that both (3.23) and (3.25) use only local information and measurements, and can thus be solved locally for each vehicle.

When choosing the weight matrices, we first note that the vector $\boldsymbol{\tau}$ contains both forces and torques. The matrix $\mathbf{Q}$ should penalize them differently. In the simulations in Section 3.6, we choose

$$\mathbf{Q} = \operatorname{diag}\left(1, 1, \frac{1}{L^2}\right), \tag{3.26}$$

where $\operatorname{diag}(\cdot)$ is a diagonal matrix and L is the smallest distance of the thrusters from the center of mass. The matrix $\mathbf{Q}$ is chosen according to (3.26) because the term $\boldsymbol{\tau}_{d,i}$ contains both forces and torques. Specifically, the third element of $\boldsymbol{\tau}_{d,i}$ is the desired yaw torque. If we divide the squared toruqe error by L^2, we effectively convert it to a squared force error.

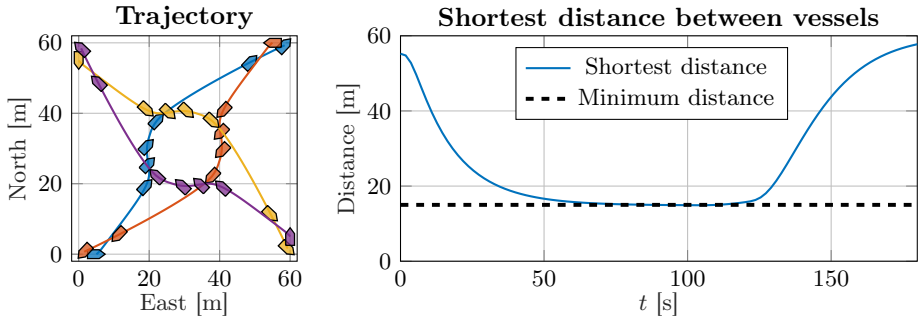
3.6 Simulations

In the simulations, we test the ability of the proposed algorithms to resolve a situation when four surface vessels are simultaneously in danger of collision. Each vessel starts in the corner of a square and is guided towards a reference located in the diagonally opposite corner.

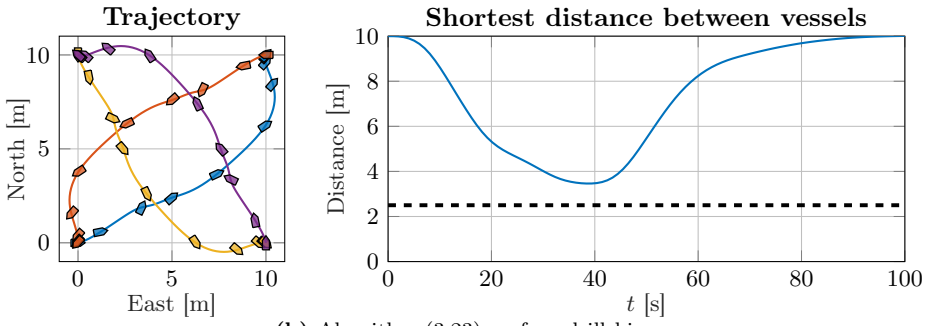
We tested the proposed algorithms on two models of ASVs — the *milliAmpere* ferry [106] and the 1 : 90 scaled model of the Inocean Cat I drillship [107] — using Simulink. Both vessels are equipped with azimuth thrusters; the *milliAmpere* has two and the drillship has six. Each thruster is parametrized by two values: its thrust force and its azimuth. The input vector for these vessels is defined as

$$\mathbf{u} = [f_1, \ldots, f_k, \alpha_1, \ldots, \alpha_k]^{\mathrm{T}}, \tag{3.27}$$

where f_i is the thrust force and α_i is the azimuth angle of the i^{th} thruster, and k is the number of thrusters. Both ASV models have 3DOFs, *i.e.,* the North-East position and the yaw angle. The function that maps the inputs to the generalized forces is



(a) Algorithm (3.23) on four *milliAmpere* vessels



(b) Algorithm (3.23) on four drillships

Figure 3.2: Simulations of the control allocation algorithm (3.23)

$$b(\mathbf{u}) = \textstyle\sum_{i=1}^{k} f_i \left[\cos\alpha_i, \sin\alpha_i, L_x^i \sin\alpha_i - L_y^i \cos\alpha_i\right]^{\mathrm{T}}, \tag{3.28}$$

where L_x^i and L_y^i are the x- and y-components of the position of the i^{th} thruster, relative to the center of mass.

For the higher-level controller that provides the desired forces, we use a nonlinear PID controller [91]. The nonlinear PID is an output-linearizing controller that transforms the nonlinear dynamical equations from (3.4) to

$$\ddot{\boldsymbol{\eta}} + 2\,\boldsymbol{\Omega}_n\,\mathbf{Z}\,\dot{\boldsymbol{\eta}} + \boldsymbol{\Omega}_n^2\,\boldsymbol{\eta} = 0, \tag{3.29}$$

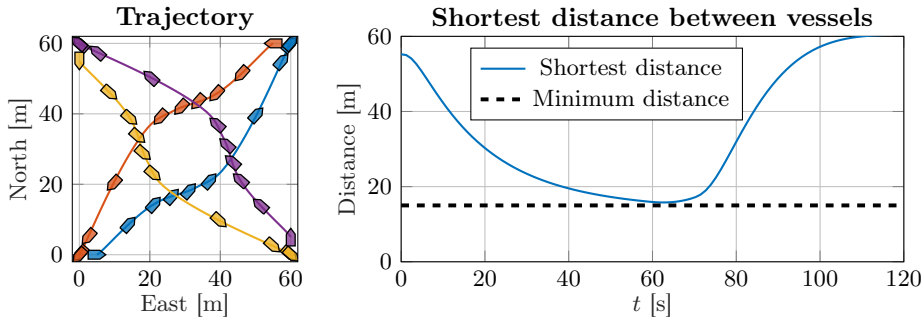
where $\mathbf{Z}$ is the diagonal relative damping matrix, and $\boldsymbol{\Omega}_n$ is the diagonal natural frequency matrix. Both matrices are tuning parameters. For convenience, we express $\boldsymbol{\Omega}_n$ in terms of a bandwidth matrix $\boldsymbol{\Omega}_{bw}$

$$\boldsymbol{\Omega}_n = \boldsymbol{\Omega}_{bw} \left(\sqrt{\mathbf{I} - 2\,\mathbf{Z}^2 + \sqrt{4\,\mathbf{Z}^4 - 4\,\mathbf{Z}^2 + 2\,\mathbf{I}}}\right)^{-1}, \tag{3.30}$$

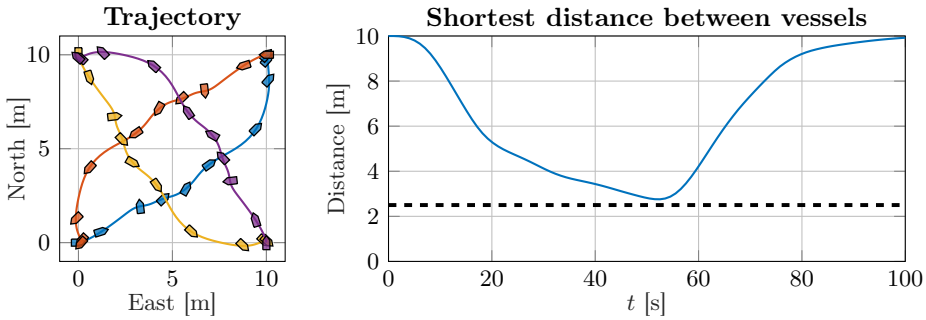
where $\sqrt{\cdot}$ is an elementwise square root.

The simulation parameters for both vessels are summarized in Table 3.1. Since the power consumption of a thruster increases with the absolute value of its thrust force and the increment of its azimuth, the matrices $\mathbf{R}_{\text{abs}}$ and $\mathbf{R}_{\text{rel}}$ are chosen as

$$\mathbf{R}_{\text{abs}} = \begin{bmatrix} r_{\text{abs}}\,\mathbf{I}_k & \mathbf{O}_{k\times k} \\ \mathbf{O}_{k\times k} & \mathbf{O}_{k\times k} \end{bmatrix}, \; \mathbf{R}_{\text{rel}} = \begin{bmatrix} \mathbf{O}_{k\times k} & \mathbf{O}_{k\times k} \\ \mathbf{O}_{k\times k} & r_{\text{rel}}\,\mathbf{I}_k \end{bmatrix}, \tag{3.31}$$



(a) Algorithm (3.25) on four *milliAmpere* vessels

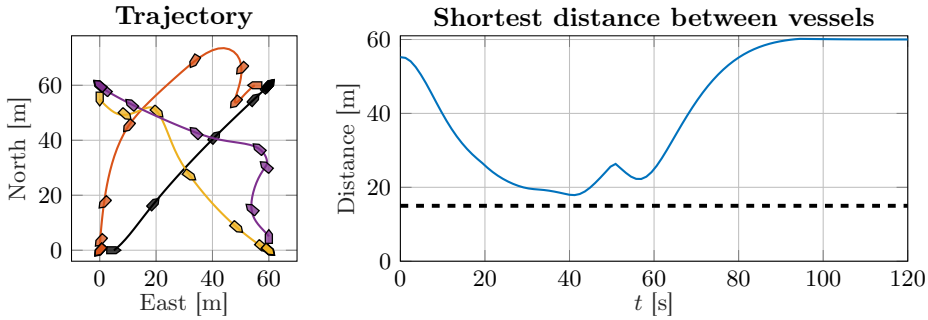


(b) Algorithm (3.25) on four drillships

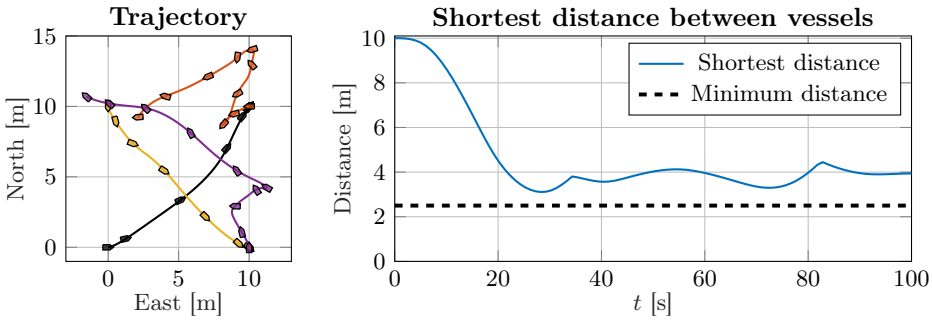
Figure 3.3: Simulations of the modified control allocation algorithm (3.25)

Parameter	*milliAmpere*	drillship
$\mathbf{\Omega}_{bw}$	diag (0.1, 0.1, 0.5)	
$\mathbf{Z}$	diag (0.95, 0.95, 0.97)	
$\mathbf{Q}$	diag $(1, 1, 0.7)$	diag $(1, 1, 1.13)$
r_{abs}	1	1
r_{rel}	100	1
$d_{\min}$ [m]	15	2.5
k_v [s]	15	15
$\gamma(h)$	$0.1\,h$	$0.1\,h$
$f_{\min}$ [N]	-350	-0.8
$f_{\max}$ [N]	500	1.5
$\Delta f_{\max}$ [N]	350	0.5
$\Delta\alpha_{\max}$ [rad]	$\frac{\pi}{8}$	$\frac{\pi}{8}$

Table 3.1: Simulation parameters. Parameters $\mathbf{\Omega}_{bw}$ and $\boldsymbol{Z}$ are identical for both scenarios, diag (.) is a diagonal matrix



(a) Algorithm (3.25) on four *milliAmpere* vessels with one uncontrolled vessel



(b) Algorithm (3.25) on four drillships with one uncontrolled vessel

Figure 3.4: Simulations of the modified control allocation algorithm (3.25) with one uncontrolled vessel (plotted in black)

The rate constraints are identical for all thrusters and symmetric, *i.e.*,

$$\Delta \mathbf{u}_{\max} = \begin{bmatrix} \Delta f_{\max} \mathbf{1}_k \\ \Delta \alpha_{\max} \mathbf{1}_k \end{bmatrix}, \qquad \Delta \mathbf{u}_{\min} = -\Delta \mathbf{u}_{\max}, \tag{3.32}$$

where $\Delta f_{\max}$ and $\Delta \alpha_{\max}$ are the force and azimuth rate constraints, respectively, and $\mathbf{1}_k$ is a vector of ones.

The results of the simulations are shown in Figures 3.2, 3.3, and 3.4. Figure 3.2 shows the results of algorithm (3.23). Figure 3.3 shows the results of algorithm (3.25). Each figure consists of two plots. The plot on the left displays the trajectory of the vessels. The colored lines show the trajectory of each vessel and the boat-shaped polygons represent the pose of the vessels at several evenly spaced time-instances. The plot on the right shows the smallest distance between the vessels compared to the minimum safe distance $d_{\min}$. In both scenarios, the vessels reach their reference position while maintaining safe distance.

We also tested a scenario where one of the vessels is uncontrolled. The results are shown in Figure 3.4. In this scenario, the uncontrolled vessel (plotted in black) solves the control allocation problem without the CBF constraints (3.23b). Although the time it takes the vessels to converge to their goal positions is greater, the minimum safe distance is still maintained. Note that in Figure 3.4b, the red vessel does not seem to converge to its desired position. This is because the sim-

Vessel	Scenario	Thruster utilization [%]		
		Maximum	Minimum	Mean
milliAmpere	**basic**	2.074	1.550	1.822
	modified	0.838	0.835	0.837
drillship	**basic**	100.000	1.282	51.496
	modified	6.161	0.259	3.816

Table 3.2: Steady-state thruster utilization of the *basic* algorithm (3.23) and the *modified* algorithm (3.25).

ulation was terminated too early, after 100 seconds. Given more time, the vessel would eventually converge to its desired position.

In this section, we have provided some insight into how to chose some of the parameters for the simulated models. When it comes to the choice of the coefficient k_v, introduced in (3.15), and the extended class-$\mathcal{K}_\infty$ function γ, introduced in (3.23), the following considerations can be made. Intuitively, increasing k_v increases the size of the "unsafe" region where the barrier function is negative, causing the system to react sooner in situations where two vehicles are on collision course. Conversely, increasing the slope of γ decreases the size of the region where the constraint (3.23b) is active, causing the system to react later.

Part I

Formation Path-Following using the Null Space Behavioral Algorithm

Chapter 4

Formation Path-Following Control of 5DOF Underactuated AUVs

An ant is very stupid ...
and yet, many ants together are smart.

Kurzgesagt — In a Nutshell, "Emergence,"
youtu.be/16W7c0mb-rE.

This chapter presents a novel method for formation path following of multiple underactuated autonomous underwater vehicles. The method combines line-of-sight guidance with null-space-behavioral control, allowing the vehicles to follow curved paths while maintaining the desired formation. We investigate the dynamics of the path-following error using cascaded systems theory, and show that the closed-loop system is uniformly semiglobally exponentially stable. We validate the theoretical results through numerical simulations. The contents of this chapter are based on [83].

The chapter is organized as follows. Section 4.1 defines the formation path-following problem that is addressed in this chapter. In Section 4.2, we describe the control system. The stability of the control system is proven in Section 4.3. Finally, Section 4.4 contains the results of a numerical simulation.

4.1 Problem Definition

In this section, we briefly present the AUV model and the formation path-following problem.

4.1.1 Vehicle Model

We consider a fleet of N underactuated AUVs. The dynamics are described using the 5DOF control-oriented model from Section 2.1.4. The pose ($\boldsymbol{\eta}$) and velocities ($\boldsymbol{\nu}$) of the AUVs are defined as

$$\boldsymbol{\eta} = [x, y, z, \theta, \psi]^{\mathrm{T}}, \qquad \boldsymbol{\nu} = [u, v, w, q, r]^{\mathrm{T}}. \tag{4.1}$$

The roll dynamics are disregarded as the roll motion is assumed to be small and self-stabilizing by the vehicle design. Let $\mathbf{V}_c \in \mathbb{R}^3$ be the velocity of an unknown, constant and irrotational ocean current.

Recalling (2.34), the dynamics of the AUVs are

$$\dot{x} = u\cos(\psi)\cos(\theta) - v\sin(\psi) + w\cos(\psi)\sin(\theta), \tag{4.2a}$$
$$\dot{y} = u\cos(\theta)\sin(\psi) + v\cos(\psi) + w\sin(\psi)\sin(\theta), \tag{4.2b}$$
$$\dot{z} = -u\sin(\theta) + w\cos(\theta), \tag{4.2c}$$
$$\dot{\theta} = q, \tag{4.2d}$$
$$\dot{\psi} = \frac{1}{\cos(\theta)} r, \tag{4.2e}$$
$$\dot{u} = f_u + F_u(u,v,w,q,r) + \boldsymbol{\phi}_u(u,v,w,q,r,\theta,\psi)^{\mathrm{T}}\mathbf{V}_c, \tag{4.2f}$$
$$\dot{v} = X_v(u,u_c)r + Y_v(u,u_c)v_r, \tag{4.2g}$$
$$\dot{w} = X_w(u,u_c)q + Y_w(u,u_c)w_r + G(\theta), \tag{4.2h}$$
$$\dot{q} = t_q + F_q(u,w,q,\theta) + \boldsymbol{\phi}_q(u,w,q,\theta,\psi)^{\mathrm{T}}\mathbb{V}_c, \tag{4.2i}$$
$$\dot{r} = t_r + F_r(u,v,r) + \boldsymbol{\phi}_r(u,v,r,\theta,\psi)^{\mathrm{T}}\mathbb{V}_c. \tag{4.2j}$$

4.1.2 Control Objectives

The goal is to control the AUVs so that they move in a prescribed formation while avoiding collisions, and their barycenter follows a given path.

The prescribed path is parametrized by a smooth function $\mathbf{p}_p : \mathbb{R} \to \mathbb{R}^3$. We assume that the parametrization is $\mathcal{C}^2$ and regular. Therefore, for every point $\mathbf{p}_p(s)$ on the path, there exist path-tangential angles, $\theta_p(s)$ and $\psi_p(s)$, and a corresponding path-tangential coordinate frame (see Section 2.2 for more details).

The path-following error $\mathbf{p}_b^p$ is given by the position of the barycenter expressed in the path-tangential coordinate frame

$$\mathbf{p}_b^p = \mathbf{R}_p(s)^{\mathrm{T}}\left(\mathbf{p}_b - \mathbf{p}_p(s)\right), \tag{4.3}$$

where

$$\mathbf{p}_b = \frac{1}{N}\sum_{i=1}^{N}\mathbf{p}_i, \qquad \mathbf{p}_i = [x_i, y_i, z_i]^{\mathrm{T}}. \tag{4.4}$$

The vehicles should converge to a dynamic formation that rotates with the desired path (see Section 2.3.2 for details). Let $\mathbf{p}_{f,1}^f, \ldots, \mathbf{p}_{f,n}^f$ be the position vectors that represent the desired formation. The objective is to control the vehicles so that

$$\mathbf{p}_i - \mathbf{p}_b \to \mathbf{R}_p(s)\mathbf{p}_{f,i}^f, \qquad \forall i \in \{1,\ldots,N\}. \tag{4.5}$$

We propose to solve this problem using an NSB algorithm. We note that the proposed algorithm is *centralized*. Consequently, to implement this algorithm in

a real-life situation, there must be a central node that can communicate and coordinate with all the AUVs. Alternatively, every AUV must have access to the complete state of all other AUVs.

4.2 Control System

To solve the formation path following problem, we propose a method that combines collision avoidance (COLAV), formation keeping, and path following in a hierarchic manner using an NSB algorithm. Since the NSB algorithm outputs inertial velocity references, we also need a low-level attitude control system to track these references.

In this section, we first present the attitude control system. Then, in Section 4.2.2, we present the NSB algorithm and the associated tasks. Finally, in Section 4.2.3, we demonstrate how to use the update law of the path variable to cancel unwanted terms in the path-following error dynamics.

4.2.1 Attitude Control System

This system controls the surge velocity, pitch, and yaw via the corresponding accelerations. The system is based on the autopilots in [71], but extended to five degrees of freedom.

Let u_d be the desired surge velocity and $\dot{u}_d$ its derivative. Let $\hat{\mathbf{V}}_c$ be the estimate of the ocean current. Furthermore, let us define $\tilde{u} = u - u_d$ and $\tilde{\mathbf{V}}_c = \hat{\mathbf{V}}_c - \mathbf{V}_c$. The surge controller consists of an output-linearizing sliding-mode P-controller and an ocean current observer

$$f_u = \dot{u}_d - F_u(\cdot) - \boldsymbol{\phi}_u(\cdot)^{\mathrm{T}}\,\hat{\mathbf{V}}_c - k_u\,\tilde{u} - k_c\,\mathrm{sign}\,(\tilde{u})\,, \tag{4.6}$$

$$\dot{\hat{\mathbf{v}}}_c = c_u\,\boldsymbol{\phi}_u(\cdot)\,\tilde{u}, \tag{4.7}$$

where k_u, k_c and c_u are positive gains.

Let θ_d be the desired pitch angle and $\dot{\theta}_d, \ddot{\theta}_d$ its derivatives. Let $\hat{\mathbb{V}}_q$ be the estimate of $\mathbb{V}_c$. Furthermore, let us define $\tilde{\theta} = \theta - \theta_d$, $\tilde{q} = q - \dot{\theta}_d$ and $\tilde{\mathbb{V}}_q = \hat{\mathbb{V}}_q - \mathbb{V}_c$. Inspired by [108], we introduce the following transformation

$$s_q = \tilde{q} + \lambda_q\,\tilde{\theta}, \tag{4.8}$$

where λ_q is a positive constant. The pitch controller consists of an output-linearizing sliding-mode PD-controller and an ocean current observer

$$\begin{aligned} t_q = \ddot{\theta}_d - F_q(\cdot) - \boldsymbol{\phi}_q(\cdot)^{\mathrm{T}}\,\hat{\mathbb{V}}_q - \lambda_q\,\tilde{q} \\ - k_\theta\,\tilde{\theta} - k_q\,s_q - k_d\,\mathrm{sign}(s_q), \end{aligned} \tag{4.9}$$

$$\dot{\hat{\mathbb{V}}}_q = c_q\,\boldsymbol{\phi}_q(\cdot)\,s_q, \tag{4.10}$$

where k_θ, k_q, k_d and c_q are positive gains.

Let ψ_d be the desired yaw angle and $\dot{\psi}_d, \ddot{\psi}_d$ its derivatives. Let $\hat{\mathbb{V}}_r$ be the estimate of $\mathbb{V}_c$. Furthermore, let us define $\tilde{\psi} = \psi - \psi_d$ and $\tilde{\mathbb{V}}_r = \hat{\mathbb{V}}_r - \mathbb{V}_c$. Similarly to the pitch controller, we introduce the following transformation

$$s_r = \dot{\tilde{\psi}} + \lambda_r\,\tilde{\psi} = \frac{r}{\cos\theta} - \dot{\psi}_d + \lambda_r\,\tilde{\psi}, \tag{4.11}$$

where λ_r is a positive constant. The yaw controller is analogous to the pitch controller introduced previously

$$\begin{aligned} t_r = &-F_r(\cdot) - \boldsymbol{\phi}_r(\cdot)^{\mathrm{T}}\,\hat{\mathbb{V}}_r - r\,\tan(\theta)\dot{\theta} \\ &+ \cos(\theta)\left(\ddot{\psi}_d - \lambda_r\,\dot{\tilde{\lambda}} - k_\psi\,\tilde{\psi} - k_r\,s_r - k_d\,\mathrm{sign}(s_r)\right), \end{aligned} \tag{4.12}$$

$$\dot{\hat{\mathbb{V}}}_r = c_r\,\boldsymbol{\phi}_r(\cdot)\,s_r, \tag{4.13}$$

where k_ψ, k_r, k_d and c_r are positive gains.

4.2.2 NSB Tasks

Let us denote the variables associated with the COLAV, formation-keeping, and path-following tasks by lower indices 1, 2, and 3, respectively. Each task produces a vector of desired velocities, $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \in \mathbb{R}^{3N}$.

First, let us consider the COLAV task. Let d_{COLAV} be the *activation distance*, *i.e.,* the distance at which the vehicles need to start performing the evasive maneuvers. The task variable is then given by a vector of relative distances between the vehicles smaller than d_{COLAV}, *i.e.,*

$$\boldsymbol{\sigma}_1 = \left[\, \|\mathbf{p}_i - \mathbf{p}_j\| \,\right]^{\mathrm{T}}, \qquad \begin{array}{c} \forall i, j \in \{1, \ldots, N\}, j > i, \\ \|\mathbf{p}_i - \mathbf{p}_j\| < d_{\mathrm{COLAV}}. \end{array} \tag{4.14}$$

The desired value of the task variable is

$$\boldsymbol{\sigma}_{d,1} = d_{\mathrm{COLAV}}\,\mathbf{1}, \tag{4.15}$$

where $\mathbf{1}$ is a vector of ones of the corresponding size. The velocity associated with the COLAV task is given by

$$\mathbf{v}_1 = -\mathbf{J}_1^{\dagger}\,\boldsymbol{\Lambda}_1\,\widetilde{\boldsymbol{\sigma}}_1, \tag{4.16}$$

where $\boldsymbol{\Lambda}_1$ is a positive definite gain matrix, and $\widetilde{\boldsymbol{\sigma}}_1 = \boldsymbol{\sigma}_1 - \boldsymbol{\sigma}_{d,1}$. Note that this task does not guarantee robust collision avoidance. During the transients, the relative distance may become smaller than d_{COLAV}. Therefore, to ensure collision avoidance, d_{COLAV} shuld be chosen as $d_{\min} + d_{\mathrm{sec}}$, where $d_{\min}$ is the minimum safe distance between the vehicles, and d_{sec} is an additional security distance.

The formation-keeping and path-following tasks are defined identically as in Section 2.5.2. The task variable of the formation-keeping task is

$$\boldsymbol{\sigma}_2 = \left[\boldsymbol{\sigma}_{2,1}^{\mathrm{T}}, \ldots, \boldsymbol{\sigma}_{2,N-1}^{\mathrm{T}}\right]^{\mathrm{T}}, \qquad \boldsymbol{\sigma}_{2,i} = \mathbf{p}_i - \mathbf{p}_b, \tag{4.17}$$

and its desired values are

$$\boldsymbol{\sigma}_{d,2} = \begin{bmatrix} \mathbf{R}\left(\theta_p(s), \psi_p(s)\right)\,\mathbf{p}_{f,1}^{f} \\ \vdots \\ \mathbf{R}\left(\theta_p(s), \psi_p(s)\right)\,\mathbf{p}_{f,N-1}^{f} \end{bmatrix}. \tag{4.18}$$

The desired velocity of the formation-keeping task is given by

$$\mathbf{v}_2 = \mathbf{J}_2^{\dagger} \left(\dot{\boldsymbol{\sigma}}_{d,2} - \boldsymbol{\Lambda}_2 \, \tilde{\boldsymbol{\sigma}}_2 \right), \tag{4.19}$$

where $\widetilde{\boldsymbol{\sigma}}_2 = \boldsymbol{\sigma}_2 - \boldsymbol{\sigma}_{d,2}$ is the error, and $\boldsymbol{\Lambda}_2$ is a positive definite gain matrix.

The task variable and the desired value of the path-following task is given by $\mathbf{p}_b$ and $\mathbf{p}_p(s)$, respectively. The desired velocity of the path-following task is obtained using the decoupled LOS guidance algorithm (2.70). We choose the same lookahead distance for the horizontal and vertical guidance schemes, *i.e.*, $\Delta_y = \Delta_z = \Delta$. Inspired by [109], we employ a time-varying error-dependent lookahead distance

$$\Delta \left(\mathbf{p}_b^p \right) = \sqrt{\Delta_0^2 + \left(x_b^p \right)^2 + \left(y_b^p \right)^2 + \left(z_b^p \right)^2}, \tag{4.20}$$

where $\Delta_0 > 0$ is a constant. The desired velocity of the path-following task is then given by

$$\mathbf{v}_3 = \mathbf{1}_N \otimes \mathbf{v}_{\mathrm{LOS}}, \tag{4.21}$$

where

$$\mathbf{v}_{\mathrm{LOS}} = U_{\mathrm{LOS}} \begin{bmatrix} \cos(\gamma_{\mathrm{LOS}}) \cos(\chi_{\mathrm{LOS}}) \\ \cos(\gamma_{\mathrm{LOS}}) \sin(\chi_{\mathrm{LOS}}) \\ -\sin(\gamma_{\mathrm{LOS}}) \end{bmatrix}, \quad \begin{aligned} \gamma_{\mathrm{LOS}} &= \theta_p + \arctan\left(\frac{z_b^p}{\Delta(\mathbf{p}_b^p)} \right), \\ \chi_{\mathrm{LOS}} &= \psi_p - \arctan\left(\frac{y_b^p}{\Delta(\mathbf{p}_b^p)} \right), \end{aligned} \tag{4.22}$$

where $U_{\mathrm{LOS}} > 0$ is the desired path-following speed.

The three tasks are then combined using the recursive NSB algorithm (2.79). If the COLAV task is active, the NSB velocity is given by

$$\mathbf{v}_{\mathrm{NSB}} = \mathbf{v}_1 + \mathbf{N}_1 \left(\mathbf{v}_2 + \mathbf{N}_2 \mathbf{v}_3 \right). \tag{4.23}$$

If the COLAV task is inactive, (4.23) is simplified to

$$\mathbf{v}_{\mathrm{NSB}} = \mathbf{v}_2 + \mathbf{v}_3, \tag{4.24}$$

thanks to the independence and orthogonality of the formation-keeping and path-following tasks.

Let $\mathbf{v}_{\mathrm{NSB},i}$ be the desired NSB velocity associated with vehicle i, *i.e.*,

$$\left[\mathbf{v}_{\mathrm{NSB},1}^{\mathrm{T}}, \ldots, \mathbf{v}_{\mathrm{NSB},N}^{\mathrm{T}} \right] = \mathbf{v}_{\mathrm{NSB}}^{\mathrm{T}}. \tag{4.25}$$

These velocities must be decomposed into surge, pitch, and yaw references that can be tracked by the attitude control system presented in Section 4.2.1. Similarly to [33], we propose a method with angle of attack and sideslip compensation

$$u_{d,i} = U_{\mathrm{NSB},i} \, \frac{1 + \cos(\gamma_{\mathrm{NSB},i} - \gamma_i) \cos(\chi_{\mathrm{NSB},i} - \chi_i)}{2}, \tag{4.26}$$

$$\theta_{d,i} = \gamma_{\mathrm{NSB},i} + \alpha_{d,i}, \quad \alpha_{d,i} = \arctan\left(\frac{w_i}{u_{d,i}} \right), \tag{4.27}$$

$$\psi_{d,i} = \chi_{\mathrm{NSB},i} - \beta_{d,i}, \quad \beta_{d,i} = \arcsin\left(\frac{v_i}{\sqrt{u_{d,i}^2 + v_i^2 + w_i^2}} \right), \tag{4.28}$$

where v_i and w_i are the sway and heave velocities, and γ_i and χ_i are the flight-path and course angles of the i^{th} vehicle, respectively, and $U_{\text{NSB},i}$, $\gamma_{\text{NSB},i}$ and $\chi_{\text{NSB},i}$ are given by

$$U_{\text{NSB},i} = \|\mathbf{v}_{\text{NSB},i}\|\,, \quad \mathbf{v}_{\text{NSB},i} = \begin{bmatrix} \dot{x}_{\text{NSB},i} \\ \dot{y}_{\text{NSB},i} \\ \dot{z}_{\text{NSB},i} \end{bmatrix}, \tag{4.29a}$$

$$\gamma_{\text{NSB},i} = -\arcsin\left(\frac{\dot{y}_{\text{NSB},i}}{U_{\text{NSB},i}}\right), \tag{4.29b}$$

$$\chi_{\text{NSB},i} = \arctan_2\left(\dot{y}_{\text{NSB},i}, \dot{x}_{\text{NSB},i}\right). \tag{4.29c}$$

4.2.3 Path Parameter Update Law

Inspired by [109], we use the update law of the path variable s to get desirable behavior of the along-track error (x_b^p).

Note that the kinematics of the i^{th} vehicle can be alternatively expressed using the total speed (U_i) and the flight-path (γ_i) and course (χ_i) angles of the vehicle as

$$\dot{\mathbf{p}}_i = \left[\cos(\chi_i)\cos(\gamma_i)\,, \cos(\gamma_i)\sin(\chi_i)\,, -\sin(\gamma_i)\right]^{\mathrm{T}} U_i. \tag{4.30}$$

Now, let us investigate the kinematics of the barycenter. Differentiating (4.3) with respect to time and substituting (4.30) yields the following equations

$$\begin{aligned} \dot{x}_b^p = \frac{1}{N}\sum_{i=1}^{N} U_i\,\Omega_x\left(\gamma_i, \theta_p, \chi_i, \psi_p\right) \\ - \left\|\frac{\partial \mathbf{p}_p(s)}{\partial s}\right\| \dot{s} + \omega_z y_b^p - \omega_y z_b^p, \end{aligned} \tag{4.31a}$$

$$\dot{y}_b^p = \frac{1}{N}\sum_{i=1}^{N} U_i\,\Omega_y\left(\gamma_i, \theta_p, \chi_i, \psi_p\right) + \omega_x z_b^p - \omega_z x_b^p, \tag{4.31b}$$

$$\dot{z}_b^p = \frac{1}{N}\sum_{i=1}^{N} U_i\,\Omega_z\left(\gamma_i, \theta_p, \chi_i, \psi_p\right) + \omega_y x_b^p - \omega_x y_b^p, \tag{4.31c}$$

where

$$\Omega_x(\cdot) = \sin(\theta_p)\sin(\gamma_i) + \cos(\theta_p)\cos(\gamma_i)\cos(\psi_p - \chi_i)\,, \tag{4.32a}$$

$$\Omega_y(\cdot) = -\cos(\gamma_i)\sin(\psi_p - \chi_i)\,, \tag{4.32b}$$

$$\Omega_z(\cdot) = -\cos(\theta_p)\sin(\gamma_i) + \cos(\gamma_i)\sin(\theta_p)\cos(\psi_p - \chi_i) \tag{4.32c}$$

$$\omega_x = -\iota\dot{s}\sin(\theta_p), \qquad \omega_y = \kappa\dot{s}, \qquad \omega_z = \iota\dot{s}\cos(\theta_p), \tag{4.32d}$$

$$\kappa(s) = \frac{\partial\theta_p(s)}{\partial s}, \qquad \iota(s) = \frac{\partial\psi_p(s)}{\partial s}. \tag{4.32e}$$

To stabilize the along-track error dynamics, we choose the following path variable update

$$\dot{s} = \left\|\frac{\partial \mathbf{p}_p(s)}{\partial s}\right\|^{-1} \left(\frac{1}{N}\sum_{i=1}^{N} U_i\,\Omega_x\left(\gamma_i, \theta_p, \chi_i, \psi_p\right) + k_s\,\frac{x_b^p}{\sqrt{1 + \left(x_b^p\right)^2}}\right), \tag{4.33}$$

where $k_s > 0$ is a constant.

4.3 Closed-Loop Analysis

In this section, we investigate the closed-loop stability of the path following task. We define two error states, $\tilde{\mathbf{X}}_1$ and $\tilde{\mathbf{X}}_2$, as

$$\tilde{\mathbf{X}}_1 = [x_b^p, y_b^p, z_b^p]^{\mathrm{T}}, \tag{4.34}$$

$$\tilde{\mathbf{X}}_2 = \left[\tilde{\mathbf{X}}_{2,1}^{\mathrm{T}}, \ldots, \tilde{\mathbf{X}}_{2,N}^{\mathrm{T}}\right]^{\mathrm{T}}, \qquad \tilde{\mathbf{X}}_{2,i} = \left[\tilde{u}_i, s_{q,i}, \tilde{\theta}_i, s_{r,i}, \tilde{\psi}_i\right]^{\mathrm{T}}, \tag{4.35}$$

Now, we can take the barycenter kinematics from (4.31) and express it in terms of $\tilde{\mathbf{X}}_1$ and $\tilde{\mathbf{X}}_2$ as

$$\dot{x}_b^p = -k_s \frac{x_b^p}{\sqrt{1 + (x_b^p)^2}} + \omega_z y_b^p - \omega_y z_b^p, \tag{4.36a}$$

$$\begin{aligned}\dot{y}_b^p = &-\frac{1}{N}\sum_{i=1}^{N} U_{d,i} \frac{\cos(\gamma_{\mathrm{LOS}})\, y_b^p}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2}} + \omega_x z_b^p - \omega_z x_b^p \\ &+ G_y\big(\tilde{u}_1, \ldots, \tilde{u}_N, \tilde{\psi}_1, \ldots, \tilde{\psi}_N, \gamma_1, \ldots, \gamma_N, \\ &\qquad u_{d,1}, \ldots, u_{d,N}, v_1, \ldots, v_N, w_1, \ldots, w_N, \mathbf{p}_b^p, \psi_p\big),\end{aligned} \tag{4.36b}$$

$$\begin{aligned}\dot{z}_b^p = &\frac{1}{N}\sum_{i=1}^{N} U_{d,i} \frac{z_b^p}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (z_b^p)^2}} + \omega_y x_b^p - \omega_x y_b^p \\ &+ G_z\big(\tilde{u}_1, \ldots, \tilde{u}_N, \tilde{\theta}_1, \ldots, \tilde{\theta}_N, \gamma_1, \ldots, \gamma_N, \chi_1, \ldots, \chi_N, \\ &\qquad u_{d,1}, \ldots, u_{d,N}, v_1, \ldots, v_N, w_1, \ldots, w_N, \mathbf{p}_b^p, \psi_p, \theta_p\big).\end{aligned} \tag{4.36c}$$

The equations for $G_y(\cdot)$ and $G_z(\cdot)$ are given in Appendix B.1. Substituting the attitude control system (4.6)–(4.13) into vehicle dynamics (4.2) yields the following closed-loop behavior of $\tilde{\mathbf{X}}_2$

$$\dot{\tilde{u}}_i = -k_u\, \tilde{u}_i - k_c \operatorname{sign}(\tilde{u}_i) - \boldsymbol{\phi}_u(\cdot)^{\mathrm{T}} \tilde{\mathbf{v}}_{c,i}, \tag{4.37a}$$

$$\dot{s}_{q,i} = -k_\theta\, \tilde{\theta}_i - k_q\, s_{q,i} - k_d \operatorname{sign}(s_{q,i}) - \boldsymbol{\phi}_q(\cdot)^{\mathrm{T}} \tilde{\mathbb{V}}_{q,i}, \tag{4.37b}$$

$$\dot{\tilde{\theta}}_i = s_{q,i} - \lambda_q\, \tilde{\theta}_i, \tag{4.37c}$$

$$\dot{s}_{r,i} = -k_\theta\, \tilde{\theta}_i - k_r\, s_{r,i} - k_d \operatorname{sign}(s_{r,i}) - \boldsymbol{\phi}_r(\cdot)^{\mathrm{T}} \tilde{\mathbb{V}}_{r,i}, \tag{4.37d}$$

$$\dot{\tilde{\psi}}_i = s_{r,i} - \lambda_r\, \tilde{\psi}_i, \tag{4.37e}$$

the ocean current estimate errors

$$\dot{\tilde{\mathbf{v}}}_{c,i} = c_u\, \boldsymbol{\phi}_u(\cdot)\, \tilde{u}_i, \tag{4.38a}$$

$$\dot{\tilde{\mathbb{V}}}_{q,i} = c_q\, \boldsymbol{\phi}_q(\cdot)\, s_{q,i}, \tag{4.38b}$$

$$\dot{\tilde{\mathbb{V}}}_{r,i} = c_r\, \boldsymbol{\phi}_r(\cdot)\, s_{r,i}, \tag{4.38c}$$

and the underactuated sway and heave dynamics

$$\dot{v}_i = X_v(u_i, u_c)\, r_i + Y_v(u_i, u_c)\,(v_i - v_c), \tag{4.39}$$

$$\dot{w}_i = X_w(u_i, u_c)\, q_i + Y_w(u_i, u_c)\,(w_i - w_c) + G(\theta_i). \tag{4.40}$$

To prove the stability of the closed-loop system, we need the results of the three following lemmas. The lemmas follow the same structure as the 2D case for two ASVs in [35], and are extended to handle an arbitrary number of AUVs moving in 3D.

Lemma 4.1. *The trajectories of the closed-loop system* (4.36)–(4.40) *are forward complete.*

Proof. The complete proof is given in Appendix B.3. Here, we only present a sketch of the proof.

The proof is split into three parts: proving the forward-completeness of the attitude control system (4.37), (4.38), the underactuated dynamics (4.39), (4.40), and the path-following errors (4.36).

Using the same arguments as for the horizontal case in [71], we can prove that the system (4.37) is GES and the ocean current estimates (4.38) are bounded. Exponential stability and boundedness imply forward completeness. Therefore, (4.37) and (4.38) are forward complete.

For the underactuated dynamics, we define Lyapunov function candidates

$$V_v(v_i) = \frac{1}{2} v_i^2, \qquad V_w(w_i) = \frac{1}{2} w_i^2, \tag{4.41}$$

and show that there exist positive constants $\alpha_v, \alpha_w, \beta_v, \beta_w$ such that

$$\dot{V}_v(v_i) \leq \alpha_v V_v(v_i) + \beta_v, \qquad \dot{V}_w(w_i) \leq \alpha_w V_w(w_i) + \beta_w. \tag{4.42}$$

Using the comparison lemma, we conclude that v_i and w_i are forward-complete.

For the path-following errors, we define a Lyapunov function candidate

$$V_b\left(\mathbf{p}_b^p\right) = \frac{1}{2}\left(\left(x_b^p\right)^2 + \left(y_b^p\right)^2 + \left(z_b^p\right)^2\right), \tag{4.43}$$

and show that there exists a class-$\mathcal{K}_\infty$ function ζ_p such that

$$\dot{V}_p\left(\mathbf{p}_b^p\right) \leq V_p\left(\mathbf{p}_b^p\right) + \zeta_p\left(v_i, w_i, \tilde{\mathbf{X}}_2\right). \tag{4.44}$$

Since all the arguments of $\zeta_p(\cdot)$ are forward complete, Corollary 2.11 of [110] is satisfied, and the barycenter dynamics is forward complete, concluding the proof of Lemma 4.1. □

Lemma 4.2. *The underactuated sway and heave dynamics are bounded near the manifold* $\left[\tilde{\mathbf{X}}_1^{\mathrm{T}}, \tilde{\mathbf{X}}_2^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$ *if* $Y_v(u, u_c) < 0$, $Y_w(u, u_c) < 0$ *and the curvature of the path satisfies*

$$|\kappa(s)| < \frac{N}{2}\left|\frac{Y_w(u, u_c)}{X_w(u, u_c)}\right|, \qquad |\iota(s)| < \frac{N}{2}\left|\frac{Y_v(u, u_c)}{X_v(u, u_c)}\right|, \tag{4.45}$$

for all $u > 0$ *and* $u_c \in [-\|\mathbf{V}_c\|, \|\mathbf{V}_c\|]$.

Proof. The complete proof is given in Appendix B.4. Here, we only present a sketch of the proof.

Consider the derivatives of the Lyapunov function candidates V_v, V_w from (4.41). Substituting $\tilde{\mathbf{X}}_1 = \mathbf{0}, \tilde{\mathbf{X}}_2 = \mathbf{0}$, we get the following inequalities

$$\dot{V}_v(v_i) \leq \left(X_v\left(u_{d,i}, u_c\right) \frac{2}{N} \left|\iota(\xi)\right| + Y_v\left(u_{d,i}, u_c\right) \right) v_i^2 + F_v(v_i), \tag{4.46}$$

$$\dot{V}_w(w_i) \leq \left(X_w\left(u_{d,i}, u_c\right) \frac{2}{N} \left|\kappa(\xi)\right| + Y_w\left(u_{d,i}, u_c\right) \right) w_i^2 + F_w(w_i), \tag{4.47}$$

where F_v and F_w grow at most linearly with v_i and w_i, respectively. Then, we conclude that for a sufficiently large v_i and w_i, the quadratic terms will dominate the linear terms. Therefore, the underactuated dynamics are bounded if the quadratic terms are negative, which is equivalent to condition (4.45). □

Lemma 4.3. *The underactuated sway and heave dynamics are bounded near the manifold $\tilde{\mathbf{X}}_2 = \mathbf{0}$, independently of $\tilde{\mathbf{X}}_1$ if the assumptions in Lemma 4.2 are satisfied and the constant term Δ_0 in the lookahead distance* (4.20) *is chosen so that*

$$\Delta_0 > \max\left\{ \frac{3}{N\left|\frac{Y_v(u,u_c)}{X_v(u,u_c)}\right| - 2\left|\iota(s)\right|}, \frac{3}{N\left|\frac{Y_w(u,u_c)}{X_w(u,u_c)}\right| - 2\left|\kappa(s)\right|} \right\}, \tag{4.48}$$

for all $u > 0$ and $u_c \in [-\left\|\mathbf{V}_c\right\|, \left\|\mathbf{V}_c\right\|]$.

Proof. The complete proof is given in Appendix B.5. Here, we only present a sketch of the proof.

Once again, we consider the derivatives of the Lyapunov function candidates V_v, V_w from (4.41). Substituting $\tilde{\mathbf{X}}_2 = \mathbf{0}$, we get the following inequalities

$$\dot{V}_v(v_i) \leq \left(X_v(u_{d,i}, u_c) \left(\frac{2}{N} \left|\iota(\xi)\right| + \frac{3}{N\,\Delta(\mathbf{p}_b^p)} \right) + Y_v\left(u_{d,i}, u_c\right) \right) v_i^2 + F_v(v_i), \tag{4.49}$$

$$\dot{V}_w(w_i) \leq \left(X_w(u_{d,i}, u_c) \left(\frac{2}{N} \left|\kappa(\xi)\right| + \frac{3}{N\,\Delta(\mathbf{p}_b^p)} \right) + Y_w\left(u_{d,i}, u_c\right) \right) w_i^2 + F_w(w_i), \tag{4.50}$$

where F_v and F_w grow at most linearly with v_i and w_i, respectively. Using the same arguments as in the proof of Lemma 4.2, we conclude that the underactuated dynamics are bounded if both (4.45) and (4.48) hold. □

Theorem 4.1. *The origin $\left[\tilde{\mathbf{X}}_1^{\mathrm{T}}, \tilde{\mathbf{X}}_2^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$ of the system described by* (4.36), (4.37) *is a USGES equilibrium point if the conditions of Lemmas 4.2 and 4.3 hold and the maximum pitch angle of the path satisfies*

$$\theta_{p,\max} = \max_{s \in \mathbb{R}} \left|\theta_p(s)\right| < \frac{\pi}{4}. \tag{4.51}$$

Moreover, the ocean current estimate errors (4.38) *and the underactuated sway and heave dynamics* (4.39), (4.40) *are bounded.*

Remark. *Condition* (4.51) *is needed to ensure that* $|\gamma_{\text{LOS}}| < \pi/2$. *Indeed, from* (4.22)*, the largest possible LOS reference angle is*

$$\begin{aligned}\gamma_{\text{LOS,max}} &= \theta_{p,\max} + \lim_{z_b^p \to \infty} \arctan\left(\frac{z_b^p}{\sqrt{\Delta_0^2 + \left(z_b^p\right)^2}}\right) \\ &= \theta_{p,\max} + \frac{\pi}{4}.\end{aligned} \tag{4.52}$$

With (4.51) *satisfied, the cosine of* γ_{LOS} *is always positive. We will use this fact in the proof.*

Proof. The proof follows along the lines of [35], but is extended to an arbitrary number of 5DOF vehicles. We will also use the results of [70] to prove that the system is USGES.

In Lemmas 4.1–4.3, we have shown that the closed-loop system is forward complete and the underactuated sway and heave dynamics are bounded near the manifold $\tilde{\mathbf{X}}_2 = \mathbf{0}$. Since (4.37) is UGES [71], we can conclude that there exists a finite time $T > t_0$ such that the solutions of (4.37) will be sufficiently close to $\tilde{\mathbf{X}}_2 = \mathbf{0}$ to guarantee boundedness of v_i and w_i. Having established that the underactuated dynamics are bounded, we will now utilize cascaded theory to analyze the cascade (4.36), (4.37), where (4.37) perturbs the nominal dynamics (4.36) through the terms $G_y(\cdot)$ and $G_z(\cdot)$.

Now, consider the nominal dynamics of $\tilde{\mathbf{X}}_1$ (*i.e.,* (4.36) without the perturbing terms G_y and G_z), and a Lyapunov function candidate

$$V(\tilde{\mathbf{X}}_1) = \frac{1}{2}\tilde{\mathbf{X}}_1^{\mathrm{T}}\,\tilde{\mathbf{X}}_1 = \frac{1}{2}\left((x_b^p)^2 + (y_b^p)^2 + (z_b^p)^2\right), \tag{4.53}$$

whose derivative along the trajectories of (4.36) is

$$\dot{V}(\tilde{\mathbf{X}}_1) = -\tilde{\mathbf{X}}_1^{\mathrm{T}}\,\mathbf{Q}\,\tilde{\mathbf{X}}_1, \qquad \mathbf{Q} = \operatorname{diag}(q_1, q_2, q_3), \tag{4.54a}$$

$$q_1 = \frac{k_s}{\sqrt{1 + \left(x_b^p\right)^2}}, \qquad q_2 = \frac{\frac{1}{N}\sum_{i=1}^{N} U_{d,i}\cos\left(\gamma_{\text{LOS}}\right)}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(y_b^p\right)^2}}, \tag{4.54b}$$

$$q_3 = \frac{\frac{1}{N}\sum_{i=1}^{N} U_{d,i}}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(z_b^p\right)^2}}. \tag{4.54c}$$

Note that $\mathbf{Q}$ is positive definite, and the nominal system is thus UGAS. Furthermore, note that the following inequality

$$\dot{V}(\tilde{\mathbf{X}}_1) \leq -q_{\min}\left\|\tilde{\mathbf{X}}_1\right\|^2, \tag{4.55a}$$

$$q_{\min} = \min\left\{\frac{k_s}{\sqrt{1+r^2}}, \frac{\frac{1}{N}\sum_{i=1}^{N} U_{d,i}\cos\left(\gamma_{\text{LOS}}\right)}{\sqrt{\Delta_0^2 + 4r^2}}\right\}, \tag{4.55b}$$

holds $\forall \tilde{\mathbf{X}}_1 \in \mathcal{B}_r$. Thus, the conditions of [70, Theorem 5] are fulfilled with $k_1 = k_2 = 1/2$, $a = 2$, and $k_3 = q_{\min}$, and the nominal system is USGES.

As discussed in the proof of Lemma 4.1, the perturbing system (4.37) is UGES, implying both UGAS and USGES. Furthermore, it is straightforward to show that the following holds for the Lyapunov function (4.53)

$$\left\| \frac{\partial V}{\partial \tilde{\mathbf{X}}_1} \right\| \left\| \tilde{\mathbf{X}}_1 \right\| = \left\| \tilde{\mathbf{X}}_1 \right\|^2 = 2\, V\left(\tilde{\mathbf{X}}_1 \right), \qquad \forall \tilde{\mathbf{X}}_1, \tag{4.56}$$

$$\left\| \frac{\partial V}{\partial \tilde{\mathbf{X}}_1} \right\| = \left\| \tilde{\mathbf{X}}_1 \right\| \leq \mu, \qquad \forall \left\| \tilde{\mathbf{X}}_1 \right\| \leq \mu. \tag{4.57}$$

Therefore, [70, Assumption 1] is satisfied with $c_1 = 2$ and $c_2 = \mu$ for any $\mu > 0$.

Finally, [70, Assumption 2] must be investigated. From (B.12), (B.22), it can be shown that for both perturbing terms there exist positive functions $\zeta_{y,1}(\cdot)$, $\zeta_{y,2}(\cdot)$, $\zeta_{z,1}(\cdot)$, $\zeta_{z,2}(\cdot)$, such that

$$|G_y(\cdot)| \leq \zeta_{y,1}\left(\left\| \tilde{\mathbf{X}}_2 \right\| \right) + \zeta_{y,2}\left(\left\| \tilde{\mathbf{X}}_2 \right\| \right) \left\| \tilde{\mathbf{X}}_1 \right\|, \tag{4.58}$$

$$|G_z(\cdot)| \leq \zeta_{z,1}\left(\left\| \tilde{\mathbf{X}}_2 \right\| \right) + \zeta_{z,2}\left(\left\| \tilde{\mathbf{X}}_2 \right\| \right) \left\| \tilde{\mathbf{X}}_1 \right\|. \tag{4.59}$$

Therefore, all conditions of [70, Proposition 9] are satisfied, and the closed-loop system is USGES. □

4.4 Simulation Results

In this section, we present the results of a numerical simulation of three light autonomous underwater vehicles (LAUVs) [12]. The parameters of the simulation are summarized in Table 4.1. The barycenter should follow a spiral path given by

$$\mathbf{p}_p(s) = \left[s, a\, \cos(\omega\, s), b\, \sin(\omega\, s) \right]^{\mathrm{T}}. \tag{4.60}$$

The maximum curvature of this path is

$$\max_{s \in \mathbb{R}} |\kappa(s)| = \frac{b\, \omega^2}{\sqrt{a^2\, \omega^2 + 1}}, \qquad \max_{s \in \mathbb{R}} |\iota(s)| = a\, \omega^2, \tag{4.61}$$

while the smallest absolute values of Y_v/X_v and Y_w/X_w for the LAUV model are approximately 0.26. Consequently, the path is chosen such that the maximum curvature is

$$\max_{s \in \mathbb{R}} |\kappa(s)| = 0.013, \qquad \max_{s \in \mathbb{R}} |\iota(s)| = 0.040, \tag{4.62}$$

and (4.45) is satisfied. From (4.48), the lookahead distance must then satisfy $\Delta_0 > 4.29$. We choose $\Delta_0 = 5$, since smaller distances guarantee faster convergence.

The very minimum relative distance to avoid collision is the length of the LAUV, *i.e.* 2.4 m. For additional safety, we design the COLAV task with $d_{\min} = 5$ m. To add a security zone during transients, d_{COLAV} is chosen to be 10 m.

The desired formation is an isosceles triangle parallel to the yz plane. Specifically, the desired positions of the three vehicles are

$$\mathbf{p}^f_{f,1} = \begin{bmatrix} 0 \\ 10 \\ 5 \end{bmatrix}, \qquad \mathbf{p}^f_{f,2} = \begin{bmatrix} 0 \\ -10 \\ 5 \end{bmatrix}, \qquad \mathbf{p}^f_{f,3} = \begin{bmatrix} 0 \\ 0 \\ -10 \end{bmatrix}. \tag{4.63}$$

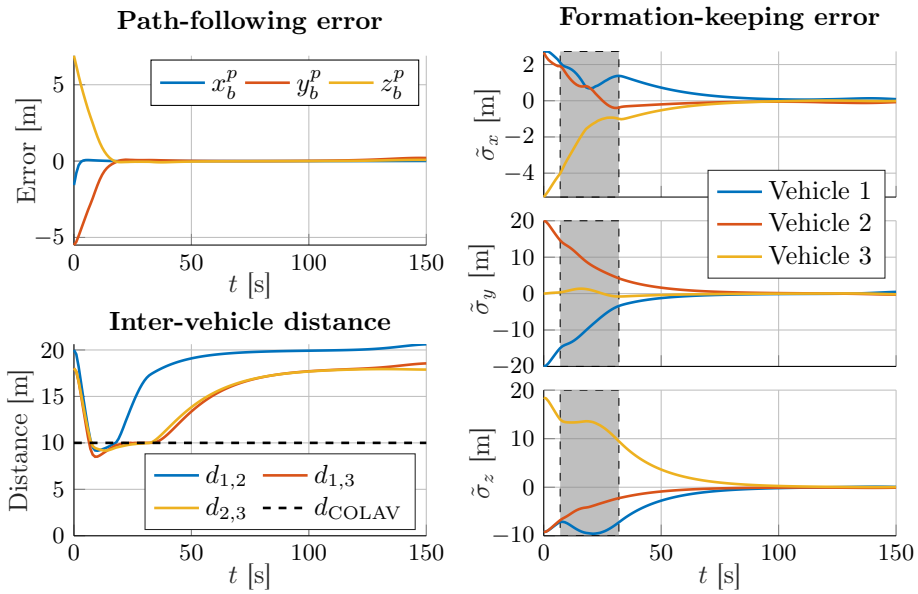


Figure 4.1: Simulation results. The top-left plot shows the x-, y- and z-components of the path-following error $\mathbf{p}_b^p$, as defined in (4.3). The bottom-left plot shows the distance between the vehicles ($d_{i,j} = \|\mathbf{p}_i - \mathbf{p}_j\|$). The plots on the right show the x-, y- and z-components of the formation-keeping error $\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}_2 - \boldsymbol{\sigma}_{d,2}$ with $\boldsymbol{\sigma}_2$ given by (4.17) and $\boldsymbol{\sigma}_{d,2}$ given by (4.18). The grey rectangles mark the intervals when the COLAV task is active.

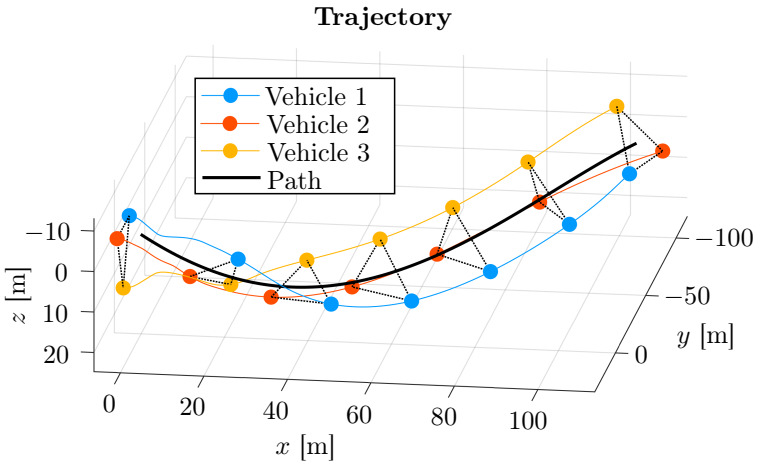


Figure 4.2: 3D trajectory of the vehicles. The markers represent the position of the vehicles at times $t = 0, 25, 50, \ldots, 150$ seconds. Markers with corresponding times are connected by dotted lines to better illustrate the resulting formation.

Parameter	Value	Parameter	Value
k_u	0.05	$\mathbf{V}_c$	$[0, 0.25, 0.05]^{\mathrm{T}}$
k_c	0.1	Δ_0	5
k_θ, k_ψ	0.0625	d_{COLAV}	10
k_q, k_r	0.25	U_{LOS}	1
k_d	0.1	k_s	1
λ_q, λ_r	0.75	$\mathbf{p}_0$	$\mathbf{0}_3$
c_u	5	a	40
c_q, c_r	1	b	20
$\mathbf{\Lambda}_1$	$\mathbf{I}$	ω	$\pi/100$
$\mathbf{\Lambda}_2$	$0.05\,\mathbf{I}$		

Table 4.1: Simulation parameters

The gains of the low-level control systems (4.6),(4.9),(4.12) are chosen such that the settling time is approximately 10 seconds. The gains of the pitch and yaw PD controllers are chosen such that the closed-loop system is critically damped.

The results of the numerical simulation are shown in Figures 4.1 and 4.2. The vehicles start in an inverted triangular formation. The COLAV task is briefly activated, and the distance between the vehicles drops to approximately 8 meters during the transient. Eventually, the vehicles resolve the situation and continue to converge to the desired path and formation.

Note that while the COLAV task is active, the formation-keeping error is diverging. After resolving the situation, the formation-keeping error converges to zero exponentially. The rate of convergence is given by the formation-keeping gain $\mathbf{\Lambda}_2$.

The path-following error seems to converge linearly at first, and then exponentially as the error gets smaller. This phenomenon is caused by the LOS guidance law (4.22), *cf.* [104], and the path parameter update law (4.33). The inverse tan in (4.22) and the last term in (4.33) act as a saturation, slowing the convergence for large errors. The rate of convergence of the along-track error (x_b^p) is given by the path parameter update gain k_s, while the rate of convergence of the cross-track errors (y_b^p, z_b^p) is given by the lookahead distance Δ_0. The path-following error seems to increase at $t = 150\,\mathrm{s}$. This increase is probably caused by low-level tracking errors. To avoid chattering, the sign functions in the low-level sliding-mode controllers (4.6)–(4.12) are approximated using hyperbolic tan. These approximations result in a non-zero steady-state error.

Chapter 5

Formation Path-Following Control of 6DOF Underactuated AUVs

This chapter presents a method for formation path-following control of a fleet of underactuated autonomous underwater vehicles. The proposed method combines several hierarchic tasks in a null-space-behavioral algorithm to safely guide the vehicles. Compared to the previous chapter, the algorithm includes both inter-vehicle and obstacle collision avoidance, and employs a scheme that keeps the vehicles within given operation limits. The algorithm is applied to a six degree-of-freedom model, using rotation matrices to describe the attitude to avoid singularities. Using the results of cascaded systems theory, we prove that the closed-loop system is uniformly semiglobally exponentially stable. We use numerical simulations to validate the results. The contents of this chapter are based on [79].

The chapter is organized as follows. Section 5.1 introduces the model of the AUVs and defines the formation path-following problem. Section 5.2 describes the proposed modified NSB algorithm. The stability of the closed-loop system is proven in Section 5.3. Finally, Section 5.4 presents the results of the numerical simulations.

5.1 Problem Definition

In this section, we briefly present the AUV model and the formation path-following problem.

5.1.1 Vehicle Model

We consider a fleet of N underactuated AUVs. The dynamics of the AUVs are described using the 6DOF control-oriented model from Section 2.1.4. Let $\mathbf{p} = [x, y, z]^{\mathrm{T}}$ be the position, and let $\boldsymbol{\upsilon} = [u, v, w]^{\mathrm{T}}$ and $\boldsymbol{\omega} = [p, q, r]^{\mathrm{T}}$ be the linear and angular velocities, respectively. To avoid the singularities caused by the use of Euler angles, the orientation of the AUV is given by a rotation matrix $\mathbf{R} \in SO(3)$. Furthermore, let $\mathbf{V}_c \in \mathbb{R}^3$ be the velocities of an unknown, constant and irrotational ocean current.

Recalling (2.8), the dynamics of the AUVs are

$$\dot{\mathbf{p}} = \mathbf{R}\boldsymbol{\upsilon}, \tag{5.1a}$$

$$\dot{\mathbf{R}} = \mathbf{R}\mathbf{S}(\boldsymbol{\omega}), \tag{5.1b}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}}_r + \big(\mathbf{C}(\boldsymbol{\nu}_r) + \mathbf{D}(\boldsymbol{\nu}_r)\big)\boldsymbol{\nu}_r + \mathbf{g}(\mathbf{R}) = \mathbf{B}\mathbf{f}. \tag{5.1c}$$

Unlike the previous chapter, here we do not design a specific low-level controller. The aim of this chapter is to demonstrate that the proposed formation path-following algorithm can be readily implemented on vehicles with existing low-level controllers. Consequently, the choice of a low-level velocity and attitude controller is not discussed in this chapter. However, there are some assumptions about the low-level controller.

Assumption 5.1. *The vehicle is equipped with a low-level controller that allows exponential tracking of the surge velocity, orientation, and angular velocity. Specifically, let $u_d, \mathbf{R}_d$ and $\boldsymbol{\omega}_d$ be the reference signals. We define an error*

$$\widetilde{\mathbf{X}} = \left[u - u_d, \operatorname{logm}\big(\widetilde{\mathbf{R}}\big)^{\mathrm{T}}, \Big(\boldsymbol{\omega} - \widetilde{\mathbf{R}}^{\mathrm{T}}\boldsymbol{\omega}_d\Big)^{\mathrm{T}}\right]^{\mathrm{T}}, \qquad \widetilde{\mathbf{R}} = \mathbf{R}_d^{\mathrm{T}}\mathbf{R}, \tag{5.2}$$

where $\operatorname{logm} : SO(3) \mapsto \mathfrak{B}_\pi^3$ *is the matrix logarithm [111]. Note that by Assumption 2.3,* $\widetilde{\mathbf{X}}$ *is controllable through the input* $\mathbf{f}$*. Consider the closed-loop system*

$$\dot{\widetilde{\mathbf{X}}} = F\left(\widetilde{\mathbf{X}}, v, w, \mathbf{V}_c\right), \tag{5.3}$$

consisting of (5.1b), (5.1c)*, and the low-level controller. We assume that* $\widetilde{\mathbf{X}} = \mathbf{0}$ *is a globally exponentially stable (GES) equilibrium of* (5.3)*.*

An example of a global exponential attitude tracking controller can be found, *e.g.,* in [112].

Note that for a complete system analysis, we need to consider the underactuated sway and heave dynamics explicitly. Recalling (2.22), the underactuated dynamics have the following form

$$\dot{v}_r = X_v(u_r)r + Y_v(u_r)v_r + Z_v(p)w_r, \tag{5.4a}$$

$$\dot{w}_r = X_w(u_r)q + Y_w(u_r)w_r + Z_w(p)v_r, \tag{5.4b}$$

where $X(\cdot), Y(\cdot), Z(\cdot)$ are affine functions of the respective variables.

5.1.2 Formation Path-Following Problem

Similarly to the previous chapter, the goal is to control a fleet of N AUVs so that they move in a prescribed formation and their barycenter follows a given path.

The prescribed path is parametrized by a smooth function $\mathbf{p}_p : \mathbb{R} \mapsto \mathbb{R}^3$. We assume that the function is $\mathcal{C}^\infty$ and regular. Therefore, for every point $\mathbf{p}_p(s)$ on the path, there exists a path-tangential coordinate frame and a corresponding rotation

matrix $\mathbf{R}_p(s)$ (see Section 2.2). Moreover, we assume that the curvature vector $\boldsymbol{\omega}_p(s)$ exists and is continuously differentiable.

The path-following error $\mathbf{p}_b^p$ is given by the position of the barycenter in the path-tangential coordinate frame

$$\mathbf{p}_b^p = \mathbf{R}_p^{\mathrm{T}} \left(\mathbf{p}_b - \mathbf{p}_p(s) \right), \qquad \mathbf{p}_b = \frac{1}{N} \sum_{i=1}^{N} \mathbf{p}_i. \tag{5.5}$$

The goal of path following is to control the vehicles so that $\mathbf{p}_b^p \to \mathbf{0}_3$.

The vehicles should converge to a dynamic formation that rotates with the desired path (see Section 2.3.2 for details). Let $\mathbf{p}_{f,1}^f, \ldots, \mathbf{p}_{f,N}^f$ be the position vectors that represent the desired formation. The objective is to control the vehicles so that

$$\mathbf{p}_i - \mathbf{p}_b \to \mathbf{R}_p(s) \mathbf{p}_{f,i}^f, \qquad \forall i \in \{1, \ldots, N\}. \tag{5.6}$$

5.2 The Proposed Algorithm

The AUVs must perform the goals stated in Section 5.1.2 safely, *i.e.*, avoid collisions with other vehicles and obstacles, and remain within a given range of depths. An upper limit on the depth of the AUVs is needed to prevent them from colliding with the seabed or exceeding their depth rating. A lower limit is needed in busy environments (*e.g.,* harbors), where the AUVs may otherwise collide or interfere with surface vessels.

To solve the formation path-following problem, we propose a method that combines inter-vehicle collision avoidance (COLAV), formation keeping, line-of-sight (LOS) path following, obstacle avoidance, and depth limiting in a hierarchic manner using an NSB algorithm. Similarly to Chapter 4, the proposed algorithm is centralized, meaning that to implement it in a real-life scenario, we require a central node that can communicate and coordinate with the AUVs. Since the NSB algorithm outputs inertial velocity references, we also need a method for converting these to surge and orientation.

In this section, we first present the NSB algorithm and the associated tasks. We then present in Section 5.2.6 a strategy for converting inertial velocity references to surge/orientation ones.

5.2.1 NSB Algorithm

Similarly to the previous chapter, we define three tasks: COLAV, formation-keeping, and path-following. Each task will be described in detail in Sections 5.2.2, 5.2.3, and 5.2.4. As we will explain in Section 5.2.5, obstacle avoidance and depth limiting will not be defined as separate tasks but rather achieved through a modification to the path-following task. Let us denote the variables associated with the COLAV, formation-keeping, and path-following tasks by lower indices 1, 2, and 3, respectively. Define the so-called *task variables* as $\boldsymbol{\sigma}_m = \boldsymbol{f}_m \left(\mathbf{p}_1, \ldots, \mathbf{p}_N \right), m \in \{1, 2, 3\}$, and their desired values as $\boldsymbol{\sigma}_{d,m}, m \in \{1, 2, 3\}$.

Furthermore, let $\mathbf{v}_m, m \in \{1,2,3\}$ be the desired velocities of each task. These velocities are combined using the recursive NSB algorithm (2.79).

$$\mathbf{v}_{\mathrm{NSB}} = \mathbf{v}_1 + \mathbf{N}_1 \left(\mathbf{v}_2 + \mathbf{N}_2 \mathbf{v}_3\right). \tag{5.7}$$

5.2.2 Inter-Vehicle Collision Avoidance

Let d_{COLAV} be the *activation distance*, *i.e.,* the distance at which the vehicles need to start performing the evasive maneuvers. The task variable is given by a vector of relative distances between the vehicles smaller than d_{COLAV}

$$\boldsymbol{\sigma}_1 = \left[\, \|\mathbf{p}_i - \mathbf{p}_j\| \,\right], \qquad \begin{array}{c} \forall i,j \in \{1,\ldots,N\}, j > i, \\ \|\mathbf{p}_i - \mathbf{p}_j\| < d_{\mathrm{COLAV}}. \end{array} \tag{5.8}$$

The desired values of the task are

$$\boldsymbol{\sigma}_{d,1} = d_{\mathrm{COLAV}}\, \mathbf{1}, \tag{5.9}$$

where $\mathbf{1}$ is a vector of ones. To ensure a faster response to a potential collision than in the previous chapter, we propose the following sliding-mode-like COLAV velocity

$$\mathbf{v}_1 = -U_{\mathrm{COLAV}} \mathbf{J}_1^{\dagger} \frac{\widetilde{\boldsymbol{\sigma}}_1}{\|\widetilde{\boldsymbol{\sigma}}_1\|}, \qquad \widetilde{\boldsymbol{\sigma}}_1 = \boldsymbol{\sigma}_1 - \boldsymbol{\sigma}_{d,1}, \tag{5.10}$$

where U_{COLAV} is a positive constant.

Similarly to the previous chapter, this task does not guarantee robust collision avoidance. During the transients, the relative distance may become smaller than d_{COLAV}. Therefore, to ensure collision avoidance, d_{COLAV} should be chosen as $d_{\min} + d_{\mathrm{sec}}$, where $d_{\min}$ is the minimum safe distance between the vehicles, and d_{sec} is an additional security distance.

To avoid collisions robustly, we would need to consider the dynamics of the AUVs and employ a reactive COLAV algorithm, *c.f.,* Section 1.1.3.

5.2.3 Formation Keeping

The formation-keeping task variable is defined as

$$\boldsymbol{\sigma}_2 = \left[\boldsymbol{\sigma}_{2,1}^{\mathrm{T}}, \ldots, \boldsymbol{\sigma}_{2,N-1}^{\mathrm{T}}\right]^{\mathrm{T}}, \qquad \boldsymbol{\sigma}_{2,i} = \mathbf{p}_i - \mathbf{p}_b, \tag{5.11}$$

and its desired values are

$$\boldsymbol{\sigma}_{d,2} = \left[\left(\mathbf{R}_p\, \mathbf{p}_{f,1}^{f}\right)^{\mathrm{T}}, \ldots, \left(\mathbf{R}_p\, \mathbf{p}_{f,N-1}^{f}\right)^{\mathrm{T}}\right]^{\mathrm{T}}. \tag{5.12}$$

The formation-keeping velocity is given by

$$\mathbf{v}_2 = \mathbf{J}_2^{\dagger} \dot{\boldsymbol{\sigma}}_{d,2} - v_{2,\max} \mathbf{J}_2^{\dagger}\, \mathrm{sat}\left(\boldsymbol{\Lambda}_2 \widetilde{\boldsymbol{\sigma}}_2\right), \qquad \widetilde{\boldsymbol{\sigma}}_2 = \boldsymbol{\sigma}_2 - \boldsymbol{\sigma}_{d,2}, \tag{5.13}$$

where $v_{2,\max}$ is a positive constant, and sat is a saturation function given by

$$\mathrm{sat}(\mathbf{x}) = \mathbf{x} \frac{\tanh\left(\|\mathbf{x}\|\right)}{\|\mathbf{x}\|}. \tag{5.14}$$

The use of a saturation function guarantees some bounds on the NSB velocity, as we will explain in Section 5.2.6.

5.2.4 Path Following

To solve the path-following, we employ the coupled LOS guidance algorithm (2.71). Let $\Delta\left(\mathbf{p}_b^p\right)$ be the lookahead distance of the LOS guidance law. Inspired by [109], we choose an error-dependent lookahead distance

$$\Delta\left(\mathbf{p}_b^p\right) = \sqrt{\Delta_0^2 + \left(x_b^p\right)^2 + \left(y_b^p\right)^2 + \left(z_b^p\right)^2} \tag{5.15}$$

where Δ_0 is a positive constant. The LOS velocity is then given by

$$\mathbf{v}_{\mathrm{LOS}} = \frac{U_{\mathrm{LOS}}}{D}\,\mathbf{R}_p\left[\Delta\left(\mathbf{p}_b^p\right), -y_b^p, -z_b^p\right]^{\mathrm{T}}, \tag{5.16}$$

where $U_{\mathrm{LOS}} > 0$ is the desired path-following speed, and

$$D = \sqrt{\Delta(\cdot)^2 + \left(y_b^p\right)^2 + \left(z_b^p\right)^2}. \tag{5.17}$$

The task velocity is then given by

$$\mathbf{v}_3 = \mathbf{1}_N \otimes \mathbf{v}_{\mathrm{LOS}}. \tag{5.18}$$

Note that the path parameter s in (5.5) can be treated as an additional degree of freedom in the control design, and used to get a stable behavior of the along-track error x_b^p. Inspired by [109], we choose the update law of s as

$$\dot{s} = \left\| \frac{\partial \mathbf{p}_p(s)}{\partial s} \right\|^{-1} U_{\mathrm{LOS}} \left(\frac{\Delta}{D} + k_s \frac{x_b^p}{\sqrt{1 + \left(x_b^p\right)^2}} \right), \tag{5.19}$$

where k_s is a positive gain.

5.2.5 Obstacle Avoidance and Depth Limiting

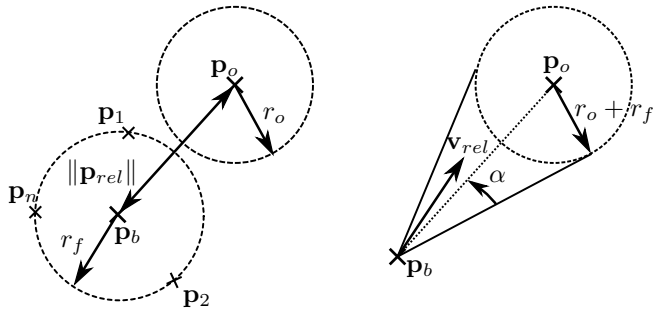
Obstacle avoidance is typically implemented individually for each vehicle [102]. However, we propose to perform this task globally by incorporating it into the path-following algorithm so that it does not interfere with the inter-vehicle COLAV.

To arrive at the proposed algorithm, we first restrict the obstacle avoidance maneuvers to the xy-plane to avoid interfering with the subsequent depth-limiting logic. Let $\mathbf{p}_o = [x_o, y_o, z_o]^{\mathrm{T}}$ be the position of the obstacle and r_o the obstacle avoidance radius. Note that r_o must be chosen sufficiently large to cover the size of both the obstacle and the AUV. Furthermore, let us define the formation radius $r_f = \max_{i\in\{1,\ldots,N\}} \left\| [x_b - x_i, y_b - y_i]^{\mathrm{T}} \right\|$ and the relative position $\mathbf{p}_{\mathrm{rel}} = [x_b - x_o, y_b - y_o]^{\mathrm{T}}$. As illustrated in Figure 5.1a, obstacle avoidance is ensured if

$$\|\mathbf{p}_{\mathrm{rel}}\| \geq r_o + r_f. \tag{5.20}$$

To guarantee obstacle avoidance, we utilize the collision cone concept [113]. Inspired by [114], we employ a constant avoidance angle and define a switching condition. More precisely, let

$$\mathbf{v}_{\mathrm{rel}} = [v_{x,\mathrm{LOS}} - \dot{x}_o, v_{y,\mathrm{LOS}} - \dot{y}_o]^{\mathrm{T}} \tag{5.21}$$



(a) Obstacle and formation radii

(b) Collision cone

Figure 5.1: Illustration motivating the obstacle avoidance constraint (5.20) and conflict condition (5.22).

denote the relative line-of-sight velocity ($v_{x,\mathrm{LOS}}$ and $v_{y,\mathrm{LOS}}$ are the components of $\mathbf{v}_{\mathrm{LOS}}$). As shown in Figure 5.1b, a conflict between the AUVs and the obstacle arises if the relative velocity lies in the so-called *collision cone*, *i.e.*, if

$$|\angle(\mathbf{p}_{\mathrm{rel}}, -\mathbf{v}_{\mathrm{rel}})| \leq \alpha, \qquad \alpha = \sin^{-1}\left(\frac{r_o + r_f}{\|\mathbf{p}_{\mathrm{rel}}\|}\right), \tag{5.22}$$

where $\angle(\mathbf{a}, \mathbf{b})$ denotes the angle between two vectors.

The obstacle avoidance task is activated if simultaneously such a conflict arises and the cone angle satisfies $\alpha \geq \alpha_{\min}$, where $0 < \alpha_{\min} \ll \pi/2$. Note that [114] use a switching condition based on distance, *i.e.,* $\|\mathbf{p}_{\mathrm{rel}}\| \leq d_{\min}$. Since our definition of a safe distance (5.20) is not constant, we instead suggest using a switching rule based on the cone angle.

When the task is active, the x- and y-components of the LOS velocity are replaced by the obstacle avoidance velocity $\mathbf{v}_{\mathrm{OA}}$ given by

$$\mathbf{v}_{\mathrm{OA}} = \|\mathbf{v}_{\mathrm{rel}}\| \left[\cos(\psi_{\mathrm{OA}}), \sin(\psi_{\mathrm{OA}})\right]^{\mathrm{T}} + \left[\dot{x}_o, \dot{y}_o\right]^{\mathrm{T}}, \tag{5.23}$$

$$\psi_{\mathrm{OA}} = \mathrm{atan}_2\left(y_o - y_b, x_o - x_b\right) \pm \alpha. \tag{5.24}$$

Note that ψ_{OA} has two solutions corresponding to the clockwise and counterclockwise directions. Inspired by [115], we propose the following method for choosing a direction: When the conflict first happens, we choose the value of ψ_{OA} that is closer to the direction of $\mathbf{v}_{\mathrm{rel}}$. Afterwards, we maintain the same direction.

As for the depth-limiting logic, let $z_{\min}$ and $z_{\max}$ be the operation limits. We assume the limits to be wide enough to accommodate the formation. We then propose to replace the z-component of the LOS velocity with a depth-limiting velocity $v_{z,\mathrm{lim}}$ given by

$$v_{z,\mathrm{lim}} = \begin{cases} v_z, & \text{if } \min_{i\in\{1,\ldots,N\}} z_i \leq z_{\min}, \\ -v_z, & \text{if } \max_{i\in\{1,\ldots,N\}} z_i \geq z_{\max}, \\ v_{z,\mathrm{LOS}}, & \text{otherwise,} \end{cases} \tag{5.25}$$

where v_z is a positive constant.

5.2.6 Surge and Orientation References

Since the NSB algorithm outputs inertial velocity references, we also need a method for converting these to surge and orientation references. The strategy for choosing these references changes depending on whether the avoidance or depth-limiting tasks are active. The proposed strategy allows us to prove the closed-loop stability of both the path-following and formation-keeping tasks (*c.f.* [33], where no stability proofs are given, and [35, 83], that only prove the stability of the path-following task).

First, let us consider the case when neither the avoidance nor depth-limiting tasks are active. Because the formation-keeping and the path-following task are independent and orthogonal, (5.7) can be simplified to

$$\mathbf{v}_{\text{NSB}} = \mathbf{v}_2 + \mathbf{v}_3. \tag{5.26}$$

Let $\mathbf{v}_{\text{NSB},i}$ denote the desired velocity of vehicle i. To achieve the desired behavior, the surge reference $u_{d,i}$ should be chosen such that

$$\left\| [u_{d,i}, v_i, w_i]^{\text{T}} \right\| = \left\| \mathbf{v}_{\text{NSB},i} \right\|, \tag{5.27}$$

However, since we cannot directly control the sway and heave velocities, (5.27) can only be satisfied if

$$\left\| \mathbf{v}_{\text{NSB},i} \right\|^2 \geq v_i^2 + w_i^2. \tag{5.28}$$

In addition, AUVs typically need to maintain a minimum surge velocity to be able to maneuver, implying a stricter inequality

$$\left\| \mathbf{v}_{\text{NSB},i} \right\|^2 \geq u_{\min}^2 + v_i^2 + w_i^2 \tag{5.29}$$

where $u_{\min} > 0$. We will show that this inequality can be satisfied by choosing a time-varying path-following speed U_{LOS}.

Substituting task velocity definitions (5.13) and (5.18) into (5.26) and exploiting the structure of the task Jacobian $\mathbf{J}_2$, we get that the NSB velocity of vehicle i is given by

$$\mathbf{v}_{\text{NSB},i} = \mathbf{v}_{\text{LOS}} + \dot{\mathbf{R}}_p(s)\mathbf{p}_{f,i}^f + \mathbf{v}_{2,i}, \tag{5.30}$$

where

$$\left[\mathbf{v}_{2,1}^{\text{T}}, \ldots, \mathbf{v}_{2,N}^{\text{T}}\right]^{\text{T}} = -v_{2,\max} \,\text{sat}\left(\mathbf{J}_2^{\dagger} \boldsymbol{\Lambda}_2 \widetilde{\boldsymbol{\sigma}}_2\right). \tag{5.31}$$

From (2.51), the time-derivative of $\mathbf{R}_p(s)$ is given by

$$\dot{\mathbf{R}}_p(s) = \mathbf{R}_p(s)\, \mathbf{S}\big(\boldsymbol{\omega}_p(s)\big)\, \dot{s}. \tag{5.32}$$

From (5.19), we get the following upper bound on $\dot{s}$

$$|\dot{s}| \leq \left\| \frac{\partial \mathbf{p}_p(s)}{\partial s} \right\|^{-1} U_{\text{LOS}} \left(1 + k_s\right). \tag{5.33}$$

Substituting (5.31), (5.32), and (5.33) into (5.30), we get the following lower bound on the NSB velocity

$$\|\mathbf{v}_{\mathrm{NSB},i}\| \geq U_{\mathrm{LOS}} \left(1 - \|\boldsymbol{\omega}_p\| \left\| \tfrac{\partial \mathbf{p}_p}{\partial s} \right\|^{-1} \left\| \mathbf{p}_{f,i}^f \right\| (1 + k_s)\right) - v_{2,\max}. \tag{5.34}$$

Now, assuming the existence of an upper bound on the product

$$\|\boldsymbol{\omega}_p(s)\| \, \|\partial \mathbf{p}_p(s)/\partial s\|^{-1},$$

there exists a positive constant k_{NSB} such that for every vehicle

$$\|\mathbf{v}_{\mathrm{NSB},i}\| \geq (1 - k_{\mathrm{NSB}}) U_{\mathrm{LOS}} - v_{2,\max}. \tag{5.35}$$

Assuming that $k_{\mathrm{NSB}} < 1$, we can satisfy (5.29) by choosing

$$U_{\mathrm{LOS}} = \frac{v_{2,\max} + \max_i \sqrt{v_i^2 + w_i^2 + u_{\min}^2}}{1 - k_{\mathrm{NSB}}}. \tag{5.36}$$

However, the max function would introduce switching behavior. To avoid this, we approximate the former with

$$U_{\mathrm{LOS}} = \frac{v_{2,\max} + \sqrt{\sum_{i=1}^{N} (v_i^2 + w_i^2) + u_{\min}^2}}{1 - k_{\mathrm{NSB}}}. \tag{5.37}$$

If the avoidance or depth-limiting tasks are active, we still choose U_{LOS} in accordance with (5.37). However, since (5.29) cannot be satisfied with a generic NSB velocity (5.7), we choose the surge reference as

$$u_{d,i} = \begin{cases} \sqrt{\|\mathbf{v}_{\mathrm{NSB},i}\|^2 - v_i^2 - w_i^2}, & \text{if (5.29) satisfied,} \\ u_{\min}, & \text{otherwise.} \end{cases} \tag{5.38}$$

Finally, let us discuss the choice of desired orientation. Let $\overline{\mathbf{v}}_{\mathrm{NSB},i}$ and $\overline{\boldsymbol{v}}_i$ denote normalized vectors. We are seeking $\mathbf{R}_{d,i} \in SO(3)$ such that

$$\overline{\mathbf{v}}_{\mathrm{NSB},i} = \mathbf{R}_{d,i} \, \overline{\boldsymbol{v}}_i. \tag{5.39}$$

Assume that at a given time, there is $\mathbf{R}_{d,i}$ that satisfies (5.39). Differentiating (5.39) with respect to time yields

$$\dot{\overline{\mathbf{v}}}_{\mathrm{NSB},i} = \mathbf{R}_{d,i} \, \mathbf{S}(\boldsymbol{\omega}_{d,i}) \, \overline{\boldsymbol{v}}_i + \mathbf{R}_{d,i} \, \dot{\overline{\boldsymbol{v}}}_i, \tag{5.40}$$

where $\boldsymbol{\omega}_{d,i}$ is the desired angular velocity of the vehicle. Let us define

$$\boldsymbol{\omega}_{\mathbf{v}_{\mathrm{NSB},i}} = \overline{\mathbf{v}}_{\mathrm{NSB},i} \times \dot{\overline{\mathbf{v}}}_{\mathrm{NSB},i}, \qquad \boldsymbol{\omega}_{\boldsymbol{v}_i} = \overline{\boldsymbol{v}}_i \times \dot{\overline{\boldsymbol{v}}}_i. \tag{5.41}$$

Then, (5.40) can be rewritten as

$$\boldsymbol{\omega}_{\mathbf{v}_{\mathrm{NSB},i}} \times \overline{\mathbf{v}}_{\mathrm{NSB},i} = \mathbf{R}_{d,i} \left(\boldsymbol{\omega}_{d,i} \times \overline{\boldsymbol{v}}_i + \boldsymbol{\omega}_{\boldsymbol{v}_i} \times \overline{\boldsymbol{v}}_i\right). \tag{5.42}$$

Therefore, the desired angular velocity must satisfy

$$\left(\boldsymbol{\omega}_{d,i} + \boldsymbol{\omega}_{\boldsymbol{\upsilon}_i} - \mathbf{R}_{d,i}^{\mathrm{T}} \boldsymbol{\omega}_{\mathbf{v}_{\mathrm{NSB},i}}\right) \times \overline{\boldsymbol{\upsilon}}_i = \mathbf{0}. \tag{5.43}$$

Thus, instead of finding $\mathbf{R}_{d,i}$ directly, we propose to choose

$$\boldsymbol{\omega}_{d,i} = \mathbf{R}_{d,i}^{\mathrm{T}} \boldsymbol{\omega}_{\mathbf{v}_{\mathrm{NSB},i}} - \boldsymbol{\omega}_{\boldsymbol{\upsilon}_i}, \tag{5.44}$$

and then evolve the desired orientation according to

$$\dot{\mathbf{R}}_{d,i} = \mathbf{R}_{d,i} \mathbf{S}(\boldsymbol{\omega}_{d,i}). \tag{5.45}$$

Note that choosing $\boldsymbol{\omega}_{d,i}$ according to (5.44) leads to the smallest (in terms of Euclidean norm) angular velocity that satisfies (5.43). We also note that there exists a subspace of angular velocities that satisfy (5.43) and a subspace of rotation matrices that satisfy (5.39). This differs from 3DOF [33, 35] and 5DOF [83] models, for which only one solution exists.

5.3 Closed-Loop Analysis

In this section, we analyze the closed-loop behavior of the system. Throughout this section, we assume that neither the avoidance nor depth-limiting tasks are active. Let us define the combined formation-keeping and path-following error as

$$\widetilde{\boldsymbol{\sigma}} = \left[\widetilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}, (\mathbf{p}_b^p)^{\mathrm{T}}\right]^{\mathrm{T}}, \tag{5.46}$$

and the combined low-level controller error as

$$\widetilde{\mathbf{X}} = \left[\widetilde{\mathbf{X}}_1^{\mathrm{T}}, \ldots, \widetilde{\mathbf{X}}_N^{\mathrm{T}}\right]^{\mathrm{T}}. \tag{5.47}$$

First, let us investigate the closed-loop dynamics of $\widetilde{\boldsymbol{\sigma}}$. Differentiating (5.11), (5.12), and (5.5) with respect to time yields

$$\dot{\widetilde{\boldsymbol{\sigma}}}_2 = \mathbf{J}_2 \dot{\mathbf{p}} - \dot{\boldsymbol{\sigma}}_{d,2}, \qquad \dot{\mathbf{p}} = \left[\dot{\mathbf{p}}_1^{\mathrm{T}}, \ldots, \dot{\mathbf{p}}_N^{\mathrm{T}}\right]^{\mathrm{T}} \tag{5.48a}$$

$$\dot{\mathbf{p}}_b^p = \mathbf{R}_p^{\mathrm{T}} \left(\frac{1}{N} \sum_{i=1}^{N} \dot{\mathbf{p}}_i - \dot{\mathbf{p}}_p\right) - \mathbf{S}\left(\boldsymbol{\omega}_p \dot{s}\right) \mathbf{p}_b^p. \tag{5.48b}$$

From (5.1a) and (5.2) it follows that $\dot{\mathbf{p}}_i$ is given by

$$\dot{\mathbf{p}}_i = \mathbf{R}_i \boldsymbol{\upsilon}_i = \operatorname{expm}(\boldsymbol{\delta}_i) \mathbf{R}_{d,i} \left[u_{d,i} + \widetilde{u}_i, v_i, w_i\right]^{\mathrm{T}}, \tag{5.49}$$

with

$$\operatorname{expm}(\boldsymbol{\delta}) = \cos\theta \, \mathbf{I} + s\mathbf{S}(\boldsymbol{\delta}) + c\mathbf{S}(\boldsymbol{\delta})^2, \qquad \begin{aligned} \theta &= \|\boldsymbol{\delta}\|, \\ s &= \tfrac{\sin(\theta)}{\theta}, \\ c &= \tfrac{1-\cos(\theta)}{\theta^2}. \end{aligned} \tag{5.50}$$

Substituting (5.50), (5.27), and (5.39) into (5.49) we get

$$\dot{\mathbf{p}}_i = \mathbf{v}_{\mathrm{NSB},i} + s(\boldsymbol{\delta}_i \times \mathbf{v}_{\mathrm{NSB},i}) + c\,\boldsymbol{\delta}_i \times (\boldsymbol{\delta}_i \times \mathbf{v}_{\mathrm{NSB},i}) + \mathbf{R}_i \left[\widetilde{u}_i, 0, 0\right]^{\mathrm{T}}. \tag{5.51}$$

Defining a perturbing term $\mathbf{g}_i$ as

$$\mathbf{g}_i = s(\boldsymbol{\delta}_i \times \mathbf{v}_{\mathrm{NSB},i}) + c\,\boldsymbol{\delta}_i \times (\boldsymbol{\delta}_i \times \mathbf{v}_{\mathrm{NSB},i}) + \mathbf{R}_i \left[\widetilde{u}_i, 0, 0\right]^{\mathrm{T}}, \tag{5.52}$$

and substituting (5.52) and (5.51) into (5.48) yields

$$\dot{\widetilde{\boldsymbol{\sigma}}}_2 = \mathbf{J}_2 \mathbf{v}_{\mathrm{NSB}} - \dot{\boldsymbol{\sigma}}_{d,2} + \mathbf{J}_2 \mathbf{G}, \qquad \mathbf{G} = \left[\mathbf{g}_1^{\mathrm{T}}, \ldots, \mathbf{g}_N^{\mathrm{T}}\right]^{\mathrm{T}} \tag{5.53a}$$

$$\dot{\mathbf{p}}_b^p = \mathbf{R}_p^{\mathrm{T}} \left(\frac{1}{N} \sum_{i=1}^{N} (\mathbf{v}_{\mathrm{NSB},i} + \mathbf{g}_i) - \dot{\mathbf{p}}_p \right) - \mathbf{S}\big(\boldsymbol{\omega}_p \dot{s}\big)\mathbf{p}_b^p. \tag{5.53b}$$

Now, to account for the underactuated dynamics, we define a vector of concatenated sway and heave velocities as

$$\boldsymbol{\upsilon}_u = \left[v_1, w_1, \ldots, v_N, w_N\right]^{\mathrm{T}}, \qquad \boldsymbol{\upsilon}_{u,c} = \mathbf{1}_N \otimes \left[v_c, w_c\right]^{\mathrm{T}}. \tag{5.54}$$

The underactuated dynamics can then be written as

$$\dot{\boldsymbol{\upsilon}}_u = \mathbf{X}\boldsymbol{\Omega} + \mathbf{Y}\left(\boldsymbol{\upsilon}_u - \boldsymbol{\upsilon}_{u,c}\right) + \dot{\boldsymbol{\upsilon}}_{u,c}, \tag{5.55}$$

where $\boldsymbol{\Omega} = \left[\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_N\right]^{\mathrm{T}}$, and $\mathbf{X}$ and $\mathbf{Y}$ are block diagonal matrices consisting of blocks $\mathbf{X}_1, \ldots, \mathbf{X}_N$ and $\mathbf{Y}_1, \ldots, \mathbf{Y}_N$, that are given by

$$\mathbf{X}_i = \begin{bmatrix} 0 & 0 & X_v(u_{r,i}) \\ 0 & X_w(u_{r,i}) & 0 \end{bmatrix}, \qquad \mathbf{Y}_i = \begin{bmatrix} Y_v(u_{r,i}) & Z_v(p_i) \\ Z_w(p_i) & Y_w(u_{r,i}) \end{bmatrix}. \tag{5.56}$$

Theorem 5.1. *Let Assumptions 2.1–5.1 be satisfied. Then,* $\left[\widetilde{\boldsymbol{\sigma}}^{\mathrm{T}}, \widetilde{\boldsymbol{X}}^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$ *is a uniformly semiglobally exponentially stable (USGES) equilibrium point of the closed-loop system* (5.53), (5.3), (5.55). *Moreover, let* X *be the largest signular value of* $\mathbf{X}$ *and let* Y *be the smallest eigenvalue of* $-\mathbf{Y}$. *Then, the underactuated sway and heave dynamics are bounded near the manifold* $\left[\widetilde{\boldsymbol{\sigma}}^{\mathrm{T}}, \widetilde{\boldsymbol{X}}^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$ *if the second and third partial derivatives of* $\mathbf{p}_p(s)$ *with respect to* s *are bounded and* X *and* Y *satisfy* $Y > aX$, *where* a *is a positive constant that will be defined later in the proof.*

Proof. We analyze the closed-loop system as a cascade where $\widetilde{\boldsymbol{X}}$ perturbs the dynamics of $\widetilde{\boldsymbol{\sigma}}$ through $\mathbf{G}$. Consider the nominal dynamics of $\widetilde{\boldsymbol{\sigma}}$ (*i.e.*, (5.53) with $\mathbf{G} = \mathbf{0}$) and the following Lyapunov function candidate

$$V = \frac{1}{2}\,\widetilde{\boldsymbol{\sigma}}^{\mathrm{T}}\widetilde{\boldsymbol{\sigma}} = \frac{1}{2}\left(\widetilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}\widetilde{\boldsymbol{\sigma}}_2 + \left(\mathbf{p}_b^p\right)^{\mathrm{T}}\mathbf{p}_b^p\right). \tag{5.57}$$

The time-derivative of V is

$$\dot{V} = \widetilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}(\mathbf{J}_2\mathbf{v}_{\mathrm{NSB}} - \dot{\boldsymbol{\sigma}}_{d,2}) - \left(\mathbf{p}_b^p\right)^{\mathrm{T}}\mathbf{S}(\boldsymbol{\omega}_p\dot{s})\mathbf{p}_b^p + \left(\mathbf{p}_b^p\right)^{\mathrm{T}}\mathbf{R}_p^{\mathrm{T}}\left(\frac{1}{N}\sum_{i=1}^{N}\mathbf{v}_{\mathrm{NSB},i} - \dot{p}_p\right). \tag{5.58}$$

Due to the properties of the NSB tasks defined in Sections 5.2.3 and 5.2.4, the following identities hold:

$$\mathbf{J}_2\mathbf{v}_{\mathrm{NSB}} = \mathbf{J}_2\mathbf{v}_2, \qquad \sum_{i=1}^{N} \mathbf{v}_{\mathrm{NSB},i} = \mathbf{v}_{\mathrm{LOS}}. \tag{5.59}$$

By definition (see Section 5.1.2), $\mathbf{R}_p$ must satisfy

$$\mathbf{R}_p^{\mathrm{T}}\dot{\mathbf{p}}_p = \dot{s}\left\|\frac{\partial \mathbf{p}_p(s)}{\partial s}\right\|^{-1}\mathbf{e}_1, \qquad \mathbf{e}_1 = [1,0,0]^{\mathrm{T}}. \tag{5.60}$$

Substituting (5.13) and (5.16) into (5.58) leads to

$$\begin{aligned}\dot{V} = &-v_{2,\max}\widetilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}\mathrm{sat}\left(\boldsymbol{\Lambda}_2\widetilde{\boldsymbol{\sigma}}_2\right)\\ &- U_{\mathrm{LOS}}\left(k_s\frac{\left(x_b^p\right)^2}{\sqrt{1+\left(x_b^p\right)^2}} + \frac{\left(y_b^p\right)^2}{D} + \frac{\left(z_b^p\right)^2}{D}\right).\end{aligned} \tag{5.61}$$

For any $\widetilde{\boldsymbol{\sigma}} \in \mathfrak{B}_r^{3N}$, the following holds:

$$\begin{aligned}\dot{V} \leq &-v_{2,\max}\lambda_{2,\min}\frac{\tanh(r)}{r}\left\|\widetilde{\boldsymbol{\sigma}}_2\right\|^2\\ &- U_{\mathrm{LOS}}\min\left\{\frac{k_s}{\sqrt{1+r^2}}, \frac{1}{\sqrt{\Delta_0^2+2r^2}}\right\}\left\|\mathbf{p}_b^p\right\|^2,\end{aligned} \tag{5.62}$$

where $\lambda_{2,\min}$ is the smallest eigenvalue of $\boldsymbol{\Lambda}_2$. From (5.62), we conclude that the derivative of V satisfies

$$\dot{V} \leq -k_r\left\|\widetilde{\boldsymbol{\sigma}}\right\|^2, \tag{5.63}$$

where

$$k_r = \min\left\{v_{2,\max}\lambda_{2,\min}\frac{\tanh(r)}{r}, \frac{U_{\mathrm{LOS}}k_s}{\sqrt{1+r^2}}, \frac{U_{\mathrm{LOS}}}{\sqrt{\Delta_0^2+2r^2}}\right\} \tag{5.64}$$

All assumptions of [70, Theorem 5] are thus satisfied, and the origin of the nominal system is USGES.

Moreover, note that the low-level controller is GES by Assumption 5.1. Therefore, if the two assumptions of Proposition 2.2 hold, the origin of the cascade is USGES. Since $\|\partial V/\partial\widetilde{\boldsymbol{\sigma}}\| = \|\widetilde{\boldsymbol{\sigma}}\|$, the first assumption is satisfied for $c_1 = 1/2$, $c_2 = \eta$, and any $\eta \in \mathbb{R}_{\geq 0}$.

To validate the second assumption, we first need to investigate the perturbing terms $\mathbf{g}_i$ from (5.52). From (5.30) we get the following upper bound on $\mathbf{v}_{\mathrm{NSB},i}$

$$\left\|\mathbf{v}_{\mathrm{NSB},i}\right\| \leq U_{\mathrm{LOS}}\left(1+k_{\mathrm{NSB}}\right) + v_{2,\max}\tanh\left(\left\|\widetilde{\boldsymbol{\sigma}}_2\right\|\right), \tag{5.65}$$

and from (5.50), we get the inequalities

$$s \leq 1, \qquad \|c\,\boldsymbol{\delta}\| \leq \sqrt{2}/2. \tag{5.66}$$

Therefore, $\mathbf{g}_i$ can be upper-bounded by

$$\left\|\mathbf{g}_i\right\| \leq \left\|\mathbf{v}_{\mathrm{NSB},i}\right\|\left(1+\sqrt{2}/2\right)\left\|\boldsymbol{\delta}_i\right\| + \left|\widetilde{u}_i\right|. \tag{5.67}$$

Consider then the two functions $\alpha_{1,i}, \alpha_{2,i} : \mathbb{R}_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$

$$\alpha_{1,i}(r) = \left(U_{\text{LOS}} \left(1 + k_{\text{NSB}}\right) \left(1 + \sqrt{2}/2\right) + 1\right) r, \tag{5.68}$$

$$\alpha_{2,i}(r) = v_{2,\max} \left(1 + \sqrt{2}/2\right) r. \tag{5.69}$$

Then, the following holds:

$$\|\mathbf{g}_i\| \leq \alpha_{1,i} \left(\left\|\widetilde{\mathbf{X}}_i\right\|\right) + \alpha_{2,i} \left(\left\|\widetilde{\mathbf{X}}_i\right\|\right) \|\widetilde{\boldsymbol{\sigma}}\|. \tag{5.70}$$

Therefore, (2.112) can be satisfied by

$$\alpha_1(r) = \sum_{i=1}^{N} \alpha_{1,i}(r), \qquad \alpha_2(r) = \sum_{i=1}^{N} \alpha_{2,i}(r), \tag{5.71}$$

and consequently all assumptions of [70, Proposition 9] are satisfied. To summarize, the origin of the closed-loop system is USGES.

As for the underactuated dynamics, the assumption $\widetilde{\mathbf{X}} = \mathbf{0}$ implies $\boldsymbol{\omega}_i = \boldsymbol{\omega}_{d,i}$ and $u_i = u_{d,i}$. Therefore the underactuated dynamics depend on the desired angular velocity. Recall the definition of $\boldsymbol{\omega}_{d,i}$ in (5.44). To find a closed-loop expression for $\boldsymbol{\omega}_{d,i}$, we shall analyze $\boldsymbol{\omega}_{\mathbf{v}_{\text{NSB},i}}$ and $\boldsymbol{\omega}_{\boldsymbol{v}_i}$.

First, we consider $\boldsymbol{\omega}_{\mathbf{v}_{\text{NSB},i}}$. In Appendix C.1, we show that there exist positive constants a_{NSB} and b_{NSB} such that

$$\left\|\boldsymbol{\omega}_{\mathbf{v}_{\text{NSB},i}}\right\| \leq a_{\text{NSB}} \|\boldsymbol{v}_u\| + b_{\text{NSB}}. \tag{5.72}$$

Now, let us consider $\boldsymbol{\omega}_{\boldsymbol{v}_i}$. In Appendix C.2, we show that $\boldsymbol{\omega}_{\boldsymbol{v}_i}$ depends on the angular velocities of the vehicle, thus forming an algebraic loop. However, under certain conditions, this loop can be resolved.

We show that $\boldsymbol{\omega}_{\boldsymbol{v}_i}$ is affine in $\boldsymbol{\omega}_i$. In other words, there exist $\boldsymbol{\omega}_{0,i}$ and $\mathbf{A}_{\boldsymbol{\omega}_i}$ such that

$$\boldsymbol{\omega}_{\boldsymbol{v}_i} = \boldsymbol{\omega}_{0,i} + \mathbf{A}_{\boldsymbol{\omega}_i} \boldsymbol{\omega}_i. \tag{5.73}$$

Moreover, we show that $\mathbf{A}_{\boldsymbol{\omega}_i}$ satisfies

$$\det\left(\mathbf{I} + \mathbf{A}_{\boldsymbol{\omega}_i}\right) \geq 1 - k_a, \tag{5.74}$$

where k_a is a positive constant depending on the physical properties of the vehicle, the minimum surge velocity, and the ocean current. If $k_a < 1$, then $(\mathbf{I} + \mathbf{A}_{\boldsymbol{\omega}_i})$ is invertible, and the desired angular velocity is

$$\boldsymbol{\omega}_{d,i} = \left(\mathbf{I} + \mathbf{A}_{\boldsymbol{\omega}_i}\right)^{-1} \left(\mathbf{R}_{d,i}^{\mathrm{T}} \boldsymbol{\omega}_{\mathbf{v}_{\text{NSB},i}} - \boldsymbol{\omega}_{0,i}\right). \tag{5.75}$$

In addition, there exist positive constants a_v, and b_v such that

$$\|\boldsymbol{\omega}_{0,i}\| \leq a_v \|\boldsymbol{v}_u\| + b_v. \tag{5.76}$$

By combining (5.72), (5.74), and (5.76), we can upper bound the angular velocity with

$$\|\boldsymbol{\omega}_{d,i}\| \leq \frac{\left(a_{\text{NSB}} + a_v\right) \|\boldsymbol{v}_u\| + b_{\text{NSB}} + b_v}{1 - k_a}. \tag{5.77}$$

The Lyapunov function candidate

$$V_u = \frac{1}{2}\boldsymbol{\upsilon}_u^{\mathrm{T}}\boldsymbol{\upsilon}_u \tag{5.78}$$

for the underactuated dynamics may then be shown that, leveraging (5.55), has its time-derivative bounded by

$$\dot{V}_u \leq \boldsymbol{\upsilon}_u^{\mathrm{T}}\mathbf{Y}\boldsymbol{\upsilon}_u + aX_{\max}\left\|\boldsymbol{\upsilon}_u\right\|^2 + H\left(\left\|\boldsymbol{\upsilon}_u\right\|, \left\|\mathbf{V}_c\right\|\right), \tag{5.79}$$

where $a = (a_{\mathrm{NSB}} + a_v)/(1 - k_a)$, $X_{\max}$ is the largest singular value of $\mathbf{X}$, and H represents the terms that grow at most linearly with $\boldsymbol{\upsilon}_u$. Since $\mathbf{Y}$ contains terms associated with hydrodynamic damping, it is negative definite. Therefore, $\dot{V}_u$ can be further bounded by

$$\dot{V}_u \leq -\left(Y_{\min} - aX_{\max}\right)\left\|\boldsymbol{\upsilon}_u\right\|^2 + H(\cdot), \tag{5.80}$$

where $Y_{\min}$ is the real part of the smallest eigenvalue of $-\mathbf{Y}$. For a sufficiently large $\boldsymbol{\upsilon}_u$, the quadratic terms will dominate the linear terms. Consequently, the underactuated dynamics are bounded if

$$Y_{\min} > aX_{\max}. \tag{5.81}$$

We have thus shown that the origin of the closed-loop system (5.53), (5.3), (5.55) is USGES, and the underactuated sway and heave dynamics are bounded. □

5.4 Simulations

We simulate the proposed approach on a fleet of six LAUVs [12] using MATLAB, delegating low-level control to an attitude-tracking PID controller as in [116] and an output-linearizing P surge controller as in [83].

The desired path is a spiral given by

$$\mathbf{p}_p(s) = \mathbf{p}_{p,0} + \left[s, a_p\cos(\omega_p s), b_p\sin(\omega_p s)\right]^{\mathrm{T}} \tag{5.82}$$

where

$$\mathbf{p}_{p,0} = \left[0, -40, 25\right]^{\mathrm{T}}, \qquad a_p = 40, \qquad b_p = 20, \qquad \omega_p = \tfrac{\pi}{100},$$

while the desired formation is an isosceles triangle parallel to the yz plane. Specifically, the desired positions in the formation-centered frame are

$$\mathbf{p}_{f,1}^f = \begin{bmatrix}0\\10\\5\end{bmatrix}, \qquad \mathbf{p}_{f,2}^f = \begin{bmatrix}0\\-10\\5\end{bmatrix}, \qquad \mathbf{p}_{f,3}^f = \begin{bmatrix}0\\0\\-10\end{bmatrix}. \tag{5.83}$$

For the simulation parameters, we choose the velocity of the ocean current to be $\mathbf{V}_c = [0, 0.15, 0.05]^{\mathrm{T}}$, the formation-keeping gain $\boldsymbol{\Lambda}_2 = 0.1\mathbf{I}$, the maximum formation-keeping velocity $v_{2,\max} = 0.5\,\mathrm{m\,s^{-1}}$, and the lookahead distance $\Delta_0 = 5\,\mathrm{m}$.

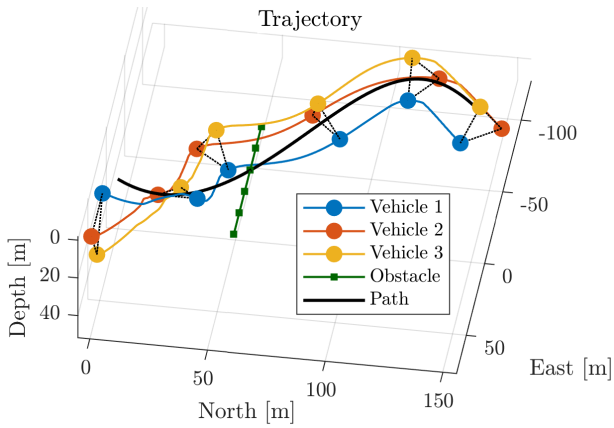


Figure 5.2: The 3D trajectory of the vehicles. The markers represent the position of the vehicles at times $t = 0, 50, \ldots, 250$ seconds. Markers with corresponding times are connected by dotted lines to better illustrate the resulting formation.

The very minimum relative distance to avoid collision is the length of the LAUV, *i.e.*, 2.4 m. For additional safety, we design the COLAV task with $d_{\min} = 5$ m. For additional safety during transients, d_{COLAV} is chosen to be 10 m.

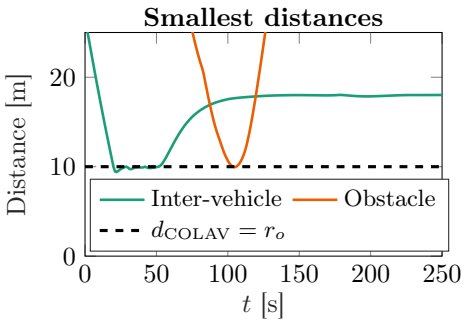
We then let the vehicles encounter an obstacle of similar size as the LAUV that moves east at a constant speed of $0.3\,\text{m}\,\text{s}^{-1}$. Given its size, we choose $r_o = d_{\text{COLAV}}$. The minimum cone angle is set to $\alpha_{\min} = 15°$. The operation limits are chosen as $z_{\min} = 1\,\text{m}$, $z_{\max} = 49\,\text{m}$, and the depth-limiting velocity is $v_z = 0.3\,\text{m}\,\text{s}^{-1}$. Note that the limits are deliberately chosen too small for the given path and formation, so that depth limiting is activated.

Figures 5.2 and 5.3 show the results of this numerical simulation. Figure 5.3a shows the distance between the vehicles and the distance to the obstacle. At $t = 20\,\text{s}$, the COLAV task is activated, and the distance between the vehicles drops to approximately 9.5 meters during the transient. The situation is resolved after 30 seconds. At $t = 35\,\text{s}$, the vehicles enter the collision cone and perform an evasive maneuver in a clockwise direction. The distance to the obstacle is always above the required limit.

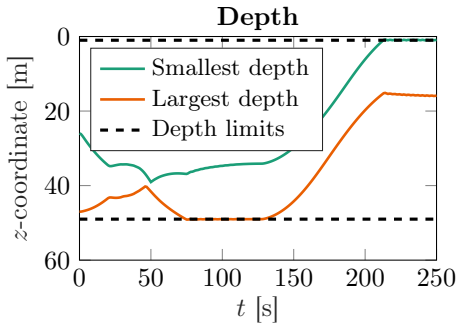
Figure 5.3b shows the depth of the vehicles. At $t = 73\,\text{s}$ and $t = 212\,\text{s}$, the depth-limiting task is activated. When the task is active, the depth of the vehicles fluctuates around the prescribed limit.

Figures 5.3c and 5.3d show the path-following and formation-keeping errors. We can see that the path-following errors diverge when obstacle avoidance or depth limiting is active. Conversely, the formation-keeping errors diverge during inter-agent COLAV. This behavior corresponds to the interpretation of the NSB tasks — path-following is global and thus cannot be satisfied during obstacle avoidance, whereas formation-keeping works with relative velocities and thus cannot be satisfied during inter-agent COLAV.

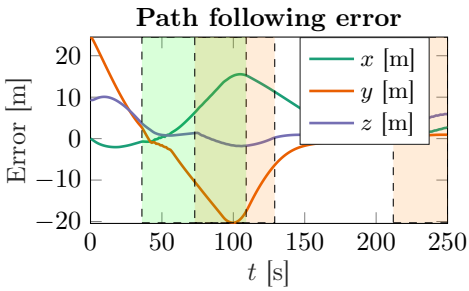
Figure 5.3e shows the surge velocity of the vehicles. We can see that the surge velocities are always above the required limit. In fact, our solution appears to be overly conservative. Figure 5.3f shows the sway and heave velocities. We can see



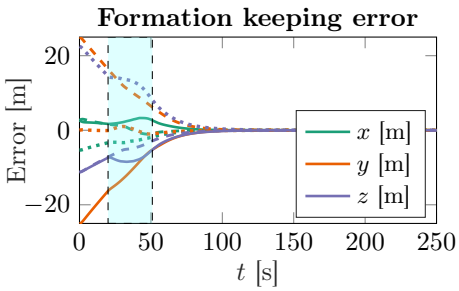
(a) The smallest inter-vehicle and vehicle-to-obstacle distance.



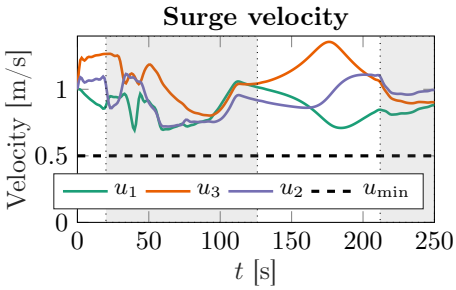
(b) The smallest and largest vehicle depth.



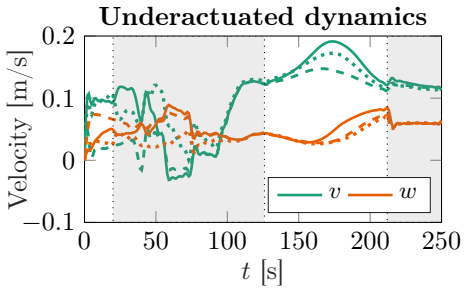
(c) The path-following error. The green rectangle represents the time when obstacle avoidance is active. The red rectangle represents the time when depth limiting is active.



(d) The formation-keeping error. The full, dashed, and dotted lines correspond to vehicles 1, 2, and 3, respectively. The blue rectangle represents the time when inter-agent COLAV is active.



(e) The surge velocities of the vehicles. The grey rectangle represents the time when inter-agent COLAV, obstacle avoidance, or depth limiting tasks are active.



(f) The sway and heave velocities. The full, dashed, and dotted lines correspond to vehicles 1, 2, and 3, respectively.

Figure 5.3: Simulation results.

that the velocities change abruptly when the collision avoidance or depth limiting tasks are active, as the vehicles switch to a different behavior. However, the velocities still remain bounded during the whole simulation. The peak in sway velocities at $t = 180\,\mathrm{s}$ coincides with the sharpest turn (*i.e.,* the largest $\boldsymbol{\omega}_p(s)$) of the desired path.

Chapter 6

A Distributed NSB Algorithm for Formation Path Following

This chapter presents a distributed null-space-behavioral algorithm for the formation path-following problem of vehicles moving in three dimensions. The algorithm is applied to fleets of underactuated autonomous underwater vehicles. The algorithm combines null-space-behavioral control with consensus methods. First, we present a continuous-time version of the algorithm and prove its stability using Lyapunov analysis. Then, we present a discrete-time event-triggered version that, compared to similar formation path-following methods, can achieve the same steady state-error performance with fewer inter-vehicle transmissions. The effectiveness of both the continuous-time and the discrete-time algorithm is verified in numerical simulations. Furthermore, the discrete-time version is tested in field experiments. The contents of this chaper are based on [79].

The chapter is organized as follows. Section 6.1 defines the formation path following problem. Section 6.2 presents the centralized NSB algorithm. In Section 6.3, we present the continuous-time distributed NSB algorithm, and in Section 6.4, we analyze its closed-loop behavior. Section 6.5 discusses how to apply the proposed algorithm to underactuated AUVs and proposes the discrete-time modification. Finally, Sections 6.6 and 6.7 present the results of numerical simulations and experiments, respectively.

6.1 Problem Definition

We begin by considering a group of N vehicles with single-integrator dynamics. An extension to more complex vehicle models will be shown in Section 6.5. Let $\mathbf{p}_i \in \mathbb{R}^3$ be the position of vehicle i. The kinematic equation of vehicle i is

$$\dot{\mathbf{p}}_i = \mathbf{v}_i, \tag{6.1}$$

where $\mathbf{v}_i \in \mathbb{R}^3$ is the input velocity.

The vehicles should follow a predefined path in a given formation. Let $\mathbf{p}_p : \mathbb{R} \mapsto \mathbb{R}^3$ be a parametrization of the desired path. We assume that the function is $\mathcal{C}^1$ and regular. Consequently, for any point $\mathbf{p}_p(s)$, there exists a path-tangential coordinate frame and a corresponding rotation matrix $\mathbf{R}_p(s)$.

The path-following error $\mathbf{p}_b^p$ is given by the position of the barycenter in the path-tangential coordinate frame

$$\mathbf{p}_b^p = \mathbf{R}_p^{\mathrm{T}} \left(\mathbf{p}_b - \mathbf{p}_p(s) \right), \qquad \mathbf{p}_b = \frac{1}{N} \sum_{i=1}^{N} \mathbf{p}_i. \tag{6.2}$$

The goal of path following is to control the vehicles so that $\mathbf{p}_b^p \to \mathbf{0}_3$.

The vehicles should converge to a dynamic formation that rotates with the desired path (see Section 2.3.2 for details). Let $\mathbf{p}_{f,1}^f, \ldots, \mathbf{p}_{f,N}^f$ be the position vectors that represent the desired formation. The objective is to control the vehicles so that

$$\mathbf{p}_i - \mathbf{p}_b \to \mathbf{R}_p(s) \mathbf{p}_{f,i}^f, \qquad \forall i \in \{1, \ldots, N\}. \tag{6.3}$$

6.2 NSB Algorithm for the Formation Path Following Problem

In this section, we present the centralized formation path-following NSB algorithm for vehicles moving in three dimensions. The algorithm is a simplified version of the two methods presented in Chapters 4 and 5.

For the sake of simplicity, we do not consider inter-vehicle collision avoidance. Instead, we focus on the formation path-following problem with obstacle avoidance. To solve the problem, we define two tasks: path following and formation keeping. Obstacle avoidance is not implemented as a separate task but rather as an extension of path following.

In the *nominal* case (*i.e.,* when obstacle avoidance is not active), the path-following velocity is given by the coupled line-of-sight (LOS) guidance law (2.71)

$$\mathbf{v}_{\mathrm{LOS}} = \frac{U_d}{D} \mathbf{R}_p \left[\Delta, -y_b^p, -z_b^p \right]^{\mathrm{T}}, \qquad D = \left\| \left[\Delta, -y_b^p, -z_b^p \right]^{\mathrm{T}} \right\|, \tag{6.4}$$

with a constant lookahead distance $\Delta > 0$.

The method for obstacle avoidance is identical to the one presented in Chapter 5. Let $\mathbf{p}_o \in \mathbb{R}^2$ and $\mathbf{v}_o \in \mathbb{R}^2$ denote the position and velocity of the obstacle. When the task is active, the x- and y-components of the LOS velocity are replaced by the obstacle avoidance velocity $\mathbf{v}_{\mathrm{OA}}$ given by

$$\mathbf{v}_{\mathrm{OA}} = \|\mathbf{v}_{\mathrm{rel}}\| \left[\cos(\psi_{\mathrm{OA}}), \sin(\psi_{\mathrm{OA}}) \right]^{\mathrm{T}} + \left[\dot{x}_o, \dot{y}_o \right]^{\mathrm{T}}, \tag{6.5}$$

$$\psi_{\mathrm{OA}} = \arctan_2 \left(y_o - y_b, x_o - x_b \right) \pm \alpha, \tag{6.6}$$

where $\mathbf{v}_{\mathrm{rel}}$ is the relative velocity and α is the collision cone angle, as defined in (5.21) and (5.22), respectively. Note that ψ_{OA} has two solutions corresponding to avoiding the obstacle in the clockwise and counterclockwise directions. Inspired by [115], we propose the following method for choosing a direction: When the conflict first happens, we choose the value of ψ_{OA} that is closer to the direction of $\mathbf{v}_{\mathrm{rel}}$. Afterwards, we maintain the same direction.

Finally, let us discuss formation keeping. The proposed formation-keeping law is analogous to (2.90) and (4.19). The formation-keeping velocity of the whole fleet is given by

$$\mathbf{v}_f = \mathbf{J}_f^{\dagger} \left(\dot{\boldsymbol{\sigma}}_{d,f} - \boldsymbol{\Lambda}_f (\boldsymbol{\sigma}_f - \boldsymbol{\sigma}_{d,f}) \right), \tag{6.7}$$

where $\mathbf{J}_f$ is the Jacobian of the formation-keeping task, $\boldsymbol{\sigma}_f$ is the formation-keeping task variable, and $\boldsymbol{\sigma}_{d,f}$ is its desired value, as defined in (2.88), (2.86), and (2.89), respectively.

Assuming that the gain matrix is chosen as $\boldsymbol{\Lambda}_f = k_f \mathbf{I}$, where $k_f > 0$ is a constant, then the formation-keeping velocity of vehicle i is given by

$$\mathbf{v}_{f,i} = -k_f \left(\boldsymbol{\sigma}_i - \boldsymbol{\sigma}_{d,i} \right) + \dot{\boldsymbol{\sigma}}_{d,i}, \tag{6.8}$$

where

$$\boldsymbol{\sigma}_i = \mathbf{p}_i - \mathbf{p}_b, \qquad \boldsymbol{\sigma}_{d,i} = \mathbf{R}_p \mathbf{p}_{f,i}^f. \tag{6.9}$$

Since the formation-keeping and path-following tasks are independent and orthogonal, the total desired NSB velocity of vehicle i is given by

$$\mathbf{v}_{\mathrm{NSB},i} = \mathbf{v}_{\mathrm{LOS}} + \mathbf{v}_{f,i}. \tag{6.10}$$

Similarly to (5.19), we choose the following update law for the path parameter

$$\dot{s} = U_d \left\| \frac{\partial \mathbf{p}_p(s)}{\partial s} \right\|^{-1} \left(\frac{\Delta}{D} + k_s f_s \left(x_b^p \right) \right), \tag{6.11}$$

where

$$f_s(x) = \frac{x}{\sqrt{1 + x^2}}, \tag{6.12}$$

and k_s is a positive gain.

6.3 Distributed NSB Algorithm

Now, let us discuss how to make the algorithm presented in the previous section distributed. To perform the NSB algorithm in a distributed manner, the vehicles do not need to know the exact positions of each other. They only need to know the position of the barycenter and the radius of the formation, and converge to a common value of the path parameter.

In this section, we assume that the AUVs exchange information continuously. The connections between the vehicles can be represented using an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \ldots, N\}$ is the set of vertices corresponding to the AUVs, and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of edges, where $(i, j) \in \mathcal{E}$ means that vehicle i can exchange information with vehicle j. Let $\mathbf{A}$ be the adjacency matrix and let $\mathbf{L}$ be the Laplacian matrix of the communications graph, respectively. Moreover, let $\mathcal{N}_i$ denote the set of neighbors of vehicle i. The set of neighbors is defined as $\mathcal{N}_i = \left\{ j \,\middle|\, \mathbf{A}_{ji} = 1 \right\}$, where $\mathbf{A}_{ji}$ is the element of $\mathbf{A}$ at row j, column i.

Remark. *To exchange information continuously, the vehicles require a communication channel with high bandwidth and small delays. For AUVs, underwater optical communications may provide such a channel [117]. However, optical communications tend to be unreliable. We address this issue in Section 6.5.3, where we modify the distributed NSB algorithm to handle discrete-time delayed communications.*

Let $\mathbf{p}_{b,i}$, $r_{f,i}$, and s_i denote vehicle i's estimates of the barycenter, formation radius, and path parameter, respectively. The vehicle can then calculate its LOS velocity as

$$\mathbf{v}_{\mathrm{LOS},i} = \frac{U_d}{D_i}\mathbf{R}_p(s_i)\left[\Delta, -y^p_{b,i}, -z^p_{b,i}\right]^{\mathrm{T}}, \tag{6.13}$$

where

$$\left[x^p_{b,i}, y^p_{b,i}, z^p_{b,i}\right]^{\mathrm{T}} = \mathbf{R}_p(s_i)^{\mathrm{T}}\left(\mathbf{p}_{b,i} - \mathbf{p}_p\left(s_i\right)\right), \tag{6.14}$$

$$D_i = \sqrt{\Delta^2 + \left(y^p_{b,i}\right)^2 + \left(z^p_{b,i}\right)^2}. \tag{6.15}$$

In addition, the vehicle can estimate the cone angle as

$$\alpha_i = \arcsin\left(\frac{r_o + r_{f,i}}{\|\mathbf{p}_{\mathrm{rel},i}\|}\right), \qquad \mathbf{p}_{\mathrm{rel},i} = \mathbf{p}_o - [x_{b,i}, y_{b,i}]^{\mathrm{T}}, \tag{6.16}$$

where $x_{b,i}$ and $y_{b,i}$ are the x- and y-components of $\mathbf{p}_{b,i}$. The vehicle then determines if there is a need for obstacle avoidance according to the rules described in Chapter 5, and calculates its obstacle avoidance velocity as

$$\mathbf{v}_{\mathrm{OA},i} = \|\mathbf{v}_{\mathrm{rel},i}\|\left[\cos(\psi_{\mathrm{OA},i}), \sin(\psi_{\mathrm{OA},i})\right]^{\mathrm{T}} + [\dot{x}_o, \dot{y}_o]^{\mathrm{T}}, \tag{6.17}$$

$$\psi_{\mathrm{OA},i} = \arctan_2\left(y_o - y_{b,i}, x_o - x_{b,i}\right) \pm \alpha_i. \tag{6.18}$$

The distributed formation-keeping velocity is calculated as

$$\mathbf{v}_{f,i} = -k_f\left(\widehat{\boldsymbol{\sigma}}_i - \widehat{\boldsymbol{\sigma}}_{d,i}\right) + \dot{\widehat{\boldsymbol{\sigma}}}_{d,i}, \tag{6.19}$$

where

$$\widehat{\boldsymbol{\sigma}}_i = \mathbf{p}_i - \mathbf{p}_{b,i}, \qquad \widehat{\boldsymbol{\sigma}}_{d,i} = \mathbf{R}_p\mathbf{p}^f_{f,i}. \tag{6.20}$$

The path-following and formation-keeping velocities are then combined to obtain the desired NSB velocity

$$\mathbf{v}_{\mathrm{NSB},i} = \mathbf{v}_{\mathrm{LOS},i} + \mathbf{v}_{f,i}. \tag{6.21}$$

Finally, let us discuss the update laws for the estimates $\mathbf{p}_{b,i}$, $r_{f,i}$, and s_i. We assume that the vehicles communicate over a connected undirected graph. Each vehicle transmits its own estimates $\mathbf{p}_{b,i}$, s_i, and $r_{f,i}$. We propose the following equations to update the barycenter and path parameter estimates

$$\dot{\mathbf{p}}_{b,i} = \mathbf{v}_{\mathrm{LOS},i} + k_b\left(\widehat{\boldsymbol{\sigma}}_i - \widehat{\boldsymbol{\sigma}}_{d,i}\right) + c_p\sum_{j\in\mathcal{N}_i}\left(\mathbf{p}_{b,j} - \mathbf{p}_{b,i}\right), \tag{6.22}$$

$$\dot{s}_i = U_d\left\|\frac{\partial\mathbf{p}_p(s_i)}{\partial s_i}\right\|^{-1}\left(\frac{\Delta}{D_i} + k_s f_s\left(x^p_{b,i}\right)\right) + c_s\sum_{j\in\mathcal{N}_i}(s_j - s_i), \tag{6.23}$$

where k_b, c_p, and c_s are positive gains, and $\mathcal{N}_i$ is the set of neighbors of vehicle i. Let us briefly discuss the reasoning behind these equations. The first term in (6.22) assumes that the barycenter moves according to the LOS guidance algorithm. One can verify that if the barycenter estimates are correct and the path parameters are equal for all vehicles, then the true barycenter will, in fact, follow the LOS guidance law. The second term provides feedback from the vehicle's position. In the next section, we will show that this term steers the barycenter estimates towards the true barycenter. The first term in (6.23) is analogous to the path parameter update law (6.11). The last terms in (6.22) and (6.23) represent the consensus algorithm. This algorithm allows the barycenter estimates and path parameters to converge to a common value.

The formation radius estimate is updated according to the following hybrid scheme

$$\dot{r}_{f,i} = k_r \left(\widehat{r}_{f,i} - r_{f,i} \right), \tag{6.24a}$$

$$r_{f,i}^{+} = \widehat{r}_{f,i}, \qquad \text{if } \widehat{r}_{f,i} > r_{f,i}, \tag{6.24b}$$

$$r_{f,i}^{+} = \max_{j \in \mathcal{N}_i} r_{f,j}, \qquad \text{if } \max_{j \in \mathcal{N}_i} r_{f,j} > r_{f,i}, \tag{6.24c}$$

where k_r is a positive constant and $\widehat{r}_{f,i} = \|\mathbf{p}_{b,i} - \mathbf{p}_i\|$. We note that if the barycenter estimates are accurate, the formation radius is given by $r_f = \max_i \widehat{r}_{f,i}$. Equations (6.24b) and (6.24c) represent a max-consensus scheme, ensuring that the values of $r_{f,i}$ are always greater than or equal to $\max_i \widehat{r}_{f,i}$. The continuous-time update law (6.24a) allows the values of $r_{f,i}$ to converge to $\max_i \widehat{r}_{f,i}$.

6.4 Closed-Loop Analysis

In this section, we define the error variables and investigate their closed-loop behavior. In the subsequent proofs, we consider the *nominal* case without obstacle avoidance. Furthermore, we assume that the desired path is a straight line. Consequently, the rotation matrix $\mathbf{R}_p$ is constant, and the path is parametrized by

$$\mathbf{p}_p(s) = \mathbf{p}_0 + \mathbf{R}_p \left[s, 0, 0 \right]^{\mathrm{T}}, \tag{6.25}$$

where $\mathbf{p}_0 \in \mathbb{R}^3$ is the origin of the path.

Now, let us define the error variables. The path-following error is given by $\mathbf{p}_b^p$. For formation-keeping, we define $\widetilde{\boldsymbol{\sigma}}_1, \ldots, \widetilde{\boldsymbol{\sigma}}_N$ as

$$\widetilde{\boldsymbol{\sigma}}_i = \mathbf{p}_i - \mathbf{p}_b - \mathbf{R}_p \mathbf{p}_{f,i}^f. \tag{6.26}$$

The "true value" of the path parameter, s, is given by the mean of s_i, *i.e.*, $s = \frac{1}{N} \sum_{i=1}^{N} s_i$. Consequently, the barycenter and path parameter estimate errors, $\widetilde{\mathbf{p}}_{b,i}$ and $\tilde{s}_i$, are given by

$$\widetilde{\mathbf{p}}_{b,i} = \mathbf{p}_{b,i} - \mathbf{p}_b, \qquad \tilde{s}_i = s_i - s. \tag{6.27}$$

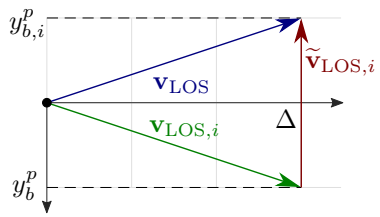


Figure 6.1: Illustration of the inequality (6.31).

6.4.1 Closed-Loop Dynamics

In this section, we derive closed-loop equations for the error variables.

First, let us investigate the closed-loop properties of the path-following error. The barycenter kinematics are given by

$$\dot{\mathbf{p}}_b^p = \mathbf{R}_p^{\mathrm{T}} \left(\frac{1}{N} \sum_{i=1}^{N} \mathbf{v}_{\mathrm{NSB},i} - \dot{\mathbf{p}}_p(s) \right) = \frac{1}{N} \sum_{i=1}^{N} \mathbf{R}_p^{\mathrm{T}} \left(\mathbf{v}_{\mathrm{LOS},i} + \mathbf{v}_{f,i} \right) - [\dot{s}, 0, 0]^{\mathrm{T}} . \quad (6.28)$$

Let us define the "true" LOS velocity, $\mathbf{v}_{\mathrm{LOS}}$, according to (6.4). Then, the LOS velocity calculated by vehicle i can be expressed as

$$\mathbf{v}_{\mathrm{LOS},i} = \mathbf{v}_{\mathrm{LOS}} + \widetilde{\mathbf{v}}_{\mathrm{LOS},i}, \quad (6.29)$$

where

$$\widetilde{\mathbf{v}}_{\mathrm{LOS},i} = \frac{U_d}{D_i} \mathbf{R}_p \left[\Delta, -y^p_{b,i}, -z^p_{b,i} \right]^{\mathrm{T}} - \frac{U_d}{D} \mathbf{R}_p \left[\Delta, -y^p_b, -z^p_b \right]^{\mathrm{T}} \quad (6.30)$$

It is straightforward to show that $\widetilde{\mathbf{v}}_{\mathrm{LOS},i} = \mathbf{0}$ if $\widetilde{\mathbf{p}}_{b,i} = \mathbf{0}$. Moreover, the norm of the LOS velocity error satisfies the following inequality

$$\| \widetilde{\mathbf{v}}_{\mathrm{LOS},i} \| \leq \frac{U_d}{\Delta} \| \widetilde{\mathbf{p}}_{b,i} \| . \quad (6.31)$$

We can illustrate this property on the following two-dimensional example. Consider a situation where y^p_b and $y^p_{b,i}$ have the same magnitude but opposite signs, and $z^p_b = z^p_{b,i} = 0$ (see Figure 6.1). Then, the norm of the LOS error is given by

$$\begin{aligned} \| \widetilde{\mathbf{v}}_{\mathrm{LOS},i} \| &= \left\| \frac{U_d}{\sqrt{\Delta^2 - (y^p_b)^2}} \left(\begin{bmatrix} \Delta \\ -y^p_{b,i} \end{bmatrix} - \begin{bmatrix} \Delta \\ -y^p_b \end{bmatrix} \right) \right\| = \frac{2 U_d y^p_b}{\sqrt{\Delta^2 - (y^p_b)^2}} \\ &\leq \frac{U_d}{\Delta} \| \widetilde{\mathbf{p}}_{b,i} \| . \end{aligned} \quad (6.32)$$

The case where the errors have opposite signs is, in fact, the "worst-case scenario". In all other cases, the inequality (6.31) holds as well.

To further investigate the barycenter kinematics in (6.28), we need to derive a closed-loop expression for $\dot{s}$. From (6.23), $\dot{s}$ is given by

$$\dot{s} = \frac{1}{N} \sum_{i=1}^{N} \dot{s}_i = \frac{1}{N} \sum_{i=1}^{N} U_d \left(\frac{\Delta}{D_i} + k_s f_s \left(x^p_{b,i} \right) \right) . \quad (6.33)$$

Note that since the vehicles communicate over an undirected graph, the consensus terms cancel out, and from the definition of a straight-line path in (6.25), it follows that $\left\| \frac{\partial \mathbf{p}_p(s)}{\partial s} \right\| = 1$.

Now, let us define path parameter update errors, $g_1, \ldots, g_N$, as

$$g_i = U_d \left(\frac{\Delta}{D_i} - \frac{\Delta}{D} + k_s \left(f_s\left(x_{b,i}^p\right) - f_s(x_b^p) \right) \right). \tag{6.34}$$

Note that if both $\widetilde{\mathbf{p}}_{b,i}$ and $\tilde{s}_i$ are zero, then g_i is zero as well. Moreover, it can be shown that g_i satisfies the following inequality

$$|g_i| \leq U_d \left(\frac{1}{\Delta} + k_s \right) \|\widetilde{\mathbf{p}}_{b,i}\| + U_d k_s \left| \tilde{s}_i \right|. \tag{6.35}$$

Substituting (6.34) into (6.33), we get

$$\dot{s} = U_d \left(\frac{\Delta}{D} + k_s f_s(x_b^p) \right) + \frac{1}{N} \sum_{i=1}^{N} g_i. \tag{6.36}$$

Finally, let us investigate the sum of formation-keeping velocities. From (6.19), we get

$$\begin{aligned} \frac{1}{N} \sum_{i=1}^{N} \mathbf{R}_p^{\mathrm{T}} \mathbf{v}_{f,i} &= -\frac{1}{N} \sum_{i=1}^{N} \mathbf{R}_p^{\mathrm{T}} k_f \left(\mathbf{p}_i - \mathbf{p}_{b,i} - \mathbf{R}_p \mathbf{p}_{f,i}^f \right) \\ &= -\frac{1}{N} \sum_{i=1}^{N} \mathbf{R}_p^{\mathrm{T}} k_f \left(\mathbf{p}_i - \mathbf{p}_b - \widetilde{\mathbf{p}}_{b,i} - \mathbf{R}_p \mathbf{p}_{f,i}^f \right) \\ &= -\frac{k_f}{N} \mathbf{R}_p^{\mathrm{T}} \left(\sum_{i=1}^{N} (\mathbf{p}_i - \mathbf{p}_b) - \sum_{i=1}^{N} \widetilde{\mathbf{p}}_{b,i} - \sum_{i=1}^{N} \mathbf{R}_p \mathbf{p}_{f,i}^f \right) \\ &= \frac{k_f}{N} \mathbf{R}_p^{\mathrm{T}} \sum_{i=1}^{N} \widetilde{\mathbf{p}}_{b,i}. \end{aligned} \tag{6.37}$$

Substituting (6.29), (6.36), and (6.37) into (6.28), we get

$$\dot{\mathbf{p}}_b^p = -U_d \left[k_s f_s(x_b^p), \frac{1}{D} y_b^p, \frac{1}{D} z_b^p \right]^{\mathrm{T}} + \frac{1}{N} \sum_{i=1}^{N} \left(\mathbf{R}_p^{\mathrm{T}} (\widetilde{\mathbf{v}}_{\mathrm{LOS},i} + k_f \widetilde{\mathbf{p}}_{b,i}) - g_i \right). \tag{6.38}$$

Next, we find a closed-loop equation for the formation-keeping errors. Differentiating (6.26) with respect to time yields

$$\dot{\widetilde{\boldsymbol{\sigma}}}_i = \dot{\mathbf{p}}_i - \dot{\mathbf{p}}_b = \mathbf{v}_{\mathrm{LOS},i} + \mathbf{v}_{f,i} - \frac{1}{N} \sum_{j=1}^{N} \left(\mathbf{v}_{\mathrm{LOS},j} + \mathbf{v}_{f,j} \right) \tag{6.39}$$

Substituting (6.29) and (6.19) into (6.39) yields

$$\dot{\widetilde{\boldsymbol{\sigma}}}_i = -k_f \widetilde{\boldsymbol{\sigma}}_i + \widetilde{\mathbf{v}}_{\mathrm{LOS},i} - k_f \widetilde{\mathbf{p}}_{b,i} + \frac{1}{N} \sum_{j=1}^{N} \left(\widetilde{\mathbf{v}}_{\mathrm{LOS},j} - k_f \widetilde{\mathbf{p}}_{b,j} \right). \tag{6.40}$$

This equation can be rewritten in a more compact form. To do so, let us define the following concatenated error vectors

$$\widetilde{\mathbf{V}}_{\mathrm{LOS}}^{\mathrm{T}} = \left[\widetilde{\mathbf{v}}_{\mathrm{LOS},1}^{\mathrm{T}}, \ldots, \widetilde{\mathbf{v}}_{\mathrm{LOS},N}^{\mathrm{T}}\right], \qquad \widetilde{\boldsymbol{\Sigma}}^{\mathrm{T}} = \left[\widetilde{\boldsymbol{\sigma}}_1^{\mathrm{T}}, \ldots, \widetilde{\boldsymbol{\sigma}}_N^{\mathrm{T}}\right], \tag{6.41a}$$

$$\widetilde{\mathbf{P}}_b^{\mathrm{T}} = \left[\widetilde{\mathbf{p}}_{b,1}^{\mathrm{T}}, \ldots, \widetilde{\mathbf{p}}_{b,N}^{\mathrm{T}}\right], \qquad \widetilde{\mathbf{S}}^{\mathrm{T}} = \left[\tilde{s}_1, \ldots, \tilde{s}_N\right]. \tag{6.41b}$$

Furthermore, let $\mathbf{L}_F$ denote the Laplacian matrix of a fully connected graph with N nodes ($\mathbf{L}_F = N\mathbf{I}_N - \mathbf{1}_N$). Then, (6.40) can be written in the following form

$$\dot{\widetilde{\boldsymbol{\Sigma}}} = -k_f \widetilde{\boldsymbol{\Sigma}} + \left(\frac{1}{N}\mathbf{L}_F \otimes \mathbf{I}_3\right)\left(\widetilde{\mathbf{V}}_{\mathrm{LOS}} - k_f \widetilde{\mathbf{P}}_b\right). \tag{6.42}$$

Finally, we find a closed-loop equation for the estimation errors. From (6.27), the time-derivative of $\widetilde{\mathbf{p}}_{b,i}$ is given by

$$\begin{aligned}\dot{\widetilde{\mathbf{p}}}_{b,i} &= \dot{\mathbf{p}}_{b,i} - \dot{\mathbf{p}}_b \qquad (6.43)\\ &= \widetilde{\mathbf{v}}_{\mathrm{LOS},i} + k_b\left(\widetilde{\boldsymbol{\sigma}}_i - \widetilde{\mathbf{p}}_{b,i}\right) + c_p \sum_{j\in\mathcal{N}_i}\left(\widetilde{\mathbf{p}}_{b,j} - \widetilde{\mathbf{p}}_{b,i}\right) - \frac{1}{N}\sum_{j=1}^{N}\left(\widetilde{\mathbf{v}}_{\mathrm{LOS},j} - k_f\widetilde{\mathbf{p}}_{b,j}\right).\end{aligned}$$

This equation can be written in the following compact form

$$\dot{\widetilde{\mathbf{P}}}_b = -\left(\left(k_b\mathbf{I}_N + \frac{k_f}{N}\mathbf{1}_N + c_p\mathbf{L}\right)\otimes\mathbf{I}_3\right)\widetilde{\mathbf{P}}_b + k_b\widetilde{\boldsymbol{\Sigma}} + \left(\frac{1}{N}\mathbf{L}_F\otimes\mathbf{I}_3\right)\widetilde{\mathbf{V}}_{\mathrm{LOS}}, \tag{6.44}$$

where $\mathbf{L}$ is the Laplacian matrix of the communications graph.

The time-derivative of $\tilde{s}_i$ is given by

$$\dot{\tilde{s}}_i = \dot{s}_i - \dot{s} = c_s\sum_{j\in\mathcal{N}_i}(\tilde{s}_j - \tilde{s}_i) + g_i - \frac{1}{N}\sum_{j=1}^{N} g_j. \tag{6.45}$$

This equation can be written in the following compact form

$$\dot{\widetilde{\mathbf{S}}} = -c_s\mathbf{L}\widetilde{\mathbf{S}} + \frac{1}{N}\mathbf{L}_F\left[g_1, \ldots, g_N\right]^{\mathrm{T}}. \tag{6.46}$$

6.4.2 Stability Analysis

In this section, we analyze the stability of the closed-loop system derived in the previous section. To perform this analysis, we split the system into several connected subsystems and analyze them as a cascade.

We begin by analyzing the subsystem given by (6.44) and (6.42).

Lemma 6.1. *The origin,* $\left[\widetilde{\mathbf{P}}_b^{\mathrm{T}}, \widetilde{\boldsymbol{\Sigma}}^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$*, is a globally exponentially stable (GES) equilibrium of the subsystem* (6.44)*,* (6.42) *if the communications graph* $\mathcal{G}$ *is connected and the control gains* k_b *and* k_f *are chosen such that* $k_b, k_f > \frac{3U_d}{\Delta}$*.*

Proof. From (6.44) and (6.42), this subsystem can be written in the following form

$$\begin{bmatrix}\dot{\widetilde{\mathbf{P}}}_b\\ \dot{\widetilde{\mathbf{\Sigma}}}\end{bmatrix} = (\mathbf{F}\otimes\mathbf{I}_3)\begin{bmatrix}\widetilde{\mathbf{P}}_b\\ \widetilde{\mathbf{\Sigma}}\end{bmatrix} + \begin{bmatrix}\frac{1}{N}\mathbf{L}_F\otimes\mathbf{I}_3\\ \frac{1}{N}\mathbf{L}_F\otimes\mathbf{I}_3\end{bmatrix}\widetilde{\mathbf{V}}_{\mathrm{LOS}}, \tag{6.47}$$

where

$$\mathbf{F} = \begin{bmatrix}-\left(k_b\mathbf{I}_N + \frac{k_f}{N}\mathbf{1}_N + c_p\mathbf{L}\right) & k_b\mathbf{I}_N\\ -\frac{k_f}{N}\mathbf{L}_F & -k_f\mathbf{I}_N\end{bmatrix}. \tag{6.48}$$

Due to its structure, $\mathbf{F}$ is a negative definite matrix. Let $\lambda_{\min}$ denote the real part of the smallest eigenvalue of $-\mathbf{F}$. Note that $\lambda_{\min} \geq \min\{k_b, k_f\}$.

Consider the following Lyapunov function candidate

$$V_p = \frac{1}{2}\left(\left\|\widetilde{\mathbf{P}}_b\right\|^2 + \left\|\widetilde{\mathbf{\Sigma}}\right\|^2\right). \tag{6.49}$$

The derivative of V_p along the trajectories of (6.47) is given by

$$\dot{V}_p = \begin{bmatrix}\widetilde{\mathbf{P}}_b^{\mathrm{T}} & \widetilde{\mathbf{\Sigma}}^{\mathrm{T}}\end{bmatrix}(\mathbf{F}\otimes\mathbf{I}_3)\begin{bmatrix}\widetilde{\mathbf{P}}_b\\ \widetilde{\mathbf{\Sigma}}\end{bmatrix} + \begin{bmatrix}\widetilde{\mathbf{P}}_b^{\mathrm{T}} & \widetilde{\mathbf{\Sigma}}^{\mathrm{T}}\end{bmatrix}\begin{bmatrix}\frac{1}{N}\mathbf{L}_F\otimes\mathbf{I}_3\\ \frac{1}{N}\mathbf{L}_F\otimes\mathbf{I}_3\end{bmatrix}\widetilde{\mathbf{V}}_{\mathrm{LOS}} \tag{6.50}$$

By applying the inequality (6.31), we get

$$\begin{aligned}\dot{V}_p &\leq -\lambda_{\min}\left(\left\|\widetilde{\mathbf{P}}_b\right\|^2 + \left\|\widetilde{\mathbf{\Sigma}}\right\|^2\right) + \frac{2U_d}{\Delta}\left\|\widetilde{\mathbf{P}}_b\right\|\left(\left\|\widetilde{\mathbf{P}}_b\right\| + \left\|\widetilde{\mathbf{\Sigma}}\right\|\right)\\ &\leq -\lambda_{\min}\left(\left\|\widetilde{\mathbf{P}}_b\right\|^2 + \left\|\widetilde{\mathbf{\Sigma}}\right\|^2\right) + \frac{2U_d}{\Delta}\left(\frac{3}{2}\left\|\widetilde{\mathbf{P}}_b\right\|^2 + \frac{1}{2}\left\|\widetilde{\mathbf{\Sigma}}\right\|^2\right).\end{aligned} \tag{6.51}$$

From (6.51), we can conclude that $\dot{V}_p$ is negative definite, and the subsystem is thus GES, if $\lambda_{\min} \geq \min\{k_b, k_f\} > \frac{3U_d}{\Delta}$. □

In the remainder of this section, we use the results from the cascaded systems theory. Specifically, we use Proposition 9 from [70], which, for the sake of convenience, is re-stated in Section 2.6. Now, let us analyze the subsystem (6.46).

Lemma 6.2. *The origin, $\widetilde{\mathbf{S}} = \mathbf{0}$, is a USGES equilibrium of the subsystem* (6.46) *if $k_b, k_f > \frac{3U_d}{\Delta}$, and the consensus gain c_s is chosen such that $c_s\lambda_2 > 2U_d k_s$, where λ_2 is the Fiedler eigenvalue of $\mathbf{L}$.*

Proof. Recall the equation for $\dot{\widetilde{\mathbf{S}}}$ from (6.46)

$$\dot{\widetilde{\mathbf{S}}} = -c_s\mathbf{L}\widetilde{\mathbf{S}} + \frac{1}{N}\mathbf{L}_F\mathbf{G}, \qquad \mathbf{G} = [g_1, \ldots, g_N]^{\mathrm{T}}. \tag{6.52}$$

We intend to analyze this subsystem as a cascade where $\widetilde{\mathbf{P}}_b$ perturbs the dynamics of $\widetilde{\mathbf{S}}$ through $\mathbf{G}$. The following equation describes the *nominal* dynamics of $\widetilde{\mathbf{S}}$

$$\dot{\widetilde{\mathbf{S}}} = -c_s\mathbf{L}\widetilde{\mathbf{S}} + \frac{1}{N}\mathbf{L}_F\mathbf{G}_s, \tag{6.53}$$

where $\mathbf{G}_s$ is obtained by substituting $\widetilde{\mathbf{P}}_b = 0$ into $\mathbf{G}$. From (6.35) the following inequality holds true for $\mathbf{G}_s$

$$\|\mathbf{G}_s\| \leq U_d k_s \left\|\widetilde{\mathbf{S}}\right\|. \tag{6.54}$$

Consider then the following Lyapunov function candidate for the nominal system

$$V_s = \frac{1}{2}\left\|\widetilde{\mathbf{S}}\right\|^2. \tag{6.55}$$

The derivative of V_s along the trajectories of (6.53) is

$$\dot{V}_s = -c_s \widetilde{\mathbf{S}}^{\mathrm{T}} \mathbf{L} \widetilde{\mathbf{S}} + \frac{1}{N} \widetilde{\mathbf{S}}^{\mathrm{T}} \mathbf{L}_F \mathbf{G}_s \leq (-c_s \lambda_2 + 2U_d k_s) \left\|\widetilde{\mathbf{S}}\right\|^2. \tag{6.56}$$

From (6.56), we can conclude that $\dot{V}_s$ is negative definite, and the nominal system is thus GES, if $c_s \lambda_2 > 2U_d k_s$.

Note that both the nominal and the perturbing system are GES. GES implies both UGAS and USGES. Consider the Lyapunov function candidate V_s. The first assumption in [70, Proposition 9] is satisfied with $c_1 = \frac{1}{2}$, an arbitrary $\eta > 0$, and $c_2 = \eta$. The second assumption in [70, Proposition 9] is satisfied with $\alpha_1 = U_d\left(\frac{1}{\Delta} + k_s\right)$, and $\alpha_2 = 0$. Consequently, all assumptions of [70, Proposition 9] are satisfied, and the cascaded system is USGES. □

Finally, let us analyze the subsystem (6.38).

Lemma 6.3. *The origin, $\mathbf{p}_b^p = \mathbf{0}$, is a USGES equilibrium of the subsystem* (6.38) *if $k_b, k_f > \frac{3U_d}{\Delta}$ and $c_s \lambda_2 > 2U_d k_s$.*

Proof. Similarly to the proof of the previous lemma, this subsystem can be analyzed as a cascade where $\widetilde{\mathbf{S}}$ and $\widetilde{\mathbf{P}}_b$ perturb the dynamics of $\mathbf{p}_b^p$ through $\widetilde{\mathbf{V}}_{\mathrm{LOS}}$, $\widetilde{\mathbf{P}}_b$, and $\mathbf{G}$. Consider the following nominal system

$$\dot{\mathbf{p}}_b^p = -U_d \left[k_s f_s\left(x_b^p\right), \frac{1}{D} y_b^p, \frac{1}{D} z_b^p\right]^{\mathrm{T}}. \tag{6.57}$$

This type of system has been proven USGES in Chapter 5, Theorem 5.1.

Consider then the following Lyapunov function candidate

$$V_b = \frac{1}{2}\left\|\mathbf{p}_b^p\right\|^2. \tag{6.58}$$

Similarly to the previous lemma, the first assumption in [70, Proposition 9] is satisfied with $c_1 = \frac{1}{2}$, an arbitrary $\eta > 0$, and $c_2 = \eta$.

Let $\mathbf{h}$ denote the perturbing term in (6.38). This term is given by

$$\mathbf{h} = \frac{1}{N} \sum_{i=1}^{N} \left(\mathbf{R}_p^{\mathrm{T}} \left(\widetilde{\mathbf{v}}_{\mathrm{LOS},i} + k_f \widetilde{\mathbf{p}}_{b,i}\right) - g_i\right). \tag{6.59}$$

From (6.31) and (6.35), we arrive at the following upper bound on the norm of $\mathbf{h}$

$$\begin{aligned}\|\mathbf{h}\| &\leq \left(2\frac{U_d}{\Delta} + k_f + U_d k_s\right)\left\|\widetilde{\mathbf{P}}_b\right\| + U_d k_s \left\|\widetilde{\mathbf{S}}\right\| \\ &\leq \left(2\frac{U_d}{\Delta} + k_f + U_d k_s\right)\left\|\left[\widetilde{\mathbf{P}}_b^{\mathrm{T}}, \widetilde{\mathbf{S}}^{\mathrm{T}}\right]\right\|. \end{aligned} \tag{6.60}$$

Consequently, the second assumption in [70, Proposition 9] is satisfied with $\alpha_1 = \left(2\frac{U_d}{\Delta} + k_f + U_d k_s\right)$, $\alpha_2 = 0$, and the subsystem is thus USGES. □

The results of the three lemmas can be summarized in the following theorem.

Theorem 6.1. *The origin,* $\left[(\mathbf{p}_b^p)^{\mathrm{T}}, \widetilde{\mathbf{\Sigma}}^{\mathrm{T}}, \widetilde{\mathbf{P}}_b^{\mathrm{T}}, \widetilde{\mathbf{S}}^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$*, is a uniformly semiglobally exponentially stable (USGES) equilibrium of the closed-loop system* (6.38)*,* (6.42)*,* (6.44)*,* (6.46) *if* $k_b, k_f > \frac{3U_d}{\Delta}$ *and* $c_s\lambda_2 > 2U_d k_s$*.*

Proof. The stability of the complete system can be proven by applying the results from Lemmas 6.1—6.3. □

6.5 Application to Underactuated AUVs

To apply the proposed distributed NSB algorithm to underactuated AUVs, we need to resolve two issues. Firstly, unlike vehicles with single-integrator dynamics (6.1), the position of AUVs cannot be controlled directly. Specifically, due to underactuation, there is no input to directly control the sway and heave dynamics, so these states must be controlled indirectly through the surge, pitch, and yaw dynamics. Secondly, in real-life situations, the continuous-time consensus algorithm (6.22), (6.23) cannot be implemented, as the vehicles communicate at discrete time instances. In fact, the limited bandwidth of most underwater communication methods (*e.g.,* acoustics) introduce a significant delay and long periods between transmissions. In this section, we introduce the model of underactuated AUVs and propose solutions to the two aforementioned problems.

6.5.1 AUV Model

Let $\mathbf{p}_i \in \mathbb{R}^3$ denote the position of vehicle i. Let $\mathbf{R}_i \in SO(3)$ be a rotation matrix that describes the orientation. The kinematics of the vehicle are [91]

$$\dot{\mathbf{p}}_i = \mathbf{R}_i \boldsymbol{\upsilon}_i, \qquad \dot{\mathbf{R}}_i = \mathbf{R}_i \mathbf{S}(\boldsymbol{\omega}_i). \tag{6.61}$$

Similarly to the previous chapter, we assume that the AUVs are equipped with a low-level controller that can track a desired surge velocity and orientation. The vehicle dynamics can then be expressed as

$$\begin{bmatrix}\dot{\boldsymbol{\upsilon}}_i \\ \dot{\boldsymbol{\omega}}_i\end{bmatrix} = f\left(\mathbf{R}_i, \mathbf{R}_{d,i}, \boldsymbol{\upsilon}_i, u_{d,i}, \boldsymbol{\omega}_i\right), \tag{6.62}$$

where f is a function representing the closed-loop dynamics, and $\mathbf{R}_{d,i}$ and $u_{d,i}$ are the desired orientation and surge velocity, respectively.

6.5.2 Handling the Underactuated Dynamics

In this section, we propose a method for choosing the desired surge velocity and orientation in order to handle the underactuated dynamics.

First, let us discuss the method for choosing the desired surge velocity. Ideally, $u_{d,i}$ should be chosen such that $\|[u_{d,i}, v_i, w_i]\| = \|\mathbf{v}_{\mathrm{NSB},i}\|$. However, as discussed in the previous chapter, this equation cannot always be satisfied. In the previous chapter, we solved this problem by scaling the LOS velocity. However, such scaling must be done globally for all vehicles, which would require additional communications and consensus. Consequently, we propose the following simpler method for choosing the desired surge velocity

$$u_{d,i} = \begin{cases} \sqrt{U_{\mathrm{NSB},i}}, & \text{if } U_{\mathrm{NSB},i} \geq u_{\min}^2, \\ u_{\min}, & \text{otherwise,} \end{cases} \tag{6.63}$$

where $U_{\mathrm{NSB},i} = \|\mathbf{v}_{\mathrm{NSB},i}\|^2 - v_i^2 - w_i^2$, and $u_{\min} > 0$ is the minimum required surge velocity.

Now, let us discuss the method for choosing the desired orientation. In Chapter 4, we used a method that compensates for the sideslip angle and the angle of attack. In Chapter 5, we proposed to find the desired orientation indirectly by first finding the desired angular rates. Both methods require accurate estimates of the underactuated sway and heave velocities. However, in real-life applications, these estimates are often noisy, which can deteriorate the performance of the controller. Therefore, we propose a method with integral actions that provide some robustness towards measurement noise. The proposed method consists of two steps. In the first step, we find the desired course, $\chi_{d,i}$, and flight-path angle, $\gamma_{d,i}$, as

$$\chi_{d,i} = \arctan_2\left(y_{\mathrm{NSB},i}, x_{\mathrm{NSB},i}\right), \tag{6.64}$$

$$\gamma_{d,i} = -\arcsin\left(\frac{z_{\mathrm{NSB},i}}{\|\mathbf{v}_{\mathrm{NSB},i}\|}\right). \tag{6.65}$$

In the second step, we calculate the desired pitch ($\theta_{d,i}$) and yaw ($\psi_{d,i}$) angles

$$\theta_{d,i} = \mathrm{sat}_{\theta_{\max}}\left(\gamma_{d,i} - k_\gamma \int_0^t \widetilde{\gamma}_i(\tau)\mathrm{d}\tau\right), \tag{6.66}$$

$$\psi_{d,i} = \chi_{d,i} - k_\chi \int_0^t \widetilde{\chi}_i(\tau)\mathrm{d}\tau, \tag{6.67}$$

where k_γ and k_χ are positive gains, $\mathrm{sat}(\cdot)$ is a saturation function

$$\mathrm{sat}_{x_{\max}}(x) = \begin{cases} -x_{\max}, & \text{if } x < -x_{\max}, \\ x, & \text{if } -x_{\max} \leq x \leq x_{\max}, \\ x_{\max}, & \text{if } x > x_{\max}, \end{cases} \tag{6.68}$$

and $\widetilde{\gamma}_i(\tau)$ and $\widetilde{\chi}_i(\tau)$ are the flight-path angle and course errors

$$\widetilde{\gamma}_i = \gamma_i - \gamma_{d,i} = \theta_i - \arctan\left(\frac{w_i}{u_i}\right) - \gamma_{d,i}, \tag{6.69}$$

$$\widetilde{\chi}_i = \chi_i - \chi_{d,i} = \psi_i + \arctan\left(\frac{v_i}{u_i}\right) - \chi_{d,i}. \tag{6.70}$$

The purpose of the saturation function is to avoid singularities introduced by the Euler angles. Consequently, the saturation limit must satisfy $\theta_{\max} < \frac{\pi}{2}$.

The desired orientation can then be calculated as

$$\mathbf{R}_{d,i} = \begin{bmatrix} \cos\psi_{d,i}\cos\theta_{d,i} & -\sin\psi_{d,i} & \cos\psi_{d,i}\sin\theta_{d,i} \\ \cos\theta_{d,i}\sin\psi_{d,i} & \cos\psi_{d,i} & \sin\psi_{d,i}\sin\theta_{d,i} \\ -\sin\theta_{d,i} & 0 & \cos\theta_{d,i} \end{bmatrix}. \tag{6.71}$$

6.5.3 Handling Discrete-Time Communications

In this section, we discuss the issue of discrete-time delayed communications. We assume that the AUVs can broadcast data via an acoustic modem. However, due to the relatively low bitrate and propagation speed of underwater acoustics, there is a significant delay between transmission and reception. To model the communications, we assume that when vehicle i transmits a packet at a time t, vehicle j receives the packet at a time $t + \Delta T_{ij}$, where $\Delta T_{ij} > 0$ is the delay. Since the acoustic communications channel has a range of several kilometers [118], we can assume that all vehicles can recieve the transmitted packet (in other words, that the communications graph is fully connected). To address the issues with discrete-time communications, delays, and limited bandwidth, we need to discretize the consensus scheme (6.22), (6.23), compensate for the delays, and reduce the number of transmissions.

To discretize the consensus scheme, we propose the following method. Each vehicle continuously updates its variables using the following equations

$$\dot{\mathbf{p}}_{b,i} = \mathbf{v}_{\mathrm{LOS},i} + k_b\left(\widehat{\boldsymbol{\sigma}}_i - \widehat{\boldsymbol{\sigma}}_{d,i}\right), \tag{6.72}$$

$$\dot{s}_i = U_d \left\| \frac{\partial \mathbf{p}_p(s_i)}{\partial s_i} \right\|^{-1} \left(\frac{\Delta}{D_i} + k_s f_s\left(x^p_{b,i}\right) \right), \tag{6.73}$$

$$\dot{r}_{f,i} = k_r\left(\widehat{r}_{f,i} - r_{f,i}\right), \qquad r^+_{f,i} = \widehat{r}_{f,i}, \text{if } \widehat{r}_{f,i} > r_{f,i}. \tag{6.74}$$

To perform the consensus algorithm, each AUV transmits the following packet: $(t_i, s_i, \mathbf{p}_{b,i}, r_{f,i}, \breve{\boldsymbol{\sigma}}_i)$, where t_i is the time of transmission, and $\breve{\boldsymbol{\sigma}}_i = \widehat{\boldsymbol{\sigma}}_i - \widehat{\boldsymbol{\sigma}}_{d,i}$. When vehicle i receives a consensus packet from vehicle j, it first compensates for the delay by solving the following set of differential equations

$$\dot{\mathbf{p}}_{b,j} = \mathbf{v}_{\mathrm{LOS},j} + k_b \breve{\boldsymbol{\sigma}}_j, \tag{6.75a}$$

$$\dot{\breve{\boldsymbol{\sigma}}}_j = -k_f \breve{\boldsymbol{\sigma}}_j, \tag{6.75b}$$

$$\dot{s}_j = U_d \left\| \frac{\partial \mathbf{p}_p(s_j)}{\partial s_j} \right\|^{-1} \left(\frac{\Delta}{D_j} + k_s f_s\left(x^p_{b,j}\right) \right). \tag{6.75c}$$

Note that (6.75a) and (6.75c) are analogous to (6.72) and (6.73). The equation for $\dot{\breve{\boldsymbol{\sigma}}}_j$ is motivated by the fact that $\breve{\boldsymbol{\sigma}}_j$ is used as the formation-keeping error (*c.f.,* (6.19)). Under ideal conditions (*i.e.,* if the consensus errors are zero), $\breve{\boldsymbol{\sigma}}_j$ evolves according to (6.75b). The value of $r_{f,j}$ is kept constant.

After compensating for the delay, the vehicle updates its consensus variables using the following scheme

$$\mathbf{p}_{b,i}^{+} = a_p \mathbf{p}_{b,j} + (1 - a_p)\, \mathbf{p}_{b,i}, \tag{6.76}$$

$$s_i^{+} = a_s s_j + (1 - a_s)\, s_i, \tag{6.77}$$

$$r_{f,i}^{+} = \max\left\{r_{f,i}, r_{f,j}\right\}, \tag{6.78}$$

where $a_p, a_s \in (0, 1)$ are the mixing gains.

Finally, let us discuss the method for determining when the AUVs should transmit the consensus packets. To minimize the number of transmissions, we propose an event-triggered scheme. When vehicle i receives a consensus packet from vehicle j, it also calculates the following quantity:

$$T_{j,i} = T_{\min} + \frac{1}{\sqrt{b_p \left\|\mathbf{p}_{b,i} - \mathbf{p}_{b,j}\right\|^2 + b_s \left(s_i - s_j\right)^2}}, \tag{6.79}$$

where $T_{\min}$, b_p, and b_s are positive constants. Let us assume that the last transmission of vehicle i was at a time t_i. The next transmission time is then calculated as $t_i + \max_j T_{j,i}$. This event triggering scheme, inspired by the techniques discussed in [119], was chosen because it is easy to implement and has a strictly defined minimum delay in-between transmission, making it suitable in applications with limited bandwidth and communication delays. We demonstrate that the proposed scheme works both in simulations and field trials. However, proving the effectiveness of the scheme is not the main focus of this work.

6.6 Simulations

In this section, we present the results of a numerical simulation study. The simulations were carried out in MATLAB using a model of the light autonomous underwater vehicle (LAUV) [12]. We conducted two types of simulations. We tested the continuous-time NSB algorithm proposed in Section 6.3. Furthermore, the performance of the event-triggered NSB algorithm proposed in Section 6.5.3 is compared to the performance of a cooperative path following algorithm proposed in [25].

First, let us present the parameters that are common to both simulations. We simulate four AUVs. The barycenter should follow an elliptic path given by

$$\mathbf{p}_p(s) = \left[a\cos(s), b\sin(s), c\sin(s)^2\right]^{\mathrm{T}}, \tag{6.80}$$

where $a = 60\,\mathrm{m}$, $b = 40\,\mathrm{m}$, $c = 10\,\mathrm{m}$. The desired path-following speed is $U_d = 1.3\,\mathrm{m\,s^{-1}}$. The lookahead distance is chosen as $\Delta = 5\,\mathrm{m}$. The shape of the desired formation is

$$\left[\mathbf{p}_{f,1}^{f}, \mathbf{p}_{f,2}^{f}, \mathbf{p}_{f,3}^{f}, \mathbf{p}_{f,4}^{f}\right] = \begin{bmatrix} 10 & -10 & 0 & 0 \\ 0 & 0 & 10 & -10 \\ 0 & -4 & 4 & 0 \end{bmatrix}. \tag{6.81}$$

To satisfy the assumptions in Lemma 6.1, k_b and k_f must be greater than $\frac{3U_d}{\Delta} = 0.78$. We choose $k_b = k_f = 1$. The gain of the path parameter update law is $k_s = 0.5$. The adjacency matrix of the communication graph is

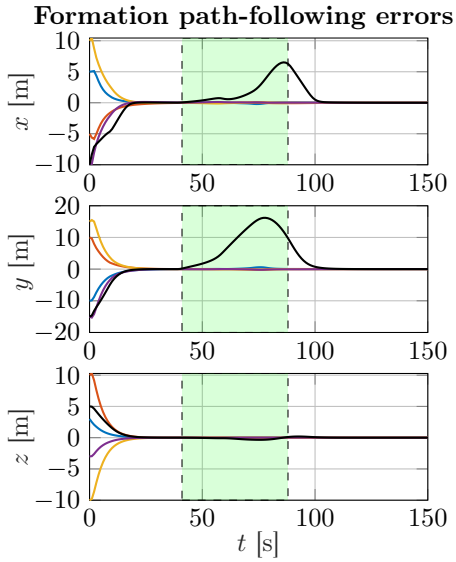
$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}. \tag{6.82}$$

6.6.1 Continuous-Time Consensus

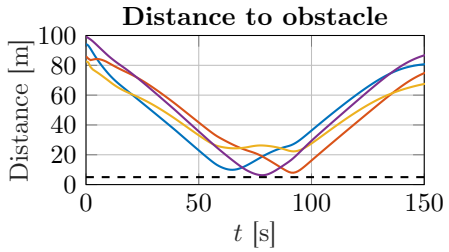
The purpose of this simulation is to demonstrate that the path-following, formation-keeping, and consensus errors of the continuous-time scheme converge to zero. To satisfy the assumptions of Lemma 6.2, the consensus gain c_s must be greater than $\frac{2U_d k_s}{\lambda_2} = 0.65$. We choose $c_s = c_p = 1$. To test the obstacle avoidance scheme, we place a static obstacle at $\mathbf{p}_o = [0, 40]^{\mathrm{T}}$ with radius $r_o = 5\,\mathrm{m}$. The minimum cone $\alpha_{\min}$ is set to 2 degrees, and the formation radius update gain is $k_r = 0.1$.

The results are shown in Figure 6.2. Figure 6.3 shows the 3D trajectory of the AUVs. We can see that the AUVs converge to their desired formation while avoiding the obstacle represented by the green cylinder. Figure 6.2a shows the path-following and formation-keeping errors. Initially, these errors exponentially converge to zero. When the obstacle avoidance scheme is activated, the path-following errors start diverging, as the LOS velocity enters the collision cone. After the vehicles successfully avoid the obstacle, the errors again converge exponentially to zero. Figure 6.2b shows the distance between the AUVs and the obstacle. We can see that the distance is always greater than r_o. Figs. 6.2c, 6.2d, and 6.2e show the errors of the consensus variables. These plots are in a logarithmic scale to demonstrate the exponential rate of convergence. Initially, the logarithmic error is clearly bounded by a decreasing straight line, demonstrating the exponential convergence. The norm of the error decreases by a factor of 10 approximately every 25 seconds. When obstacle avoidance is active, the errors start diverging but remain bounded. After avoiding the obstacle, the errors continue to decrease exponentially, but eventually, the convergence stagnates. We cannot expect that the errors continue to fall indefinitely due to numerical innacuracies, that come mostly from two sources: the inaccuracies in floating-point arithmetics and the tolerances of the ODE solver.

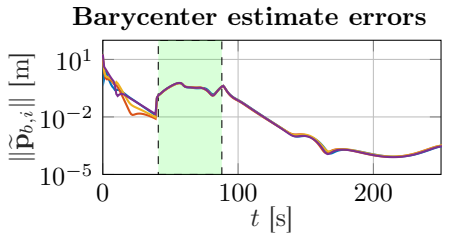
Note that although Section 6.4 presents stability proof for a simplified case of straight-line paths and vehicles with single-integrator dynamics, the simulation results show stability for curved paths and more complex vehicle models. It may be possible to extend the results of Section 6.3 to curved paths and more complex vehicle models by assuming that the curvature of the desired path is small enough and that the low-level control system is capable of exponential tracking, similarly to the proofs in Chapters 4 and 5.



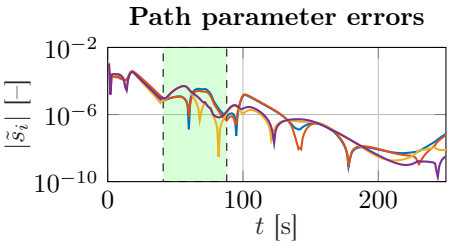
(a) The path-following and formation-keeping errors. The green area represents the time when obstacle avoidance is active. The black line is the path-following error, the colored lines represent the formation-keeping errors.



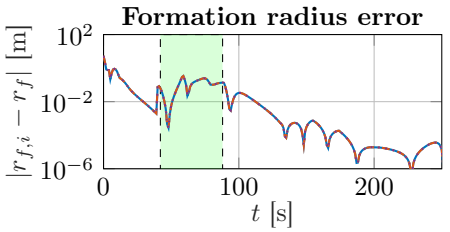
(b) The horizontal distance between the AUVs and the obstacle. The dashed line is the minimum safety distance.



(c) The norm of barycenter estimate errors, plotted in a logarithmic scale.



(d) The absolute value of path parameter errors, plotted in a logarithmic scale



(e) The difference between the estimated and true formation radius, plotted in a logarithmic scale.

Figure 6.2: Results of numerical simulations. The blue, red, yellow, and purple lines represent AUV 1, 2, 3, and 4, respectively.

6.6.2 Event-Triggered Consensus

In this section, we test the event-triggered scheme proposed in Section 6.5.3. We choose the mixing gains $a_p = a_s = 0.4$, and the penalty gains: $b_p = 10^{-4}, b_s = 4$.

We compare our algorithm with an event-triggered cooperative path-following algorithm proposed in [25]. In this algorithm, each vehicle follows its own desired path given by $\mathbf{p}_{d,i}(s) = \mathbf{p}_p(s) + \mathbf{R}_p(s)\mathbf{p}^f_{f,i}$. Coordination is then achieved by running a consensus scheme on the path parameter.

The comparison was done using a Monte Carlo simulation. We performed ten thousand simulations with randomly selected initial conditions and communication delays. The initial positions of the AUVs ranged from $[60, -40, 0]^{\mathrm{T}}$ to $[140, 40, 15]^{\mathrm{T}}$.

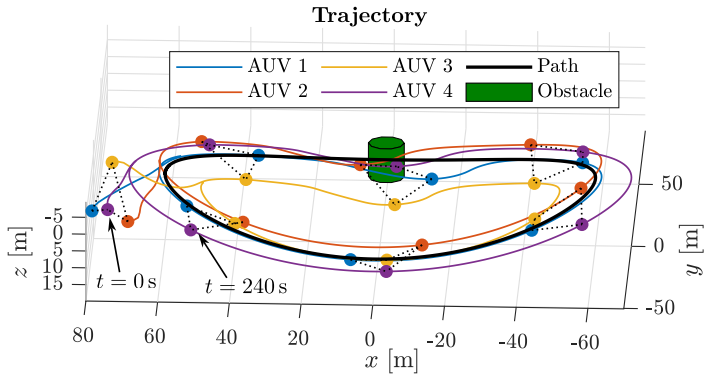


Figure 6.3: The 3D trajectory of the AUVs. The markers represent the AUVs at times $t = 0, 40, \ldots, 240$ seconds. The dotted lines represent the communications graph.

The initial orientations were specified in Euler angles, with a zero roll angle, a pitch angle from $-\frac{\pi}{8}$ to $\frac{\pi}{8}$, and a yaw angle from 0 to π. The initial linear velocities ranged from $[0.5, -0.2, -0.1]^{\mathrm{T}}$ to $[1.5, 0.2, 0.1]^{\mathrm{T}}$, and the initial angular velocities were zero. The communication delays ranged from 0 to 5 seconds.

Figure 6.4 shows the absolute value of the path parameter error and the norm of the path-following error. Both errors are plotted in a logarithmic scale. In terms of path parameter errors, both algorithms perform similarly. In terms of path-following errors, the distributed NSB algorithm is marginally better.

The communication requirements of the two algorithms are summarized in Table 6.1. This table shows the minimum, maximum, and median of the period in-between transmissions (τ_t), and the total number of transmissions in one simulation (N_t). Here, we can see that distributed NSB performs considerably better in comparison to the cooperative path following method, with longer periods in-between transmissions and a lower number of transmissions.

However, it is worth mentioning that the packets transmitted by the cooperative path following method are smaller than the NSB packets. Indeed, in the scheme

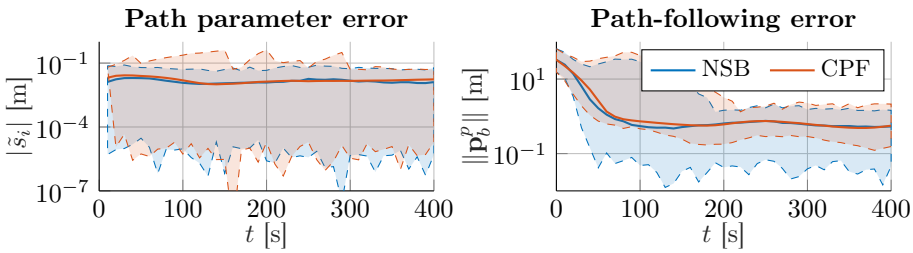


Figure 6.4: Comparison between the proposed event-triggered distributed NSB algorithm and the cooperative path-following algorithm proposed in [25]. The full lines represent the median, the colored areas represent the smallest and largest recorded value.

proposed in [25], the AUVs only need to transmit the path parameter and its derivative. In contrast, the NSB packet consists of the estimates of path parameter, barycenter, radius of formation, and formation-keeping error. In this context, the two algorithms present a trade-off between packet size and communication periods.

Table 6.1: Comparison of communication requirements.

Method	Quantity	Minimum	Median	Maximum
Distributed	τ_t	2.00	8.50	275.30
NSB	N_t	125	187	253
Coordinated	τ_t	0.10	2.15	138.65
path following	N_t	196	548	1064

6.7 Experiments

In this section, we present the results of field experiments we designed and executed to verify the effectiveness of the event-triggered distributed NSB algorithm proposed in Section 6.5.3. The experiments were conducted in the fjord of Trondheim, Norway, near the Trondheim Biological Station, using two LAUVs as in Figure 6.5.

To guarantee stable communications and accurate navigation, the vehicles were operating at the surface and communicating over WiFi. The algorithm was implemented in C++, using the Unified Navigation Environment (DUNE) [89].

The algorithm was tested in two scenarios: a *nominal* scenario, *i.e.,* formation path following without any obstacles, and a scenario with a static obstacle. In both scenarios, the barycenter of the AUVs should follow the elliptic path

$$\mathbf{p}_p(s) = [a\cos(s), b\sin(s), 0]^{\mathrm{T}}, \qquad a = 100\,\mathrm{m}, \qquad b = 80\,\mathrm{m}, \tag{6.83}$$

in the formation defined by

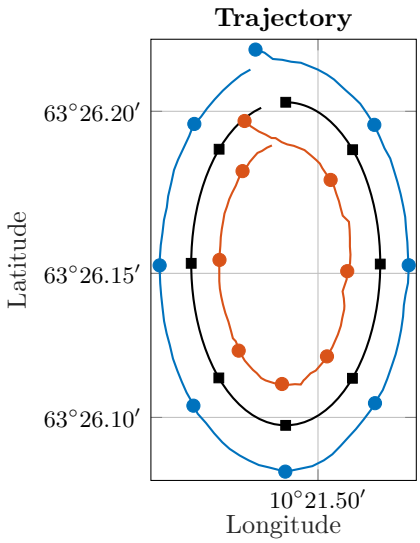
$$\mathbf{p}^f_{f,1} = [0, -25, 0]^{\mathrm{T}}, \qquad \mathbf{p}^f_{f,2} = [0, 25, 0]^{\mathrm{T}}. \tag{6.84}$$

The reason for choosing a larger path and formation, compared to the simulations in Section 6.6, was to reduce the risk of the AUVs colliding. The obstacle was placed at $\mathbf{p}_o = [0, 80]^{\mathrm{T}}$. The remaining parameters were identical to the simulations.

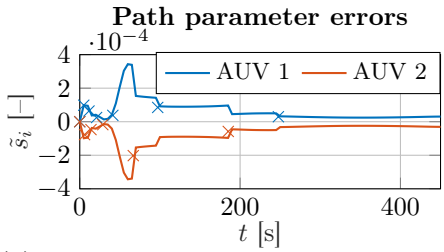
6.7.1 Nominal Scenario

The results are shown in Figure 6.6. Figure 6.6a shows the trajectories of the AUVs, as estimated by their onboard navigation system. The disturbances in the trajectories are caused by two factors: the sea loads (waves, currents, and wind), and the errors of the navigation system. However, the exponential stability of the NSB algorithm given by Theorem 6.1 provides some robustness towards these disturbances, *c.f.,* [120, Lemma 9.2]. Figure 6.6b shows the path parameter errors and the event-triggered communications. Initially, the vehicles need to communicate frequently, approximately every five seconds, because the barycenter estimates

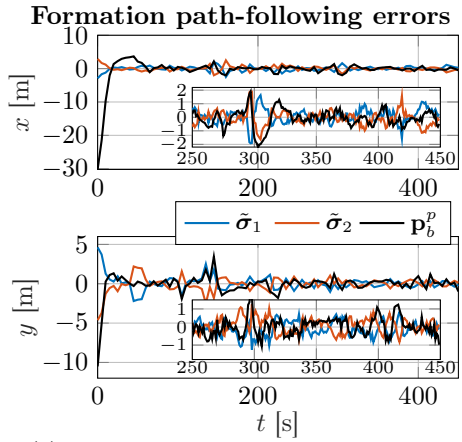
Figure 6.5: Photo of one of the two LAUVs used in the reported field experiments, courtesy of `www.ntnu.edu/aur-lab/`.



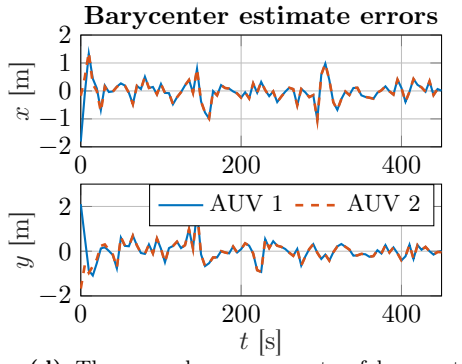
(a) The trajectory of the AUVs. The markers represent the AUVs at times $t = 0, 60, \ldots, 420$ seconds.



(b) The path parameter errors. The markers represent transmissions.



(c) The path-following and formation-keeping errors.



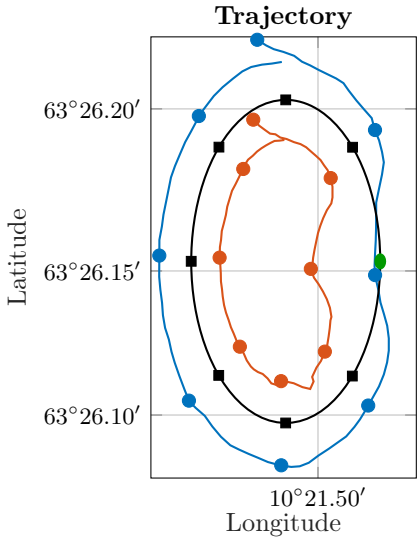
(d) The x- and y-components of barycenter estimate errors.

Figure 6.6: Results of a nominal experiment.

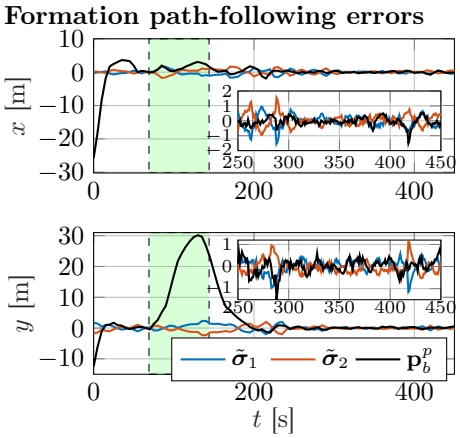
differ (as seen in Figure 6.6d). During this transient period, the path parameter estimates initially diverge before finally converging. After convergence, the communication period increases to over 100 seconds. Note that the barycenter estimate errors in Figure 6.6d converge to a common value but not to zero. This behavior is caused by the aforementioned disturbances acting on the vehicles. Figure 6.6c shows the formation-keeping and path-following errors. Due to the disturbances and event-triggered communications, these errors do not converge to zero but rather to a small area near zero.

6.7.2 Scenario with a Static Obstacle

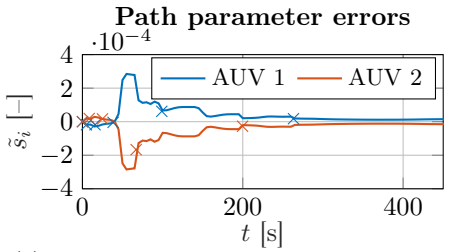
The results are shown in Figure 6.7. In general, the results are similar to the nominal scenario, so we will only highlight the differences. The path-following errors in Figure 6.7b diverge when obstacle avoidance is active. As previously mentioned, this behavior is caused by the fact that the vehicles cannot stay on the desired path while avoiding the obstacle. The estimate errors in Figs. 6.7c and 6.7d behave similarly to the nominal case. As shown in Figure 6.7e, the distance between the AUVs and the obstacle is always greater than r_o. Figure 6.7f shows the errors between the estimated and true formation radius. Note that the formation radius errors are connected to the barycenter estimates. A wrong barycenter estimate may lead to both overestimation and underestimation of the formation radius. Despite these uncertainties, the AUVs still manage to perform all control goals safely.



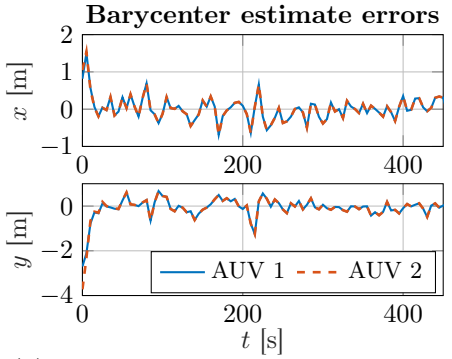
(a) The trajectory of the AUVs. The markers represent the AUVs at times $t = 0, 60, \ldots, 420$ seconds.



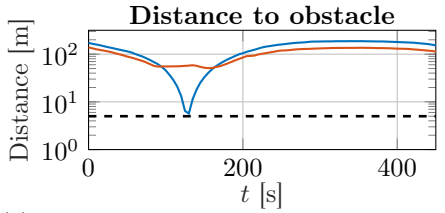
(b) The path-following and formation-keeping errors. The green area represents the time when obstacle avoidance is active.



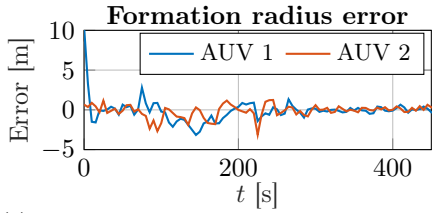
(c) The path parameter errors. The markers represent transmissions.



(d) The x- and y-components of barycenter estimate errors.



(e) The distance between the AUVs and the obstacle.



(f) The error between the estimated and true formation radius.

Figure 6.7: Experiment with a static obstacle.

Part II

Hand Position and its Applications

Chapter 7

Hand Position for Underactuated Underwater Vehicles

This chapter motivates and defines the hand position concept. Compared to previous works that utilize this concept, our approach works on six-degree-of-freedom vehicles and does not introduce singularities. By choosing the hand position as the output of the controlled system, we can apply output feedback linearization to simplify the dynamics of the vehicle. Specifically, we can then transform the six-degree-of-freedom nonlinear underactuated vehicle model into a double integrator. This transformation enables the use of numerous control strategies that could otherwise not be used on nonholonomic or underactuated vehicles. The contents of this chapter are based on [87].

The chapter is organized as follows. Section 7.1 presents the AUV model. Section 7.2 defines the hand position transformation and presents the necessary assumptions about the generic hand position-based controller. The closed-loop system is then analyzed in Section 7.3.

7.1 AUV Model

We consider an underactuated AUV with dynamics described using the 6DOF control-oriented model from Section 2.1.4. The AUV model is given by the following equations

$$\dot{\mathbf{p}} = \mathbf{R}\boldsymbol{\upsilon}_r + \mathbf{V}_c, \tag{7.1a}$$

$$\dot{\mathbf{R}} = \mathbf{R}\mathbf{S}(\boldsymbol{\omega}), \tag{7.1b}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}}_r + \mathbf{C}(\boldsymbol{\nu}_r)\boldsymbol{\nu}_r + \mathbf{D}\boldsymbol{\nu}_r + \mathbf{g}(\mathbf{R}) = \mathbf{B}\mathbf{f}, \tag{7.1c}$$

In the remainder of this section, we introduce some necessary assumptions about the AUV and rewrite (7.1c) in a more compact form. To do so, let us first decompose $\mathbf{M}$, $\mathbf{M}^{-1}$, $\mathbf{C}(\boldsymbol{\nu}_r)$, and $\mathbf{D}$ into 3-by-3 blocks

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} \\ \mathbf{M}_{21} & \mathbf{M}_{22} \end{bmatrix}, \qquad \mathbf{C}(\boldsymbol{\upsilon}_r) = \begin{bmatrix} \mathbf{C}_{11}(\boldsymbol{\upsilon}_r) & \mathbf{C}_{12}(\boldsymbol{\upsilon}_r) \\ \mathbf{C}_{21}(\boldsymbol{\upsilon}_r) & \mathbf{C}_{22}(\boldsymbol{\upsilon}_r) \end{bmatrix}, \tag{7.2a}$$

$$\mathbf{M}^{-1} = \begin{bmatrix} \mathbf{M}'_{11} & \mathbf{M}'_{12} \\ \mathbf{M}'_{21} & \mathbf{M}'_{22} \end{bmatrix}, \qquad \mathbf{D} = \begin{bmatrix} \mathbf{D}_{11} & \mathbf{D}_{12} \\ \mathbf{D}_{21} & \mathbf{D}_{22} \end{bmatrix}. \tag{7.2b}$$

In addition to Assumptions 2.1–2.4 of the control-oriented model, we need to add one more simplifying assumption.

Assumption 7.1. *The effect of gravity and buoyancy on the linear velocities is negligible. Therefore, the following approximation*

$$\mathbf{M}^{-1}\mathbf{g}(\mathbf{R}) \approx \begin{bmatrix} \mathbf{0}_3 \\ \mathbf{M}'_{22}\left(Wz_{gb}\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right) \end{bmatrix}, \tag{7.3}$$

can be used to simplify the dynamics.

Remark. *The effect of gravity and buoyancy on the linear velocities is given by*

$$\mathbf{M}'_{12}\left(Wz_{gb}\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right) = \frac{Wz_{gb}\, m_{35}}{m_{33}m_{55} - m_{35}^2}\left[0, 0, \sin\theta\right]^{\mathrm{T}}, \tag{7.4}$$

where $\theta \in [-\pi/2, \pi/2]$ is the pitch angle of the vehicle. Assumption 7.1 can thus be used if θ remains small.

Remark. *Throughout the chapter, we will sometimes show expressions with Euler angles, because they are more intuitive than rotation matrices. This does not mean that we transform our model to Euler angles, these expressions are only used for illustration.*

We can then rewrite (7.1c) in the following compact form

$$\dot{\boldsymbol{\upsilon}}_r = \left[f_u, 0, 0\right]^{\mathrm{T}} - \mathcal{D}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}_r) - \mathcal{C}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}_r), \tag{7.5a}$$

$$\dot{\boldsymbol{\omega}} = \left[f_p, f_q, f_r\right]^{\mathrm{T}} - \mathcal{D}_{\boldsymbol{\omega}}(\boldsymbol{\nu}_r) - \mathcal{C}_{\boldsymbol{\omega}}(\boldsymbol{\nu}_r) - \mathbf{M}'_{22}\left(Wz_{gb}\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right), \tag{7.5b}$$

where

$$\mathcal{D}_{\boldsymbol{\upsilon}} = \left(\mathbf{M}'_{11}\mathbf{D}_{11} + \mathbf{M}'_{12}\mathbf{D}_{21}\right)\boldsymbol{\upsilon}_r + \left(\mathbf{M}'_{12}\mathbf{D}_{22} + \mathbf{M}'_{11}\mathbf{D}_{12}\right)\boldsymbol{\omega}, \tag{7.6a}$$

$$\mathcal{C}_{\boldsymbol{\upsilon}} = \left(\mathbf{M}'_{11}\mathbf{C}_{11} + \mathbf{M}'_{12}\mathbf{C}_{21}\right)\boldsymbol{\upsilon}_r + \left(\mathbf{M}'_{12}\mathbf{C}_{22} + \mathbf{M}'_{11}\mathbf{C}_{12}\right)\boldsymbol{\omega}, \tag{7.6b}$$

$$\mathcal{D}_{\boldsymbol{\omega}} = \left(\mathbf{M}'_{21}\mathbf{D}_{11} + \mathbf{M}'_{22}\mathbf{D}_{21}\right)\boldsymbol{\upsilon}_r + \left(\mathbf{M}'_{22}\mathbf{D}_{22} + \mathbf{M}'_{21}\mathbf{D}_{12}\right)\boldsymbol{\omega}, \tag{7.6c}$$

$$\mathcal{C}_{\boldsymbol{\omega}} = \left(\mathbf{M}'_{21}\mathbf{C}_{11} + \mathbf{M}'_{22}\mathbf{C}_{21}\right)\boldsymbol{\upsilon}_r + \left(\mathbf{M}'_{22}\mathbf{C}_{22} + \mathbf{M}'_{21}\mathbf{C}_{12}\right)\boldsymbol{\omega}. \tag{7.6d}$$

7.2 Hand Position

In this section, we present the hand position transformation for the 3D case. The procedure is inspired by the 2D transformation in [14]. We begin with the following change of coordinates:

$$\mathbf{x}_1 = \mathbf{p} + \mathbf{R}\boldsymbol{\ell}, \tag{7.7a}$$

$$\mathbf{x}_2 = \mathbf{R}\boldsymbol{\upsilon}_r + \mathbf{R}\left(\boldsymbol{\omega} \times \boldsymbol{\ell}\right), \tag{7.7b}$$

where $\boldsymbol{\ell} = \left[h, 0, 0\right]^{\mathrm{T}}$, where $h > 0$ is the *hand length*.

We will treat $\mathbf{x}_1$ as the output of our system, and perform an output feedback linearization. Differentiating (7.7) with respect to time yields:

$$\dot{\mathbf{x}}_1 = \mathbf{x}_2 + \mathbf{V}_c, \tag{7.8a}$$

$$\dot{\mathbf{x}}_2 = \mathbf{R}\Big(\left[f_u, hf_r, -hf_q\right]^{\mathrm{T}} - \mathcal{D}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}) - \mathcal{C}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}) + \boldsymbol{\omega}\times\boldsymbol{\upsilon}_r + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{\ell}) \tag{7.8b}$$

$$+ \boldsymbol{\ell} \times \left(\mathcal{D}_{\boldsymbol{\omega}}(\boldsymbol{\nu}) + \mathcal{C}_{\boldsymbol{\omega}}(\boldsymbol{\nu}) + \mathbf{M}'_{22}\left(Wz_{gb}\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right)\right)\Big).$$

Note that $\dot{\mathbf{x}}_2$ does not depend on the roll torque f_p. We can therefore use f_p to stabilize the roll dynamics by canceling the Coriolis effect:

$$f_p = \mathbf{e}_1^{\mathrm{T}} \mathcal{C}_{\boldsymbol{\omega}}(\boldsymbol{\nu}), \tag{7.9}$$

To linearize the output dynamics, we employ the following change of input

$$\begin{bmatrix} f_u \\ f_q \\ f_r \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -\frac{1}{h} \\ 0 & \frac{1}{h} & 0 \end{bmatrix} \Big(\mathbf{R}^{\mathrm{T}} \boldsymbol{\mu} + \mathcal{D}_{\boldsymbol{v}}(\boldsymbol{\nu}) + \mathcal{C}_{\boldsymbol{v}}(\boldsymbol{\nu}) - \boldsymbol{\omega} \times \boldsymbol{v}_r - \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \boldsymbol{\ell}) \\ - \boldsymbol{\ell} \times \Big(\mathcal{D}_{\boldsymbol{\omega}}(\boldsymbol{\nu}) + \mathcal{C}_{\boldsymbol{\omega}}(\boldsymbol{\nu}) + \mathbf{M}'_{22} \big(W z_{gb} \mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}} \mathbf{e}_3 \big) \Big) \Big), \tag{7.10}$$

where $\boldsymbol{\mu} \in \mathbb{R}^3$ is the new control input. This procedure transforms the system (7.5) into the following form

$$\dot{\mathbf{x}}_1 = \mathbf{x}_2 + \mathbf{V}_c, \tag{7.11a}$$
$$\dot{\mathbf{x}}_2 = \boldsymbol{\mu}, \tag{7.11b}$$
$$\dot{\mathbf{R}} = \mathbf{R}\mathbf{S}(\boldsymbol{\omega}), \tag{7.11c}$$
$$\dot{\boldsymbol{\omega}} = \bar{\boldsymbol{\ell}} \times \big(\mathbf{R}^{\mathrm{T}} \boldsymbol{\mu} + \mathcal{D}_{\boldsymbol{v}}(\boldsymbol{\nu}) + \mathcal{C}_{\boldsymbol{v}}(\boldsymbol{\nu}) - \boldsymbol{\omega} \times \mathbf{R}^{\mathrm{T}} \mathbf{x}_2 \big) \\ - \big(\bar{\boldsymbol{\ell}} \boldsymbol{\ell}^{\mathrm{T}} \big) \big(\mathcal{D}_{\boldsymbol{\omega}}(\boldsymbol{\nu}) + \mathbf{M}'_{22} \big(W z_{gb} \mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}} \mathbf{e}_3 \big) \big), \tag{7.11d}$$

where $\bar{\boldsymbol{\ell}} = [1/h, 0, 0]^{\mathrm{T}}$. Note that (7.11a) and (7.11b) form a double integrator with a constant disturbance caused by the ocean current.

7.2.1 Hand Position-Based Controller

In this section, we present some necessary assumtpions about the hand position-based controller. We assume that the goal of the control algorithm is to track a desired trajectory. Although this assumption seems restrictive, we will demonstrate that many controllers fall into this category.

Let $\boldsymbol{\xi}_{1,d}$ represent the desired trajectory, and let $\boldsymbol{\xi}_{2,d} = \dot{\boldsymbol{\xi}}_{1,d}$. We assume that there exist $\boldsymbol{\xi}_{2,d,\max}$ and $\dot{\boldsymbol{\xi}}_{2,d,\max}$ such that

$$\|\mathbf{V}_c\| < \|\boldsymbol{\xi}_{2,d}\| \leq \boldsymbol{\xi}_{2,d,\max}, \qquad \left\| \dot{\boldsymbol{\xi}}_{2,d} \right\| \leq \dot{\boldsymbol{\xi}}_{2,d,\max}. \tag{7.12}$$

Furthermore, we define the following error states

$$\tilde{\boldsymbol{\xi}}_1 = \mathbf{x}_1 - \boldsymbol{\xi}_{1,d}, \tag{7.13a}$$
$$\tilde{\boldsymbol{\xi}}_2 = \mathbf{x}_2 - \boldsymbol{\xi}_{2,d} + \mathbf{V}_c. \tag{7.13b}$$

The dynamics of these error states are given by

$$\dot{\tilde{\boldsymbol{\xi}}}_1 = \tilde{\boldsymbol{\xi}}_2, \tag{7.14a}$$
$$\dot{\tilde{\boldsymbol{\xi}}}_2 = \boldsymbol{\mu} - \dot{\boldsymbol{\xi}}_{2,d}. \tag{7.14b}$$

Assumption 7.2. *The hand position-based controller is designed such that the norm of the control input $\boldsymbol{\mu}$ is finite and the origin $\left[\tilde{\boldsymbol{\xi}}_1, \tilde{\boldsymbol{\xi}}_2\right] = \mathbf{0}^{\mathrm{T}}$ is a uniformly globally asymptotically stable (UGAS) equilibrium of* (7.14).

7.3 Closed-Loop Analysis

In this section, we analyze the closed-loop behavior of the orientation and the angular rates. Because these states cannot be controlled directly through the control input $\boldsymbol{\mu}$, they are commonly referred to as the *internal states*, while $\mathbf{x}_1$ and $\mathbf{x}_2$ are referred to as the *external states* [14]. For a generic hand position-based controller and a generic trajectory, the internal states do not converge to a specific value. Consequently, we intend to prove that the internal states are bounded. The orientation is restricted to a closed set $SO(3)$, and thus inherently bounded. Only the angular rates can grow unboundedly.

By the choice of the control law (7.9), the dynamics of the roll rate no longer depend on the other angular velocities. Indeed, from (7.11d), we get

$$\begin{aligned} \dot{p} &= -\mathbf{e}_1^{\mathrm{T}} \left(\mathcal{D}_{\boldsymbol{\omega}}(\boldsymbol{\nu}) + \mathbf{M}'_{22} \left(W z_{gb} \mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}} \mathbf{e}_3 \right) \right) \\ &= -\frac{d_{44}}{m_{44}} p - \frac{1}{m_{44}} \mathbf{e}_1^{\mathrm{T}} \left(W z_{gb} \mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}} \mathbf{e}_3 \right). \end{aligned} \tag{7.15}$$

Let us define

$$a_x = \frac{d_{44}}{m_{44}}, \qquad b_x = \frac{W z_{gb}}{m_{44}}, \tag{7.16}$$

and prove the following proposition:

Lemma 7.1. *The roll rate dynamics are bounded if $a_x > 0$. Specifically, the trajectory $p(t)$ satisfies*

$$|p(t)| \leq |p(0)| \, \mathrm{e}^{-a_x t} + \frac{b_x}{a_x} \left(1 - \mathrm{e}^{-a_x t} \right). \tag{7.17}$$

Proof. Consider the following two functions

$$V_p = \frac{1}{2} p^2, \qquad W_p = \sqrt{2 V_p}. \tag{7.18}$$

The following inequality holds for the derivative of W_p along the trajectories of p

$$\dot{W}_p \leq -a_x W_p + b_x. \tag{7.19}$$

By applying the comparison lemma, we get

$$W_p(t) = |p(t)| \leq |p(0)| \, \mathrm{e}^{-a_x t} + \frac{b_x}{a_x} \left(1 - \mathrm{e}^{-a_x t} \right), \tag{7.20}$$

which concludes the proof. □

Now, we investigate the boundedness of q and r. In the subsequent analysis, we will treat the roll rate and the external dynamics as a perturbation. From (7.11d), we get

$$\begin{bmatrix}\dot{q}\\ \dot{r}\end{bmatrix} = \begin{bmatrix}0 & 0 & -\frac{1}{h}\\ 0 & \frac{1}{h} & 0\end{bmatrix}\Big(\mathbf{R}^{\mathrm{T}}\boldsymbol{\mu} + \mathcal{D}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}) + \mathcal{C}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}) - \boldsymbol{\omega}\times\mathbf{R}^{\mathrm{T}}\Big(\tilde{\boldsymbol{\xi}}_2 + \boldsymbol{\xi}_{2,d} - \mathbf{V}_c\Big)\Big). \tag{7.21}$$

Note that the linear velocities of the vehicle can be expressed in terms of the external dynamics as

$$\boldsymbol{\upsilon}_r = \mathbf{R}^{\mathrm{T}}\left(\tilde{\boldsymbol{\xi}}_2 + \boldsymbol{\xi}_{2,d} - \mathbf{V}_c\right) - \boldsymbol{\omega}\times\boldsymbol{\ell}. \tag{7.22}$$

Let us define

$$\boldsymbol{\upsilon}_e = \mathbf{R}^{\mathrm{T}}\left(\tilde{\boldsymbol{\xi}}_2 + \boldsymbol{\xi}_{2,d} - \mathbf{V}_c\right) \triangleq [\upsilon_{e,1}, \upsilon_{e,2}, \upsilon_{e,3}]^{\mathrm{T}}. \tag{7.23}$$

Note that the norm of $\boldsymbol{\upsilon}_e$ can be bounded by the following expression

$$\|\boldsymbol{\upsilon}_e\| \leq \left\|\tilde{\boldsymbol{\xi}}_2\right\| + \|\boldsymbol{\xi}_{2,d} - \mathbf{V}_c\|, \tag{7.24}$$

and since the external dynamics are assumed UGAS, $\|\boldsymbol{\upsilon}_e\|$ converges to $\|\boldsymbol{\xi}_{2,d} - \mathbf{V}_c\|$. Consider then the following Lyapunov function candidate

$$V_{\boldsymbol{\omega}} = \frac{1}{2}\left(q^2 + r^2\right). \tag{7.25}$$

Let us define $\widehat{\boldsymbol{\omega}} = [q, r]^{\mathrm{T}}$. The following inequality holds for the derivative of $V_{\boldsymbol{\omega}}$ along the trajectories of (7.11)

$$\begin{aligned}\dot{V}_{\boldsymbol{\omega}} \leq & -a_y q^2 - a_z r^2 + \|\boldsymbol{\upsilon}_e\|\,\|\widehat{\boldsymbol{\omega}}\|\left(\frac{\|\boldsymbol{\omega}\|}{h} + a_e\right) + a_{xyz}pqr + a_{xy}\upsilon_{e,2}pq \\ & + a_{xz}\upsilon_{e,3}pr + a_{ye}\upsilon_{e,1}q^2 + a_{ze}\upsilon_{e,1}r^2 + a_{ley}\upsilon_{e,3}q \\ & + a_{lez}\upsilon_{e,2}r + a_{ey}\upsilon_{e,1}\upsilon_{e,3}q + a_{ez}\upsilon_{e,1}\upsilon_{e,2}r + \|\widehat{\boldsymbol{\omega}}\|\,\mu_{\max},\end{aligned} \tag{7.26}$$

where $\mu_{\max}$ is the largest norm of the control input. The remaining coefficients are shown in Appendix D.1.

Lemma 7.2. *Let us define*

$$\bar{p} = b_x/a_x, \qquad \bar{\boldsymbol{\upsilon}}_e = \max_{t\in\mathbb{R}_{\geq 0}} \|\boldsymbol{\xi}_{2,d}(t) - \mathbf{V}_c\|, \tag{7.27a}$$

$$\bar{\alpha}_y = a_y - \left(\frac{1}{h}\,\bar{\boldsymbol{\upsilon}}_e + \frac{1}{2}\,|a_{xyz}\bar{p}| + |a_{ye}\bar{\boldsymbol{\upsilon}}_e|\right), \tag{7.27b}$$

$$\bar{\alpha}_z = a_z - \left(\frac{1}{h}\,\bar{\boldsymbol{\upsilon}}_e + \frac{1}{2}\,|a_{xyz}\bar{p}| + |a_{ze}\bar{\boldsymbol{\upsilon}}_e|\right). \tag{7.27c}$$

The angular rate dynamics are ultimately bounded if $a_x, \bar{\alpha}_y, \bar{\alpha}_z > 0$.

Proof. Consider the candidate Lyapunov function $V_{\boldsymbol{\omega}}$ and the bound on its derivative in (7.26). Using the following identities

$$\|\boldsymbol{\omega}\| \, \|\widehat{\boldsymbol{\omega}}\| \leq (|p| + \|\widehat{\boldsymbol{\omega}}\|) \, \|\widehat{\boldsymbol{\omega}}\| \,, \tag{7.28a}$$

$$|pqr| \leq \frac{1}{2} \, |p| \left(q^2 + r^2\right), \tag{7.28b}$$

we arrive at the following upper bound on $\dot{V}_{\boldsymbol{\omega}}$

$$\dot{V}_{\boldsymbol{\omega}} \leq -\alpha_y q^2 - \alpha_z r^2 + G\left(\boldsymbol{v}_e, \boldsymbol{\omega}, \tilde{\boldsymbol{\xi}}_1, \tilde{\boldsymbol{\xi}}_2, \tilde{\boldsymbol{\xi}}_I, \dot{\boldsymbol{\xi}}_{2,d}\right), \tag{7.29}$$

where

$$\alpha_y = \left(a_y - \left(\frac{1}{h} \, \|\boldsymbol{v}_e\| + \frac{1}{2} \, |a_{xyz}| \, |p| + |a_{ye}| \, \|\boldsymbol{v}_e\|\right)\right), \tag{7.30a}$$

$$\alpha_z = \left(a_z - \left(\frac{1}{h} \, \|\boldsymbol{v}_e\| + \frac{1}{2} \, |a_{xyz}| \, |p| + |a_{ze}| \, \|\boldsymbol{v}_e\|\right)\right), \tag{7.30b}$$

and $G(\cdot)$ represents the terms that grow at most linearly with q and r.

From Lemma 7.1, we can conclude that if $a_x > 0$, then

$$\lim_{t \to \infty} |p(t)| \leq \bar{p}. \tag{7.31}$$

Moreover, this limit converges exponentially. Consequently, from (7.27) and (7.30), we get the following limits

$$\lim_{t \to \infty} \alpha_y \geq \bar{\alpha}_y, \qquad\qquad \lim_{t \to \infty} \alpha_z \geq \bar{\alpha}_z. \tag{7.32}$$

Therefore, if $\bar{\alpha}_y, \bar{\alpha}_z > 0$, then there exists a finite time T after which $\alpha_y, \alpha_z > 0$.

First, let us investigate the candidate Lyapunov function for $t < T$. Since α_y and α_z may be negative, we cannot prove boundedness. However, note that the derivative of the Lyapunov function in (7.29) has the following form

$$\dot{V}_{\boldsymbol{\omega}} \leq k \, \|\widehat{\boldsymbol{\omega}}\|^2 + G(\cdot), \tag{7.33}$$

where k is a positive constant and $G(\cdot)$ grows at most linearly with $\|\widehat{\boldsymbol{\omega}}\|$. We can therefore conclude that the dynamics of q and r are forward complete [110].

For $t \geq T$, $\dot{V}_{\boldsymbol{\omega}}$ has the following form

$$\dot{V}_{\boldsymbol{\omega}} \leq -\alpha_y q^2 - \alpha_z r^2 + G(\cdot) \tag{7.34}$$

For sufficiently large angular velocities, the quadratic term will dominate the linear term $G(\cdot)$, and q and r will remain bounded.

The angular rate dynamics are thus ultimately bounded. □

In Lemma 7.2, we derived the sufficient conditions for ultimate boundedness of a hand position-based controller. In the remainder of the chapter, we provide an interpretation of the condition $a_x, \bar{\alpha}_y, \bar{\alpha}_z > 0$, and compare it to our intuition.

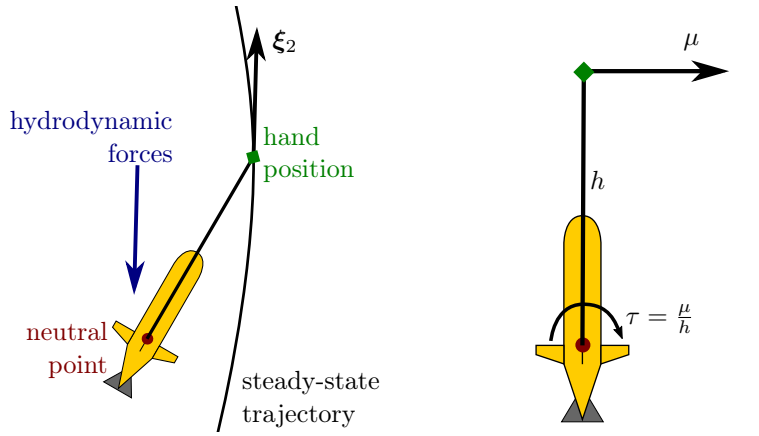
First, we analyze the term a_x. This term represents the effects of hydrodynamic damping on the roll rate of the vehicle. Based on the definition (7.16) and the fact that the inertia and damping matrices $\mathbf{M}$ and $\mathbf{D}$ are positive definite (*c.f.,* Section 2.1.3), the term a_x must be positive.

Next, we analyze the terms $\bar{\alpha}_y$ and $\bar{\alpha}_z$ defined in (7.27). First, we remark that there are some similarities between the hand position concept and a three-dimensional pendulum. In Figure 7.1a, we illustrate that the hand position point can be understood as a pivot, and the AUV can be understood as the weight of the pendulum. In a typical pendulum, the pivot is fixed, and the weight is affected by gravity. In the case of the hand position, the pivot moves and drags the weight through a resistive medium. The resulting hydrodynamic forces have a similar effect on the vehicle as gravity would have on the pendulum.

Using the pendulum analogy, let us analyze the terms $\bar{\alpha}_y$ and $\bar{\alpha}_z$. We can see that these terms depend on the physical parameters of the vehicle (*i.e.,* the terms a_x, a_y, a_{xyz}, a_{ye}, and a_{ze}), the hand length h, and the steady-state trajectory, represented by the term $\bar{\boldsymbol{v}}_e$.

First, let us investigate the terms a_y and a_z. These terms represent the effects of hydrodynamic damping on the sway and heave velocities. These terms have a stabilizing effect, meaning that the angular rate dynamics are ultimately bounded if a_y and a_z are sufficiently large. This fact is consistent with our pendulum analogy, since hydrodynamic forces have a dampening effect on the "swinging" motion of the AUV.

Next, the terms $\bar{\alpha}_y$ and $\bar{\alpha}_z$ depend on the norm of the steady-state velocity, $\bar{\boldsymbol{v}}_e$. We can see that $\bar{\boldsymbol{v}}_e$ has a destabilizing effect, meaning that the angular rates may grow unboundedly if $\bar{\boldsymbol{v}}_e$ is too large. Once again, this fact is consistent with our pendulum analogy, since dragging the pendulum at a high speed is likely to result in large oscillations. In (7.27b) and (7.27c), $\bar{\boldsymbol{v}}_e$ is multiplied by the terms a_{ye} and a_{ze}, respectively. These terms represent the cross-coupling between the surge



(a) The pendulum analogy **(b)** The effects of hand length

Figure 7.1: Illustrations of the hand position concept.

dynamics and the sway and heave dynamics of the vehicle. Note that the Lyapunov analysis in the proof is conservative and always considers the worst-case scenario. Consequently, we assume that the cross-coupling terms have a destabilizing effect.

Next, we investigate the effect of the hand length h. We can see that h has a stabilizing effect, the destabilizing term $\bar{\boldsymbol{v}}_e$ is divided by h. The intuition behind this effect is illustrated in Figure 7.1b. A greater hand length gives us more "leverage". In the figure, we show that if the lateral acceleration of the hand position point is μ, then the angular acceleration of the vehicle must be $\tau = \frac{\mu}{h}$. Consequently, a greater hand length results in smaller angular velocities.

Finally, the constants $\bar{\alpha}_y$ and $\bar{\alpha}_z$ depend on the term $|a_{xyz}\bar{p}|$. This term represents the cross-coupling between the roll dynamics and the sway and heave dynamics of the vehicle. Once again, due to the conservative nature of the underlying Lyapunov analysis, we assume that the cross-coupling has a destabilizing effect.

To summarize, when we design a hand position-based controller, we must ensure that the hand length h is sufficiently large and the steady-state velocity $\bar{\boldsymbol{v}}_e$ is sufficiently small to satisfy the requirements of Lemma 7.2. Otherwise, the angular rate dynamics may become unbounded.

Chapter 8

Trajectory Tracking and Path Following using the Hand Position Concept

This chapter presents hand position-based trajectory-tracking and path-following controllers for underactuated underwater vehicles. Using Lyapunov analysis, we show that the proposed controllers exponentially track the desired trajectory or path, while the angular velocities of the vehicle remain bounded. The theoretical results are verified both in numerical simulations and experiments. The contents of this chapter are based on [87].

The chapter is organized as follows. Sections 8.1 and 8.2 present and analyze the trajectory-tracking and the path-following controller, respectively. Sections 8.3 and 8.4 show the results of numerical simulations and experiments, respectively.

8.1 Trajectory Tracking

In this section, we propose a control law for tracking a predefined trajectory. Let $\boldsymbol{\xi}_{1,d}$ represent the desired trajectory, and let $\boldsymbol{\xi}_{2,d} = \dot{\boldsymbol{\xi}}_{1,d}$. We assume that there exist $\boldsymbol{\xi}_{2,d,\max}$ and $\dot{\boldsymbol{\xi}}_{2,d,\max}$ such that

$$\|\mathbf{V}_c\| < \|\boldsymbol{\xi}_{2,d}\| \leq \boldsymbol{\xi}_{2,d,\max}, \qquad \left\|\dot{\boldsymbol{\xi}}_{2,d}\right\| \leq \dot{\boldsymbol{\xi}}_{2,d,\max}. \tag{8.1}$$

The goal of trajectory tracking is to control the vehicle such that $\mathbf{x}_1$ converges to $\boldsymbol{\xi}_{1,d}$. To achieve the goal, we define the following error states

$$\tilde{\mathbf{x}}_1 = \mathbf{x}_1 - \boldsymbol{\xi}_{1,d}, \tag{8.2a}$$

$$\tilde{\mathbf{x}}_2 = \mathbf{x}_2 - \boldsymbol{\xi}_{2,d}, \tag{8.2b}$$

$$\tilde{\mathbf{x}}_I = \int_0^t \tilde{\mathbf{x}}_1(s)\,\mathrm{d}s, \tag{8.2c}$$

and choose the following PID controller

$$\boldsymbol{\mu} = -k_p\tilde{\mathbf{x}}_1 - k_d\tilde{\mathbf{x}}_2 - k_I\tilde{\mathbf{x}}_I + \dot{\boldsymbol{\xi}}_{2,d}, \tag{8.3}$$

where k_p, k_d, and k_I are positive gains chosen such that the matrix

$$\mathbf{H}_\xi = \begin{bmatrix} \mathbf{O}_{3\times3} & \mathbf{I}_3 & \mathbf{O}_{3\times3} \\ \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{I}_3 \\ -k_I\mathbf{I}_3 & -k_p\mathbf{I}_3 & -k_d\mathbf{I}_3 \end{bmatrix} \tag{8.4}$$

is Hurwitz.

8.1.1 Closed-Loop Analysis

In this section, we investigate the closed-loop dynamics of the system (7.11) with the control law (8.3). We begin by defining an additional change of coordinates:

$$\tilde{\boldsymbol{\xi}}_1 = \tilde{\mathbf{x}}_1, \tag{8.5a}$$

$$\tilde{\boldsymbol{\xi}}_2 = \tilde{\mathbf{x}}_2 + \mathbf{V}_c, \tag{8.5b}$$

$$\tilde{\boldsymbol{\xi}}_I = \tilde{\mathbf{x}}_I - \frac{k_d}{k_I}\mathbf{V}_c. \tag{8.5c}$$

This change is necessary to transform the equilibrium of the closed-loop system to the origin, as the error states $\tilde{\mathbf{x}}_2$ and $\tilde{\mathbf{x}}_I$ defined in (8.3) do not converge to zero.

For convenience, we will also define a concatenated state vector

$$\boldsymbol{\Xi}^{\mathrm{T}} = \left[\tilde{\boldsymbol{\xi}}_I^{\mathrm{T}}, \tilde{\boldsymbol{\xi}}_1^{\mathrm{T}}, \tilde{\boldsymbol{\xi}}_2^{\mathrm{T}}\right]. \tag{8.6}$$

Differentiating (8.6) with respect to time and substituting the external dynamics (7.11a)–(7.11b) and the control law (8.3), we get

$$\dot{\boldsymbol{\Xi}} = \mathbf{H}_\xi \boldsymbol{\Xi}. \tag{8.7}$$

Proposition 8.1. *The origin $\boldsymbol{\Xi} = \mathbf{0}$ is a globally exponentially stable (GES) equilibrium point of* (8.7)*. Consequently, $\mathbf{x}_1$, $\mathbf{x}_2$, and $\tilde{\mathbf{x}}_I$ exponentially converge to $\boldsymbol{\xi}_{1,d}$, $\boldsymbol{\xi}_{2,d} - \mathbf{V}_c$, and $k_I/k_d\,\mathbf{V}_c$, respectively.*

Moreover, let us define $a_x, \bar{\alpha}_y, \bar{\alpha}_z$ in accordance with (7.27)*. The internal dynamics are ultimately bounded if $a_x, \bar{\alpha}_y, \bar{\alpha}_z > 0$.*

Proof. Since the matrix $\mathbf{H}_\xi$, defined in (8.4) is Hurwitz by design, we can conclude that the external dynamics are GES, and $\tilde{\boldsymbol{\xi}}_1$, $\tilde{\boldsymbol{\xi}}_2$, and $\tilde{\boldsymbol{\xi}}_I$ exponentially converge to zero.

From (8.2), we can conclude that if $\tilde{\boldsymbol{\xi}}_1$ exponentially converges to zero, then the hand position $\mathbf{x}_1$ exponentially converges towards the desired trajectory $\boldsymbol{\xi}_{1,d}$. Similarly, if $\tilde{\boldsymbol{\xi}}_2$ converges to zero, then the relative hand velocity $\mathbf{x}_2$ converges to $\boldsymbol{\xi}_{2,d} - \mathbf{V}_c$. Moreover, if $\tilde{\boldsymbol{\xi}}_I$ converges to zero, then the integral state $\tilde{\mathbf{x}}_I$ converges to $k_I/k_d\,\mathbf{V}_c$. Consequently, the integral state provides an estimate of the ocean current.

Moreover, because the external dynamics are stable, the error states $\tilde{\mathbf{x}}_1$, $\tilde{\mathbf{x}}_2$, and $\tilde{\mathbf{x}}_I$ are bounded. Consequently, the control input $\boldsymbol{\mu}$ is bounded. Therefore, if $a_x, \bar{\alpha}_y, \bar{\alpha}_z > 0$, then all assumptions of Lemma 7.2 are satisfied, and the internal dynamics are ultimately bounded. □

8.1.2 Straight-Line Trajectory Tracking

In this section, we will focus on the special case when $\boldsymbol{\xi}_{2,d}$ is constant, and the vehicle is thus tracking a straight line. The purpose of this section is to demonstrate that in this special case, we can prove that both the external and internal dynamics are exponentially stable. First, let us present the necessary definitions and assumptions.

Definition 8.1. *Two vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$ are aligned if there exists $k \in \mathbb{R}$ such that $\mathbf{a} = k\mathbf{b}$. Equivalently, $\mathbf{a}$ and $\mathbf{b}$ are aligned if $\mathbf{a} \times \mathbf{b} = \mathbf{0}_3$.*

Assumption 8.1. *The distance between the centers of gravity and buoyancy, z_{gb}, is positive, and the vector $\boldsymbol{\xi}_{2,d,r} = (\boldsymbol{\xi}_{2,d} - \mathbf{V}_c)$ is not aligned with $\mathbf{e}_3$.*

Remark. *From (2.21), one can verify that if z_{gb} is positive, then the restoring forces stabilize the vehicle's roll angle around zero. Consequently, most commercial AUVs are designed so that $z_{gb} > 0$. The second part of Assumption 8.1 can be satisfied by choosing an appropriate reference trajectory.*

We begin by finding the equilibria of the closed-loop system. Since the external dynamics is a linear system, $\boldsymbol{\Xi} = \mathbf{0}_9$ is the only equilibrium. From (7.11c), we can conclude that $\dot{\mathbf{R}} = \mathbf{O}_{3\times 3}$ if and only if $\boldsymbol{\omega} = \mathbf{0}_3$. Substituting $\boldsymbol{\Xi} = \mathbf{0}_9$ and $\boldsymbol{\omega} = \mathbf{0}_3$ into (7.11d) yields

$$\dot{\boldsymbol{\omega}} = \bar{\boldsymbol{\ell}} \times (\mathcal{D}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}_r) + \mathcal{C}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}_r)) - \left(\bar{\boldsymbol{\ell}}\bar{\boldsymbol{\ell}}^{\mathrm{T}}\right)\left(\mathcal{D}_{\boldsymbol{\omega}}(\boldsymbol{\nu}_r) + \mathbf{M}'_{22}\left(W z_{gb}\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right)\right). \tag{8.8}$$

Note that the right-hand-side of (8.8) has the following form

$$\dot{\boldsymbol{\omega}} = \bar{\boldsymbol{\ell}} \times \mathbf{a} + \left(\bar{\boldsymbol{\ell}}\bar{\boldsymbol{\ell}}^{\mathrm{T}}\right)\mathbf{b}, \tag{8.9}$$

where $\mathbf{a} = [a_1, a_2, a_3]^{\mathrm{T}}$, $\mathbf{b} = [b_1, b_2, b_3]^{\mathrm{T}}$ are vectors in $\mathbb{R}^3$. From the definition of $\boldsymbol{\ell}$ and $\bar{\boldsymbol{\ell}}$, the following two equalities hold for any $\mathbf{a}$ and $\mathbf{b}$:

$$\bar{\boldsymbol{\ell}} \times \mathbf{a} = \frac{1}{h}\left[0, -a_3, a_2\right]^{\mathrm{T}}, \qquad \left(\bar{\boldsymbol{\ell}}\bar{\boldsymbol{\ell}}^{\mathrm{T}}\right)\mathbf{b} = \left[b_1, 0, 0\right]^{\mathrm{T}}. \tag{8.10}$$

Consequently, $\dot{\boldsymbol{\omega}}$ is zero if and only if both terms of the right-hand-side of (8.8) are zero.

The first term is zero only if $(\mathcal{D}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}_r) + \mathcal{C}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}_r))$ is aligned with $\mathbf{e}_1$. In other words, there exists $k \in \mathbb{R}$ such that

$$\mathcal{D}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}_r) + \mathcal{C}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}_r) = k\mathbf{e}_1. \tag{8.11}$$

Substituting (7.6) into (8.11), we get the following equation

$$\begin{bmatrix} \frac{d_{11}}{m_{11}}\, u_r \\ \frac{m_{66}d_{22} - d_{62}m_{26} + m_{26}(m_{11}-m_{22})u_r}{m_{22}m_{66} - m_{26}^2}\, v_r \\ \frac{m_{55}d_{33} - d_{53}m_{35} - m_{35}(m_{11}-m_{33})u_r}{m_{33}m_{55} - m_{35}^2}\, w_r \end{bmatrix} = \begin{bmatrix} k \\ 0 \\ 0 \end{bmatrix}, \tag{8.12}$$

which has at least one solution given by

$$[u_r, v_r, w_r]^{\mathrm{T}} = \frac{m_{11}}{d_{11}} k\mathbf{e}_1. \tag{8.13}$$

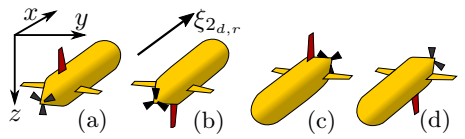


Figure 8.1: Illustration of the four equilibria. (a) Positive surge velocity, zero roll angle. (b) Positive surge velocity, π radians roll. (c) Negative surge velocity, zero roll angle. (d) Negative surge velocity, π radians roll.

Note that if

$$\frac{d_{62}m_{26} - m_{66}d_{22}}{m_{26}\left(m_{11} - m_{22}\right)} \neq \frac{m_{55}d_{33} - d_{53}m_{35}}{m_{35}\left(m_{11} - m_{33}\right)}, \tag{8.14}$$

then (8.13) is the only solution of (8.12). From (7.22), the steady-state linear velocities must also satisfy

$$\boldsymbol{\upsilon}_r = \mathbf{R}^{\mathrm{T}}\boldsymbol{\xi}_{2,d,r} \qquad \Longrightarrow \qquad \|\boldsymbol{\upsilon}_r\| = \|\boldsymbol{\xi}_{2,d,r}\|\,. \tag{8.15}$$

Combining (8.13) and (8.15) gives us two possible steady-state linear velocities

$$\boldsymbol{\upsilon}_r = \pm\,\|\boldsymbol{\xi}_{2,d,r}\|\;\mathbf{e}_1, \tag{8.16}$$

which leads to the following condition on the steady-state orientation

$$\mathbf{R}\mathbf{e}_1 = \pm\frac{\boldsymbol{\xi}_{2,d,r}}{\|\boldsymbol{\xi}_{2,d,r}\|}. \tag{8.17}$$

This condition does not uniquely define the steady-state orientation. Indeed, in terms of Euler angles, (8.17), defines the steady-state yaw and pitch angles. An additional condition on the steady-state orientation comes from the second term in (8.8). This term is zero if

$$\mathbf{e}_1^{\mathrm{T}}\left(W z_{gb}\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right) = 0. \tag{8.18}$$

If Assumption 8.1 holds, then (8.18) is equivalent to

$$\sin\phi = 0, \tag{8.19}$$

where $\phi \in [0, 2\pi)$ is the steady-state roll angle. Consequently, (8.17) and (8.18) result in four distinct steady-state orientations. Intuitively, the vehicle can have positive or negative surge velocity, and a roll angle of zero or π radians. These four equilibria are illustrated in Figure 8.1.

In the remainder of this section, we will analyze the equilibrium in which the vehicle has a positive surge velocity, and the roll angle is zero, *c.f.,* Figure 8.1a. Let $\mathbf{R}_0$ denote the steady-state orientation. We define the orientation error, $\boldsymbol{\delta}$, as

$$\boldsymbol{\delta} = \operatorname{logm}\left(\mathbf{R}_0^{\mathrm{T}}\,\mathbf{R}\right), \tag{8.20}$$

where $\operatorname{logm} : SO(3) \mapsto \mathfrak{B}_{\pi}^{3}$ is the matrix logarithm [111], and introduce the following change of coordinates

$$\tilde{\boldsymbol{\xi}}_I' = \mathbf{R}^{\mathrm{T}}\,\tilde{\boldsymbol{\xi}}_I, \qquad \tilde{\boldsymbol{\xi}}_1' = \mathbf{R}^{\mathrm{T}}\,\tilde{\boldsymbol{\xi}}_1, \qquad \tilde{\boldsymbol{\xi}}_2' = \mathbf{R}^{\mathrm{T}}\,\tilde{\boldsymbol{\xi}}_2. \tag{8.21}$$

The motivation behind the change of coordinates is to simplify the relation between the external dynamics and $\boldsymbol{\upsilon}_r$. From, (7.22), $\boldsymbol{\upsilon}_r$ is given by

$$\begin{aligned}\boldsymbol{\upsilon}_r &= \mathbf{R}^{\mathrm{T}}\left(\tilde{\boldsymbol{\xi}}_2 + \boldsymbol{\xi}_{2,d} - \mathbf{V}_c\right) - \boldsymbol{\omega}\times\boldsymbol{\ell} \\ &= \tilde{\boldsymbol{\xi}}_2' + \operatorname{expm}(\boldsymbol{\delta})^{\mathrm{T}}\left\|\boldsymbol{\xi}_{2_{d,r}}\right\|\mathbf{e}_1 - \boldsymbol{\omega}\times\boldsymbol{\ell},\end{aligned} \tag{8.22}$$

where $\operatorname{expm}:\mathfrak{B}_\pi^3 \mapsto SO(3)$ is the inverse of logm.

The complete closed-loop system is then given by

$$\dot{\tilde{\boldsymbol{\xi}}}_I' = \tilde{\boldsymbol{\xi}}_1' - \boldsymbol{\omega}\times\tilde{\boldsymbol{\xi}}_I', \tag{8.23a}$$

$$\dot{\tilde{\boldsymbol{\xi}}}_1' = \tilde{\boldsymbol{\xi}}_2' - \boldsymbol{\omega}\times\tilde{\boldsymbol{\xi}}_1', \tag{8.23b}$$

$$\dot{\tilde{\boldsymbol{\xi}}}_2' = -k_I\tilde{\boldsymbol{\xi}}_I' - k_p\tilde{\boldsymbol{\xi}}_1' - k_d\tilde{\boldsymbol{\xi}}_2' - \boldsymbol{\omega}\times\tilde{\boldsymbol{\xi}}_2', \tag{8.23c}$$

$$\dot{\boldsymbol{\delta}} = \boldsymbol{\omega}, \tag{8.23d}$$

$$\begin{aligned}\dot{\boldsymbol{\omega}} = \bar{\boldsymbol{\ell}}\times\Bigg(&- k_I\tilde{\boldsymbol{\xi}}_I' - k_p\tilde{\boldsymbol{\xi}}_1' - k_d\tilde{\boldsymbol{\xi}}_2' + \mathcal{D}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}_r) + \mathcal{C}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}_r) \\ &- \boldsymbol{\omega}\times\left(\tilde{\boldsymbol{\xi}}_2' + \operatorname{expm}(\boldsymbol{\delta})^{\mathrm{T}}\left\|\boldsymbol{\xi}_{2_{d,r}}\right\|\mathbf{e}_1\right)\Bigg) \\ &- \left(\bar{\boldsymbol{\ell}}\bar{\boldsymbol{\ell}}^{\mathrm{T}}\right)\left(\mathcal{D}_{\boldsymbol{\omega}}(\boldsymbol{\nu}_r) + \mathbf{M}_{22}'\left(Wz_{gb}\mathbf{e}_3\times\mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right)\right).\end{aligned} \tag{8.23e}$$

Next, we define a vector $\mathbf{z}^{\mathrm{T}} = \left[\tilde{\boldsymbol{\xi}}_I'^{\mathrm{T}}, \tilde{\boldsymbol{\xi}}_1'^{\mathrm{T}}, \tilde{\boldsymbol{\xi}}_2'^{\mathrm{T}}, \boldsymbol{\delta}^{\mathrm{T}}, \boldsymbol{\omega}^{\mathrm{T}}\right]$ and a function f such that $\dot{\mathbf{z}} = f(\mathbf{z})$. Let $\mathbf{J}$ denote the Jacobian of $f(\mathbf{z})$, evaluated at $\mathbf{z} = \mathbf{0}$. $\mathbf{J}$ is given by

$$\mathbf{J} = \begin{bmatrix} \mathbf{O}_{3\times3} & \mathbf{I}_3 & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} \\ \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{I}_3 & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} \\ -k_I\mathbf{I}_3 & -k_p\mathbf{I}_3 & -k_d\mathbf{I}_3 & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} \\ \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{I}_3 \\ \mathbf{J}_{\boldsymbol{\xi}_I} & \mathbf{J}_{\boldsymbol{\xi}_1} & \mathbf{J}_{\boldsymbol{\xi}_2} & \mathbf{J}_{\boldsymbol{\delta}} & \mathbf{J}_{\boldsymbol{\omega}} \end{bmatrix}, \tag{8.24}$$

where

$$\mathbf{J}_{\boldsymbol{\xi}_I} = -\mathbf{S}\left(\frac{k_I}{h}\mathbf{e}_1\right), \qquad \mathbf{J}_{\boldsymbol{\xi}_1} = -\mathbf{S}\left(\frac{k_p}{h}\mathbf{e}_1\right), \tag{8.25a}$$

$$\mathbf{J}_{\boldsymbol{\xi}_2} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \Xi_{23} \\ 0 & \Xi_{32} & 0 \end{bmatrix}, \qquad \mathbf{J}_{\boldsymbol{\omega}} = -\begin{bmatrix} \Omega_1 & 0 & 0 \\ 0 & \Omega_2 & 0 \\ 0 & 0 & \Omega_3 \end{bmatrix} \tag{8.25b}$$

$$\mathbf{J}_{\boldsymbol{\delta}} = \begin{bmatrix} -\Delta_1\cos\theta & 0 & \Delta_1\sin\theta \\ 0 & -\Delta_2 & 0 \\ 0 & 0 & -\Delta_3 \end{bmatrix}. \tag{8.25c}$$

where θ is the steady-state pitch angle. The components of $\mathbf{J}_{\boldsymbol{\xi}_2}$, $\mathbf{J}_{\boldsymbol{\omega}}$, and $\mathbf{J}_{\boldsymbol{\delta}}$ are shown in Appendix D.2.

Proposition 8.2. *The point $\mathbf{z} = \mathbf{0}_{15}$ is a (locally) exponentially stable equilibrium point of $\dot{\mathbf{z}} = f(\mathbf{z})$ if Assumption 8.1 holds, and all Δ_i and Ω_i for $i \in \{1,2,3\}$ are positive.*

Proof. Using the indirect Lyapunov method, the system is locally exponentially stable if $\mathbf{J}$ is Hurwitz. Let us define

$$\mathbf{J}_{21} = \begin{bmatrix} \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} \\ \mathbf{J}_{\boldsymbol{\xi}_I} & \mathbf{J}_{\boldsymbol{\xi}_1} & \mathbf{J}_{\boldsymbol{\xi}_2} \end{bmatrix}, \qquad \mathbf{J}_{22} = \begin{bmatrix} \mathbf{O}_{3\times3} & \mathbf{I}_3 \\ \mathbf{J}_{\delta} & \mathbf{J}_{\omega} \end{bmatrix}. \tag{8.26}$$

The matrix $\mathbf{J}$ can then be written in the following form

$$\mathbf{J} = \begin{bmatrix} \mathbf{H}_{\xi} & \mathbf{O}_{9\times6} \\ \mathbf{J}_{21} & \mathbf{J}_{22} \end{bmatrix}. \tag{8.27}$$

Due to its block triangular structure, the eigenvalues of $\mathbf{J}$ are equal to the union of eigenvalues of $\mathbf{H}_{\xi}$ and $\mathbf{J}_{22}$. Note that $\mathbf{H}_{\xi}$ is Hurwitz by design. Furthermore, the eigenvalues of $\mathbf{J}_{22}$, $\lambda_1, \ldots, \lambda_6$ are given by

$$\lambda_1, \lambda_2 = -\frac{\Omega_1}{2} \pm \frac{\sqrt{{\Omega_1}^2 - 4\,\Delta_1 \cos\theta_0}}{2}, \tag{8.28a}$$

$$\lambda_3, \lambda_4 = -\frac{\Omega_2}{2} \pm \frac{\sqrt{{\Omega_2}^2 - 4\,\Delta_2}}{2}, \tag{8.28b}$$

$$\lambda_5, \lambda_6 = -\frac{\Omega_3}{2} \pm \frac{\sqrt{{\Omega_3}^2 - 4\,\Delta_3}}{2}. \tag{8.28c}$$

If Assumption 8.1 holds, then the steady-state pitch angle satisfies $|\theta| < \pi/2$. Therefore, the real part of $\lambda_1, \ldots, \lambda_6$ is strictly negative if Assumption 8.1 holds and all Δ_i and Ω_i for $i \in \{1, 2, 3\}$ are positive. □

Remark. *For surface vessels, it has been shown that a similar controller can achieve almost global asymptotic stability [14]. For underwater vehicles, proving almost global stability is complicated. It can be shown that vehicles with rotational symmetry around the x-axis violate inequality* (8.14)*. Since most commercial AUVs have a cylindrical shape, this type of symmetry is common among underwater vehicles. If a vehicle violates inequality* (8.14)*, then there exists a subspace of unstable equilibria in addition to the four previously described equilibrium points, making almost global results impossible.*

8.2 Path Following

In this section, we present a path-following controller based on the hand position concept. Let s be the path parameter, $\mathbf{p}_p : \mathbb{R} \mapsto \mathbb{R}^3$ the parametrization of the desired path, and $U_d > 0$ the desired path following speed. We assume that $\mathbf{p}_p$ is $\mathcal{C}^2$ and parametrized by the arc length (see Section 2.2). Let us define the following three functions

$$\boldsymbol{\xi}_{1,d} = \mathbf{p}_p(s), \qquad \boldsymbol{\xi}_{2,d} = \dot{\boldsymbol{\xi}}_{1,d} = \dot{s}\frac{\partial \mathbf{p}_p(s)}{\partial s}, \qquad \boldsymbol{\xi}^*_{2,d} = U_d \frac{\partial \mathbf{p}_p(s)}{\partial s}, \tag{8.29}$$

The goal of the path following is to control the vehicle and continuously update the path parameter such that

$$\lim_{t\to\infty} \mathbf{x}_1(t) - \boldsymbol{\xi}_{1,d}(t) = \mathbf{0}_3, \qquad \lim_{t\to\infty} \dot{s}(t) = U_d. \tag{8.30}$$

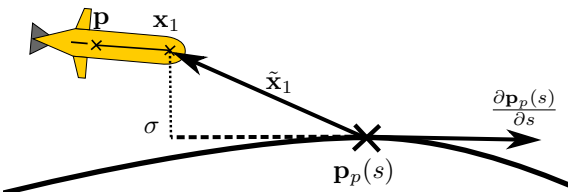


Figure 8.2: Illustration of the path function, the path following error, and its projection.

To solve the path-following problem, we define the following error states

$$\tilde{\mathbf{x}}_1 = \mathbf{x}_1 - \boldsymbol{\xi}_{1,d}, \tag{8.31a}$$

$$\tilde{\mathbf{x}}_2 = \mathbf{x}_2 - \boldsymbol{\xi}^*_{2,d}, \tag{8.31b}$$

$$\tilde{\mathbf{x}}_I = \int_0^t \tilde{\mathbf{x}}_1(\tau)\mathrm{d}\tau, \tag{8.31c}$$

and propose a PID controller analogous to the one in Section 8.1

$$\boldsymbol{\mu} = -k_p\tilde{\mathbf{x}}_1 - k_d\tilde{\mathbf{x}}_2 - k_I\tilde{\mathbf{x}}_I + \dot{\boldsymbol{\xi}}^*_{2,d}. \tag{8.32}$$

Inspired by [109], we propose the following update law for the path parameter

$$\dot{s} = U_d\left(1 + \varepsilon \tanh\left(k_\sigma \sigma\right)\right), \tag{8.33}$$

where ε and k_σ are positive gains, and $\sigma = \tilde{\mathbf{x}}_1^{\mathrm{T}} \frac{\partial \mathbf{p}_p(s)}{\partial s}$ is the projection of the path following error onto the path-tangential vector (see Figure 8.2). The motivation behind this update scheme is to allow the path to parameter "slow down" or "speed up" if the vehicle is lagging or leading the desired path.

8.2.1 Closed-Loop Analysis

Here we investigate the properties of the closed-loop system. Similarly to Section 8.1, we define the following change of coordinates

$$\tilde{\boldsymbol{\xi}}_1 = \tilde{\mathbf{x}}_1, \tag{8.34a}$$

$$\tilde{\boldsymbol{\xi}}_2 = \tilde{\mathbf{x}}_2 + \mathbf{V}_c, \tag{8.34b}$$

$$\tilde{\boldsymbol{\xi}}_I = \tilde{\mathbf{x}}_I - \frac{k_d}{k_I}\mathbf{V}_c. \tag{8.34c}$$

The external dynamics of the vehicle are then given by

$$\dot{\boldsymbol{\Xi}} = \mathbf{H}_\xi \boldsymbol{\Xi} + \left[\mathbf{0}_3^{\mathrm{T}}, \mathbf{d}^{\mathrm{T}}, k_d\mathbf{d}^{\mathrm{T}}\right]^{\mathrm{T}}, \tag{8.35}$$

where $\mathbf{H}_\xi$ is given by (8.4), and

$$\mathbf{d} = \varepsilon U_d \tanh\left(k_\sigma \sigma\right) \frac{\partial \mathbf{p}_p(s)}{\partial s}. \tag{8.36}$$

Since $\mathbf{H}_\xi$ is Hurwitz by design, for any positive definite matrix $\mathbf{Q}$ there exists a positive definite matrix $\mathbf{P}$ such that $\mathbf{H}_\xi^{\mathrm{T}}\mathbf{P} + \mathbf{P}\mathbf{H}_\xi = -\mathbf{Q}$. Let ϱ denote the ratio between the smallest eigenvalue of $\mathbf{Q}$ and the largest eigenvalue of $\mathbf{P}$, and let us choose $\mathbf{Q}$ such that ϱ is maximized.

Proposition 8.3. *The external dynamics* (8.35) *are GES if*

$$\varrho > 2\,(1 + k_d)\,\varepsilon k_\sigma U_d. \tag{8.37}$$

Consequently, $\mathbf{x}_1$, $\mathbf{x}_2$, *and* $\tilde{\mathbf{x}}_I$ *exponentially converge to* $\boldsymbol{\xi}_{1,d}$, $\boldsymbol{\xi}_{2,d}^* - \mathbf{V}_c$, *and* $k_d/k_I\,\mathbf{V}_c$, *respectively, and* $\dot{s}$ *converges to* U_d.

Moreover, let us define

$$\bar{\boldsymbol{\upsilon}}_e = \max_{s,\sigma\in\mathbb{R}} \left\| U_d\,(1 + \varepsilon\tanh(k_\sigma\sigma))\,\frac{\partial \mathbf{p}_p(s)}{\partial s} - \mathbf{V}_c \right\| \tag{8.38}$$

and $\bar{\alpha}_y$ *and* $\bar{\alpha}_z$ *in accordance with* (7.27). *The internal dynamics are ultimately bounded if* $a_x, \bar{\alpha}_y, \bar{\alpha}_z > 0$

Proof. Consider the following Lyapunov function candidate

$$V = \boldsymbol{\Xi}^{\mathrm{T}}\mathbf{P}\boldsymbol{\Xi}, \tag{8.39}$$

The derivative of V along the trajectories of the closed-loop system (8.35) is given by

$$\begin{aligned} \dot{V} &= \boldsymbol{\Xi}^{\mathrm{T}}\left(\mathbf{H}_\xi^{\mathrm{T}}\mathbf{P} + \mathbf{P}\mathbf{H}_\xi\right)\boldsymbol{\Xi} + 2\left[\mathbf{0}_3^{\mathrm{T}}, \mathbf{d}^{\mathrm{T}}, k_d\mathbf{d}^{\mathrm{T}}\right]\mathbf{P}\boldsymbol{\Xi} \\ &\leq -\lambda_{\mathbf{Q},\min}\left\|\boldsymbol{\Xi}\right\|^2 + 2\lambda_{\mathbf{P},\max}\,(1 + k_d)\left\|\mathbf{d}\right\|\left\|\boldsymbol{\Xi}\right\|, \end{aligned} \tag{8.40}$$

where $\lambda_{\mathbf{Q},\min}$ is the smallest eigenvalue of $\mathbf{Q}$, and $\lambda_{\mathbf{P},\max}$ is the largest eigenvalue of $\mathbf{P}$. From (8.36), we get the following upper bound on $\|\mathbf{d}\|$

$$\left\|\mathbf{d}\right\| \leq \varepsilon k_\sigma U_d \left\|\tilde{\boldsymbol{\xi}}_1\right\| \leq \varepsilon k_\sigma U_d \left\|\boldsymbol{\Xi}\right\|, \tag{8.41}$$

and arrive at the following expression

$$\dot{V} \leq -\left(\lambda_{\mathbf{Q},\min} - 2\lambda_{\mathbf{P},\max}\,(1 + k_d)\,\varepsilon k_\sigma U_d\right)\left\|\boldsymbol{\Xi}\right\|^2. \tag{8.42}$$

Therefore, if the following inequality holds

$$\frac{\lambda_{\mathbf{Q},\min}}{\lambda_{\mathbf{P},\max}} = \varrho > 2\,(1 + k_d)\,\varepsilon k_\sigma U_d, \tag{8.43}$$

the origin of the closed-loop system is GES, and thus $\tilde{\boldsymbol{\xi}}_1$, $\tilde{\boldsymbol{\xi}}_2$, and $\tilde{\boldsymbol{\xi}}_I$ exponentially converge to zero. Using the same arguments as in the proof of Proposition 8.1, we can conclude that $\mathbf{x}_1$, $\mathbf{x}_2$, and $\tilde{\mathbf{x}}_I$ exponentially converge to $\boldsymbol{\xi}_{1,d}$, $\boldsymbol{\xi}_{2,d}^* - \mathbf{V}_c$, and $k_d/k_I\,\mathbf{V}_c$, respectively. In addition, substituting $\tilde{\boldsymbol{\xi}}_1 = \mathbf{0}_3$ into the path parameter update law (8.33) gives $\dot{s} = U_d$.

Moreover, because the external dynamics are stable, the error states $\tilde{\mathbf{x}}_1$, $\tilde{\mathbf{x}}_2$, and $\tilde{\mathbf{x}}_I$ are bounded. Consequently, the control input $\boldsymbol{\mu}$ is bounded. Note that $\bar{\boldsymbol{\upsilon}}_e$ defined in (8.38) represents the upper bound on $\|\boldsymbol{\xi}_{2,d} - \mathbf{V}_c\|$. Therefore, if $a_x, \bar{\alpha}_y, \bar{\alpha}_z > 0$, then all assumptions of Lemma 7.2 are satisfied, and the internal dynamics are ultimately bounded. □

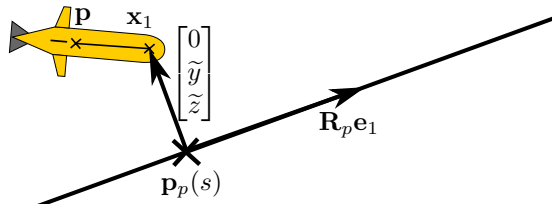


Figure 8.3: Illustration of straight-line path following.

8.2.2 Straight-Line Path Following

In this section, we propose a path following controller for straight-line paths. Similarly to Section 8.1.2, we can prove the exponential stability of the whole closed-loop system. Moreover, straight-line paths are natively supported by most guidance, navigation, and control systems, *e.g.*, the Unified Navigation Environment (DUNE) [89]. Such paths can be parametrized by the following function

$$\mathbf{p}_p(s) = \mathbf{p}_0 + \mathbf{R}_p \mathbf{e}_1 s, \tag{8.44}$$

where $\mathbf{p}_0 \in \mathbb{R}^3$ is the origin of the path, and $\mathbf{R}_p \in SO(3)$ defines the orientation of the path.

Instead of the path parameter update law (8.33), we propose to determine the path parameter by finding the closest point on the desired path to the vehicle's hand position. This approach is commonly done in the literature when following straight lines or circles [121]. From (8.44), the path parameter of the closest point to x_1 is given by

$$s = (\mathbf{x}_1 - \mathbf{p}_0)^{\mathrm{T}} \mathbf{R}_p \mathbf{e}_1. \tag{8.45}$$

In addition, let us define cross-track errors, $\widetilde{y}$ and $\widetilde{z}$, as

$$[0, \widetilde{y}, \widetilde{z}]^{\mathrm{T}} = \mathbf{R}_p^{\mathrm{T}} \left(\mathbf{x}_1 - \mathbf{p}_p(s)\right). \tag{8.46}$$

The cross-track errors are illustrated in Figure 8.3. Substituting (8.45) and (8.44) into (8.46), we get

$$\begin{bmatrix} \widetilde{y} \\ \widetilde{z} \end{bmatrix} = \hat{\mathbf{I}}^{\mathrm{T}} \mathbf{R}_p^{\mathrm{T}} \left(\mathbf{x}_1 - \mathbf{p}_0\right), \qquad \hat{\mathbf{I}}^{\mathrm{T}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{8.47}$$

For a straight-line path, the control goal (8.30) is equivalent to controlling the vehicle such that $\widetilde{y}$ and $\widetilde{z}$ converge to zero. To achieve the goal, we define the following error states

$$\tilde{\mathbf{x}}_1 = [\widetilde{y}, \widetilde{z}]^{\mathrm{T}}, \qquad \tilde{\mathbf{x}}_2 = \mathbf{R}_p^{\mathrm{T}} \mathbf{x}_2 - U_d \mathbf{e}_1, \qquad \tilde{\mathbf{x}}_I = \int_0^t \tilde{\mathbf{x}}_1(\tau) \mathrm{d}\tau, \tag{8.48}$$

and a PID control law

$$\boldsymbol{\mu} = -k_p \mathbf{R}_p \hat{\mathbf{I}} \tilde{\mathbf{x}}_1 - k_d \mathbf{R}_p \tilde{\mathbf{x}}_2 - k_I \mathbf{R}_p \hat{\mathbf{I}} \tilde{\mathbf{x}}_I, \tag{8.49}$$

where k_p, k_d, and k_I are positive gains chosen such that the matrix

$$\bar{\mathbf{H}}_\xi = \begin{bmatrix} \mathbf{O}_{2\times2} & \mathbf{I}_2 & \mathbf{O}_{2\times3} \\ \mathbf{O}_{2\times2} & \mathbf{O}_{2\times2} & \hat{\mathbf{I}}^{\mathrm{T}} \\ -k_I\hat{\mathbf{I}} & -k_p\hat{\mathbf{I}} & -k_d\mathbf{I}_3 \end{bmatrix}, \tag{8.50}$$

is Hurwitz. Similarly to the previous sections, we can perform the following change of coordinates

$$\widetilde{\boldsymbol{\xi}}_1 = \tilde{\mathbf{x}}_1, \tag{8.51a}$$

$$\widetilde{\boldsymbol{\xi}}_2 = \tilde{\mathbf{x}}_2 + \tilde{\mathbf{I}}\mathbf{R}_p^{\mathrm{T}}\mathbf{V}_c, \tag{8.51b}$$

$$\widetilde{\boldsymbol{\xi}}_I = \tilde{\mathbf{x}}_I - \frac{k_d}{k_I}\hat{\mathbf{I}}^{\mathrm{T}}\mathbf{R}_p^{\mathrm{T}}\mathbf{V}_c, \tag{8.51c}$$

where $\tilde{\mathbf{I}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$. For convenience, let us define $\boldsymbol{\Xi}'^{\mathrm{T}} = \begin{bmatrix} \widetilde{\boldsymbol{\xi}}_I'^{\mathrm{T}}, \widetilde{\boldsymbol{\xi}}_1'^{\mathrm{T}}, \widetilde{\boldsymbol{\xi}}_2'^{\mathrm{T}} \end{bmatrix}$.

Proposition 8.4. *The external dynamics are GES. Specifically, $\widetilde{y}$ and $\widetilde{z}$ converge to zero, x_2 converges to $U_d\mathbf{R}_p\mathbf{e}_1 - \mathbf{R}_p\tilde{\mathbf{I}}\mathbf{R}_p^{\mathrm{T}}\mathbf{V}_c$, and $\tilde{\mathbf{x}}_I$ converges to $\frac{k_d}{k_I}\hat{\mathbf{I}}^{\mathrm{T}}\mathbf{R}_p^{\mathrm{T}}\mathbf{V}_c$.*

Moreover, let us define $\boldsymbol{\xi}_{2,d} = U_d\mathbf{R}_p\mathbf{e}_1$, $\bar{\upsilon}_e = \|\boldsymbol{\xi}_{2,d} - \mathbf{V}_c\|$, and a_x, $\bar{\alpha}_y$, and $\bar{\alpha}_z$ in accordance with (7.27)*. The internal dynamics are ultimately bounded if $a_x, \bar{\alpha}_y, \bar{\alpha}_z > 0$.*

Proof. Differentiating (8.51) with respect to time yields

$$\dot{\boldsymbol{\Xi}} = \bar{\mathbf{H}}_\xi\boldsymbol{\Xi}, \tag{8.52}$$

with $\bar{\mathbf{H}}_\xi$ defined in (8.50). Since $\bar{\mathbf{H}}_\xi$ is Hurwitz by design, $\boldsymbol{\Xi}$ exponentially converges to zero. From (8.51), if $\boldsymbol{\Xi}$ converges to zero, then the cross-track errors $\widetilde{y}$ and $\widetilde{z}$ converge to zero, $\mathbf{x}_2$ converges to $U_d\mathbf{R}_p\mathbf{e}_1 - \mathbf{R}_p\tilde{\mathbf{I}}\mathbf{R}_p^{\mathrm{T}}\mathbf{V}_c$, and $\tilde{\mathbf{x}}_I$ converges to $\frac{k_d}{k_I}\hat{\mathbf{I}}^{\mathrm{T}}\mathbf{R}_p^{\mathrm{T}}\mathbf{V}_c$.

Moreover, because the external dynamics are stable, the error states $\tilde{\mathbf{x}}_1$, $\tilde{\mathbf{x}}_2$, and $\tilde{\mathbf{x}}_I$ are bounded. Consequently, the control input $\boldsymbol{\mu}$ is bounded. Therefore, if $a_x, \bar{\alpha}_y, \bar{\alpha}_z > 0$, then all assumptions of Lemma 7.2 are satisfied, and the internal dynamics are ultimately bounded. □

Finally, let us investigate the exponential stability of the whole closed-loop system. We begin by finding the equilibria. Let us define the desired relative velocity $\boldsymbol{\xi}_{2,d,r} = U_d\mathbf{R}_p\mathbf{e}_1 - \mathbf{R}_p\tilde{\mathbf{I}}\mathbf{R}_p^{\mathrm{T}}\mathbf{V}_c$. Using the same procedure as in Section 8.1.2, we can conclude that if Assumption 8.1 holds, then the steady-state orientation must satisfy

$$\mathbf{R}\mathbf{e}_1 = \pm\frac{\boldsymbol{\xi}_{2,d,r}}{\|\boldsymbol{\xi}_{2,d,r}\|}, \qquad \sin\phi = 0, \tag{8.53}$$

where ϕ is the steady-state roll angle.

Using the orientation error $\boldsymbol{\delta}$, as defined in (8.20), the complete closed-loop system is given by

$$\dot{\tilde{\boldsymbol{\xi}}}_I = \tilde{\boldsymbol{\xi}}_1, \tag{8.54a}$$

$$\dot{\tilde{\boldsymbol{\xi}}}_1 = \tilde{\boldsymbol{\xi}}_2, \tag{8.54b}$$

$$\dot{\tilde{\boldsymbol{\xi}}}_2 = -k_I \hat{\mathbf{I}} \tilde{\boldsymbol{\xi}}_I - k_p \hat{\mathbf{I}} \tilde{\boldsymbol{\xi}}_1 - k_d \tilde{\boldsymbol{\xi}}_2, \tag{8.54c}$$

$$\dot{\boldsymbol{\delta}} = \boldsymbol{\omega}, \tag{8.54d}$$

$$\begin{aligned} \dot{\boldsymbol{\omega}} = \bar{\boldsymbol{\ell}} \times \Bigg(& - k_I \mathbf{R}^{\mathrm{T}} \mathbf{R}_p \hat{\mathbf{I}} \tilde{\boldsymbol{\xi}}_I - k_p \mathbf{R}^{\mathrm{T}} \mathbf{R}_p \hat{\mathbf{I}} \tilde{\boldsymbol{\xi}}_1 \\ & - k_d \mathbf{R}^{\mathrm{T}} \mathbf{R}_p \tilde{\boldsymbol{\xi}}_2 + \mathcal{D}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}) + \mathcal{C}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}) \\ & - \boldsymbol{\omega} \times \left(\mathbf{R}^{\mathrm{T}} \mathbf{R}_p \tilde{\boldsymbol{\xi}}_2 + \operatorname{expm}(\boldsymbol{\delta})^{\mathrm{T}} \left\| \boldsymbol{\xi}_{2_{d,r}} \right\| \mathbf{e}_1 \right) \Bigg) \\ & - \left(\bar{\boldsymbol{\ell}} \bar{\boldsymbol{\ell}}^{\mathrm{T}} \right) \left(\mathcal{D}_{\boldsymbol{\omega}}(\boldsymbol{\nu}) + \mathbf{M}_{22} \left(W z_{gb} \mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}} \mathbf{e}_3 \right) \right), \end{aligned} \tag{8.54e}$$

Next, we define a vector $\bar{\mathbf{z}}^{\mathrm{T}} = \left[\boldsymbol{\Xi}^{\mathrm{T}}, \boldsymbol{\delta}^{\mathrm{T}}, \boldsymbol{\omega}^{\mathrm{T}} \right]$ and a function $\bar{f}$ such that $\dot{\bar{\mathbf{z}}} = \bar{f}(\bar{\mathbf{z}})$. Let $\bar{\mathbf{J}}$ denote the Jacobian of $\bar{f}(\bar{\mathbf{z}})$, evaluated at $\bar{\mathbf{z}} = \mathbf{0}_{13}$. $\bar{\mathbf{J}}$ is given by

$$\bar{\mathbf{J}} = \begin{bmatrix} \mathbf{O}_{2\times2} & \mathbf{I}_2 & \mathbf{O}_{2\times3} & \mathbf{O}_{2\times3} & \mathbf{O}_{2\times3} \\ \mathbf{O}_{2\times2} & \mathbf{O}_{2\times2} & \hat{\mathbf{I}}^{\mathrm{T}} & \mathbf{O}_{2\times3} & \mathbf{O}_{2\times3} \\ -k_I \hat{\mathbf{I}} & -k_p \hat{\mathbf{I}} & -k_d \mathbf{I}_3 & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} \\ \mathbf{O}_{3\times2} & \mathbf{O}_{3\times2} & \mathbf{O}_{3\times3} & \mathbf{O}_{3\times3} & \mathbf{I}_3 \\ \mathbf{R}^{\mathrm{T}} \mathbf{R}_p \hat{\mathbf{I}} \mathbf{J}_{\boldsymbol{\xi}_I} & \mathbf{R}^{\mathrm{T}} \mathbf{R}_p \hat{\mathbf{I}} \mathbf{J}_{\boldsymbol{\xi}_1} & \mathbf{R}^{\mathrm{T}} \mathbf{R}_p \mathbf{J}_{\boldsymbol{\xi}_2} & \mathbf{J}_{\delta} & \mathbf{J}_{\boldsymbol{\omega}} \end{bmatrix}. \tag{8.55}$$

The blocks of $\bar{\mathbf{J}}$ are shown in (8.25). Using the same reasoning as in the proof of Proposition 8.2, we can conclude that the closed-loop system is exponentially stable if Assumption 8.1 holds and all Δ_i and Ω_i for $i \in \{1, 2, 3\}$ are positive. □

8.3 Simulations

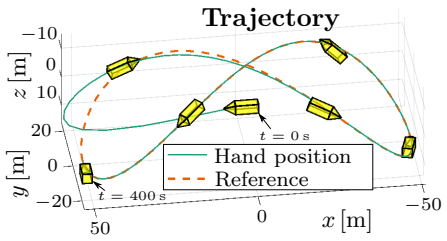
In this section, we present the results of numerical simulations. The simulations were carried out in MATLAB using a model of the light autonomous underwater vehicle (LAUV).

We tested the trajectory tracking algorithm proposed in Section 8.1, the curved path following algorithm proposed in Section 8.2, and the straight-line path following algorithm in Section 8.2.2. The following parameters are common for the first two tests: The initial state of the vehicle is $\mathbf{p}(0) = \mathbf{0}_3$, $\mathbf{R}(0) = \mathbf{I}_3$, $\boldsymbol{\upsilon}_r(0) = \mathbf{e}_1$, $\omega(0) = \mathbf{0}_3$. The hand length is $h = 5\,\mathrm{m}$, the velocity of the ocean current is $\mathbf{V}_c^{\mathrm{T}} = [0.15, -0.1, 0.05]$, and the PID gains are $k_p = 0.03, k_d = 0.4, k_I = 8.5 \cdot 10^{-4}$.

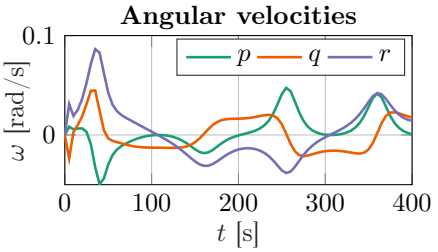
8.3.1 Trajectory Tracking

In this test, the vehicle should track a figure eight trajectory

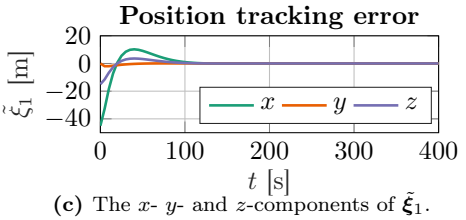
$$\boldsymbol{\xi}_{1,d}(t) = \left[50 \cos\left(\frac{\pi}{200} t \right), 25 \sin\left(\frac{2\pi}{200} t \right), 15 \cos\left(\frac{2\pi}{200} t \right) \right]^{\mathrm{T}}. \tag{8.56}$$



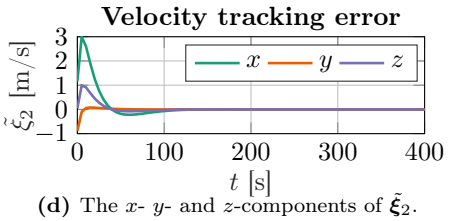
(a) Trajectory. The green line represents the hand position $\mathbf{x}_1$, the red line represents the desired trajectory $\boldsymbol{\xi}_{1,d}$, and the polygons represent the position and orientation of the AUV.



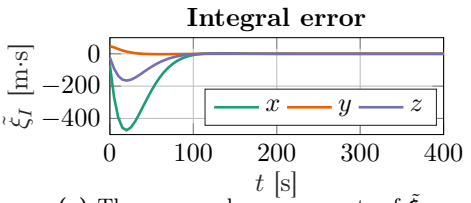
(b) The angular velocities.



(c) The x- y- and z-components of $\tilde{\boldsymbol{\xi}}_1$.



(d) The x- y- and z-components of $\tilde{\boldsymbol{\xi}}_2$.



(e) The x- y- and z-components of $\tilde{\boldsymbol{\xi}}_I$.

Figure 8.4: Simulation results of the trajectory-tracking algorithm proposed in Section 8.1.

We use the trajectory-tracking controller proposed in Section 8.1. The PID gains are chosen such that $\mathbf{H}_\xi$ is Hurwitz, guaranteeing the stability of the external dynamics. The value of $\bar{\alpha}_y = \bar{\alpha}_z$ for the chosen parameters is 0.05, guaranteeing the boundedness of the internal dynamics by Proposition 8.1.

The results are shown in Figure 8.4. Figure 8.4a shows the 3D trajectory of the vehicle, and Figures 8.4c, 8.4d, and 8.4e show the external dynamics. The vehicle converges to the desired trajectory after approximately 120 seconds. Figure 8.4b shows the angular velocities of the vehicle. Initially, the angular velocities grow. However, after the external dynamics have converged, the angular velocities remain bounded.

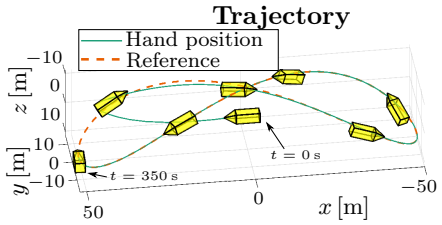
8.3.2 Path Following

In this test, we have chosen a path with the same shape as in the trajectory-tracking test. The path parametrization is given by

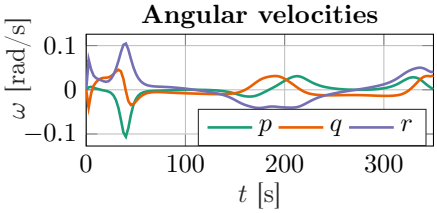
$$\mathbf{p}_p(s) = \left[50\cos\left(\gamma(s)\right), 25\sin\left(2\gamma(s)\right), 15\cos\left(2\gamma(s)\right)\right]^{\mathrm{T}}, \tag{8.57}$$

where $\gamma : \mathbb{R} \mapsto \mathbb{R}$ is a function chosen such that $\mathbf{p}_p(s)$ is a parametrization by arc length (see Section 2.2).

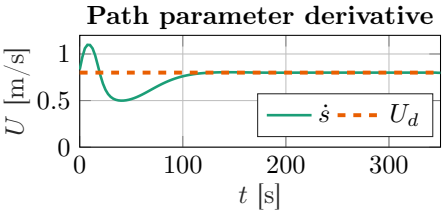
We use the path-following controller proposed in Section 8.2. The gains of the path parameter update law are chosen as $\varepsilon = 0.5$, and $k_\sigma = 0.1$. The value of



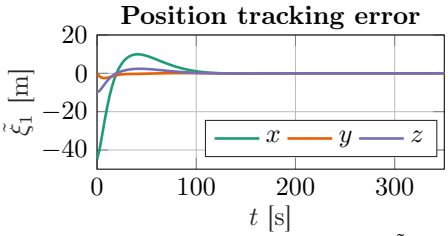
(a) Trajectory. The green line represents the hand position $\mathbf{x}_1$, the red line represents the desired trajectory $\boldsymbol{\xi}_{1,d}$, and the polygons represent the position and orientation of the AUV.



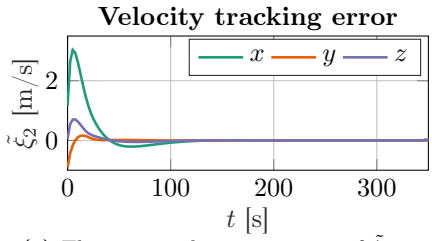
(b) The angular velocities.



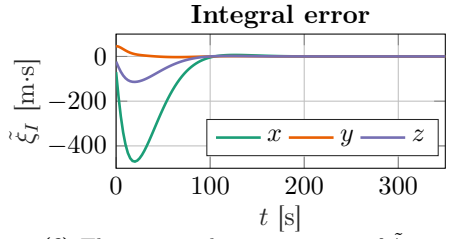
(c) The time-derivative of the path parameter.



(d) The x- y- and z-components of $\tilde{\boldsymbol{\xi}}_1$.



(e) The x- y- and z-components of $\tilde{\boldsymbol{\xi}}_2$.



(f) The x- y- and z-components of $\tilde{\boldsymbol{\xi}}_I$.

Figure 8.5: Simulation results of the path-following algorithm proposed in Section 8.2.

$\bar{\alpha}_y = \bar{\alpha}_z$ for the chosen parameters is 0.02, guaranteeing the boundedness of the internal dynamics by Proposition 8.3.

The results are shown in Figure 8.5. Figure 8.5a shows the 3D trajectory of the vehicle, and Figures 8.5d, 8.5e, and 8.5f show the external dynamics. Compared to the trajectory tracking simulation, the vehicle converges to the desired path faster, after approximately 90 seconds. Figure 8.5b shows the angular velocities of the vehicle. The behavior is very similar to the trajectory tracking simulation. Figure 8.5c shows the derivative of the path parameter. We can see that $\dot{s}$ decreases or increases depending on whether the vehicle is "behind" or "in front of" the desired path. After the transient period, $\dot{s}$ converges to U_d.

Since the vehicle model and parameters of the controller are identical for both simulations, the results of these simulations are very similar. The main difference between the proposed controllers is that in path following, one has an additional "degree of freedom" when choosing the path parameter, while in trajectory tracking, the trajectory is parametrized in time and thus fixed. In [122], it is argued that the control signals of path-following controllers are smoother and have a lower peak

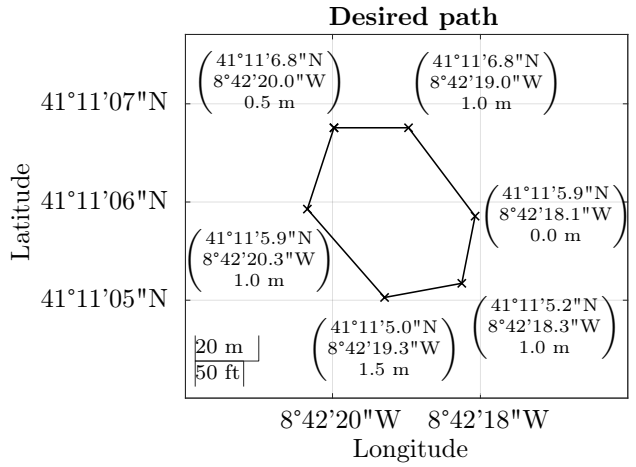


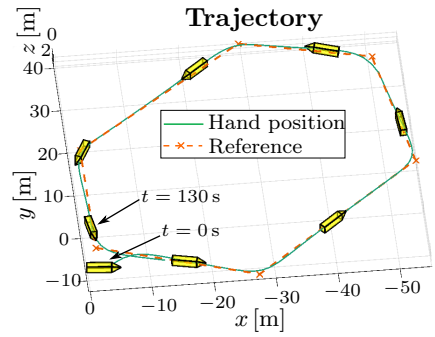
Figure 8.6: The desired path used in the simulation in Section 8.3.3 and experiment, with waypoints in (latitude, longitude, depth) format.

value than in the case of trajectory-tracking controllers. This fact is confirmed by the simulations, as the peak value of the control input $\boldsymbol{\mu}$ is $0.1\,\mathrm{m\,s^{-2}}$ lower, and the surge velocity u_r is $0.2\,\mathrm{m\,s^{-1}}$ lower in the path-following simulation.

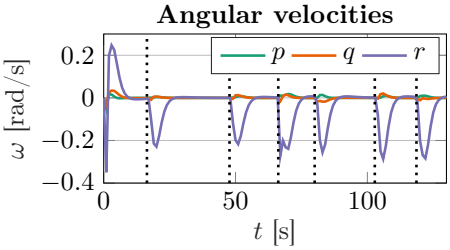
8.3.3 Straight-Line Path Following

In this test, the vehicle should follow a path consisting of a series of waypoints connected by line segments. The vehicle switches to the next line segment when the distance to the current waypoint is less than five meters. The desired path is shown in Figure 8.6. The parameters of the simulation are chosen identically to the experiment described in the next section. The initial position of the vehicle is $\mathbf{p}(0) = [-1.8, -9.3, 0]^{\mathrm{T}}$, the initial yaw angle is 185 degrees, and the remaining angles and velocities are zero. The desired path-following speed is $U_d = 1.3\,\mathrm{m\,s^{-1}}$, and the gains of the PID controller are $k_p = 0.2$, $k_d = 0.9$, $k_I = 0.01$. The gains are chosen so that $\bar{\mathbf{H}}_\xi$ is Hurwitz, guaranteeing the exponential stability of the external dynamics by Proposition 8.4.

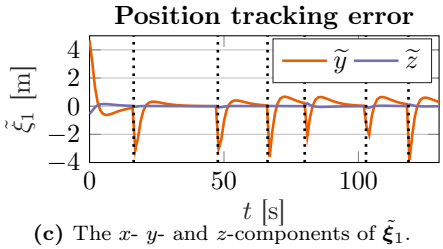
The results are shown in Figure 8.7. Figure 8.7a shows the trajectory of the vehicle. The vehicle starts converging to the desired line segment. When it reaches the circle of acceptance, *i.e.,* five meters within the current waypoint, it switches to the next segment. Figure 8.7b shows the angular velocities. The dotted vertical lines indicate when the waypoints change. We can see that after each change, there is a transient period where the angular velocities increase before converging back to zero. Figures 8.7c–8.7e show the position, velocity, and integral errors. When the waypoints change, the steady-state value of the external states changes as well, causing an abrupt increase in the error states. The error states then exponentially converge to zero.



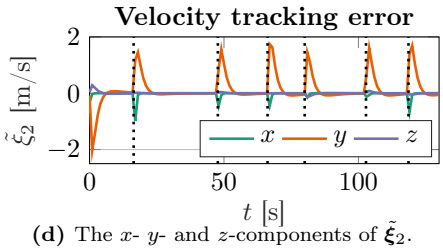
(a) Trajectory. The green line represents the hand position $\mathbf{x}_1$, the red line represents the desired trajectory $\boldsymbol{\xi}_{1,d}$, and the polygons represent the position and orientation of the AUV.



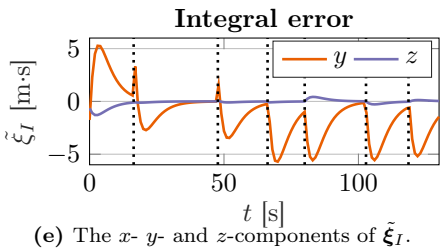
(b) The angular velocities.



(c) The x- y- and z-components of $\tilde{\boldsymbol{\xi}}_1$.



(d) The x- y- and z-components of $\tilde{\boldsymbol{\xi}}_2$.



(e) The x- y- and z-components of $\tilde{\boldsymbol{\xi}}_I$.

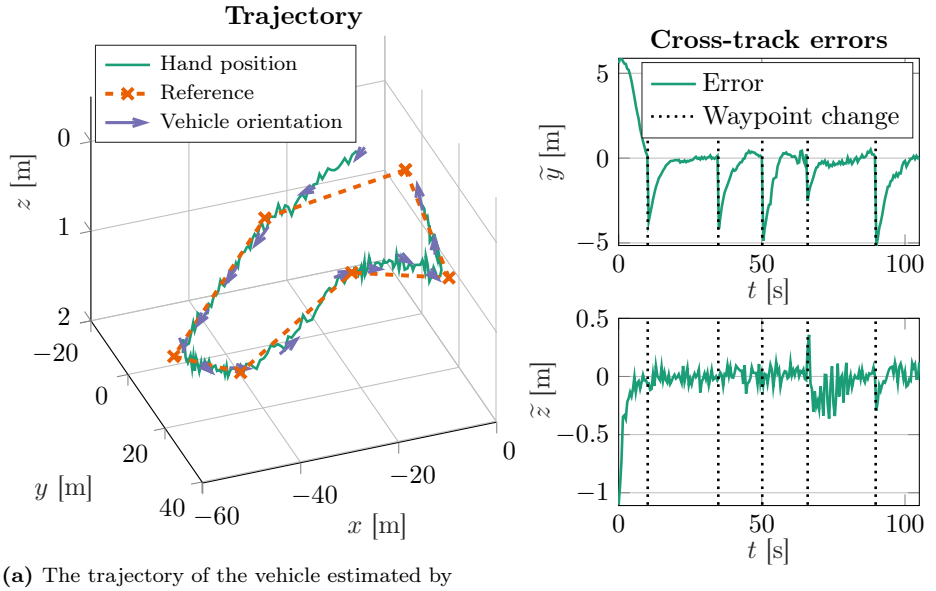
Figure 8.7: Simulation results of the path-following algorithm proposed in Section 8.2.

8.4 Experiments

In this section, we present the results of an experiment performed on the LAUV. In the experiment, we verify the effectiveness of the straight-line path following controller proposed in Section 8.2.2. The reason for choosing this specific controller for experimental validation is that straight-line paths are natively supported by DUNE [89], the onboard software running on the LAUV.

The experiment was performed at the harbor of Porto, Portugal. Due to shallow water, the depth of the vehicle had to be restricted to two meters. To fully utilize the available space, the desired path was given by a series of waypoints with varying depths arranged in a hexagon (see Figure 8.6). The vehicle follows straight-line segments given by the waypoints, and switches to the next waypoint when the distance to the current waypoint is less than five meters. The parameters of the controller are identical to the simulation in Section 8.3.3.

The results of the experiment are shown in Figure 8.8. Figure 8.8a shows the trajectory of the AUV. The green line represents the hand position $\mathbf{x}_1$, the red line represents the desired path, and the arrows represent the orientation of the vehicle, with the base of the arrow located at $\mathbf{p}$, and the tip of the arrow pointing



(a) The trajectory of the vehicle estimated by its onboard navigation system. The z-axis is upscaled 20 times.

(b) Cross-track errors calculated from the trajectory estimates.

Figure 8.8: Experimental results.

towards $\mathbf{x}_1$. Figure 8.8b shows the cross-track errors. The dotted lines show when the waypoints change. The sudden increase in cross-track errors is caused by the switching logic explained in the previous paragraph. The errors then exponentially converge to within 0.2 meters of zero, which is approximately the measurement noise of the navigation system. In addition to measurement noise, the control system is also subject to disturbances caused by the sea loads, and perturbations caused by modeling errors. However, the experiments confirm that the integral state and the overall exponential stability of the controller provide some robustness to these effects.

Chapter 9

Distributed MPC for Formation Path-Following of Multi-Vehicle Systems

The chapter considers the problem of formation path-following of multiple vehicles and proposes a solution based on combining distributed model predictive control with parametrizations of the trajectories of the vehicles using polynomial splines. Introducing such parametrization leads indeed to two potential benefits: a) reducing the number of optimization variables, and b) enabling enforcing constraints on the vehicles in a computationally efficient way. Moreover, the proposed solution formulates the formation path-following problem as a distributed optimization problem that may then be solved using the alternating direction method of multipliers (ADMM). The proposed approach is applicable to all vehicles that can be modeled as differentially flat systems. In this chapter, we present numerical simulations with autonomous underwater vehicles and differential drive robots. The contents of this chapter are based on [84].

The chapter is organized as follows. In Section 9.1, we present the general assumptions on the model of the vehicles and formally define the formation path-following problem. In Section 9.2, we propose the distributed spline-based MPC scheme. Finally, Section 9.3 presents two numerical case studies.

9.1 Problem Description

In this section, we first introduce the assumptions on the model of the vehicles. Then, we define the objective of formation path following and pose it as an optimization problem.

9.1.1 Vehicle Model

As mentioned in the introduction to this chapter, the proposed MPC algorithm can be applied to a wide range of vehicles, not just AUVs. Consequently, the model presented in this section is more general than the hand position model presented in Chapter 7. A case study showing how the theory developed in this chapter can be applied to AUVs will be presented in Section 9.3.1.

Here, we discuss the dynamics of a single agent in the network. Let $\mathbf{x} \in \mathbb{R}^{n_\mathbf{x}}$ be the vector of states. We assume that the state vector includes the position of the agent. Without loss of generality, let the first $n_\mathbf{q}$ states be the position of the vehicle. We can then define the position vector of the vehicle as

$$\mathbf{q} = \left[x_1, \ldots, x_{n_\mathbf{q}}\right]^\mathrm{T}. \tag{9.1}$$

Since vehicles typically move in either two or three dimensions, we assume that $n_\mathbf{q} \in \{2, 3\}$. Let $\mathbf{u} \in \mathbb{R}^{n_\mathbf{q}}$ be the vector of control inputs. We assume the dynamics of the vehicle to be given by an ordinary differential equation

$$\dot{\mathbf{x}} = f\left(\mathbf{x}, \mathbf{u}\right). \tag{9.2}$$

Note that we assume the number of inputs to be equal to $n_\mathbf{q}$. In cases where this assumption does not hold because the vehicle is overactuated, we need to reduce the number of inputs by introducing a control allocation scheme (see, *e.g.,* [65]).

Let $\mathbf{y} \in \mathbb{R}^{n_\mathbf{q}}$ be the output of the system. Note that, in general, the output can be different from the position of the vehicle. However, as discussed in the next paragraph, it must be possible to obtain the position from the output.

We assume the system to be differentially flat [123], *i.e.*, we assume that the input and state can be determined from the output, its derivatives, and antiderivatives. Moreover, we assume that the relation between the output and the position is polynomial. In other words, there exist suitable (nonlinear) functions $\boldsymbol{\phi}_\mathbf{u}$ and $\boldsymbol{\phi}_\mathbf{x}$, and a multidimensional polynomial function $\boldsymbol{\phi}_\mathbf{q}$ of suitable dimensions such that, at any time,

$$\mathbf{u} = \boldsymbol{\phi}_\mathbf{u}\left(\mathbf{y}, \dot{\mathbf{y}}, \ddot{\mathbf{y}}, \ldots, \mathbf{y}^{(r')}\right), \tag{9.3}$$

$$\mathbf{x} = \boldsymbol{\phi}_\mathbf{x}\left(\mathbf{y}^{(-r'')}, \ldots, \mathbf{y}, \dot{\mathbf{y}}, \ddot{\mathbf{y}}, \ldots, \mathbf{y}^{(r')}\right), \tag{9.4}$$

$$\mathbf{q} = \boldsymbol{\phi}_\mathbf{q}\left(\mathbf{y}^{(-r'')}, \ldots, \mathbf{y}, \dot{\mathbf{y}}, \ddot{\mathbf{y}}, \ldots, \mathbf{y}^{(r')}\right), \tag{9.5}$$

where r' and r'' are positive integers.

To model the constraints on the dynamics of the vehicle, we use a multidimensional function $\mathbf{e}$. The set of feasible states and inputs is given by

$$\left\{ \left(\mathbf{x}, \mathbf{u}\right) \middle| \mathbf{e}(\mathbf{x}, \mathbf{u}) \geq \mathbf{0} \right\} \tag{9.6}$$

where the inequality is defined component-wise. We assume that substituting (9.3), (9.4) into (9.6) yields a set of polynomial constraints. In other words, we assume that there exists a multidimensional polynomial function $\mathbf{h}$ such that

$$\mathbf{e}(\mathbf{x}, \mathbf{u}) \geq \mathbf{0} \iff \mathbf{h}\left(\mathbf{y}^{(-r'')}, \ldots, \mathbf{y}^{(r')}\right) \geq \mathbf{0}. \tag{9.7}$$

9.1.2 Formation Path-Following Problem

The goal is to control N vehicles, all subject to the dynamics introduced in Section 9.1.1, so that they move in a prescribed formation while their barycenter

follows a given path. Let $\mathbf{p}_p : \mathbb{R} \to \mathbb{R}^{n_\mathbf{q}}$ be a parametrization by arc length that is continuously differentiable. This implies that for every path point $\mathbf{p}(s)$, we can define a path-tangential coordinate frame and a corresponding rotation matrix $\mathbf{R}_p(s)$ between the inertial and path-tangential frames (see Section 2.2).

The vehicles should converge to a dynamic formation that rotates with the desired path (see Section 2.3.2). Let $\mathbf{p}^f_{f,1}, \ldots, \mathbf{p}^f_{f,N}$ be the position vectors that represent the desired formation. Using this notation, the desired trajectory for agent i is given by

$$\mathbf{q}_{d,i}(s) = \mathbf{p}_p(s) + \mathbf{R}_p(s)\, \mathbf{p}^f_{f,i}, \tag{9.8}$$

for a given s.

The objective of the control system is to steer the actual vehicle positions $\mathbf{q}_i(t)$ to follow the desired trajectories $\mathbf{q}_{d,i}$. Ideally, this means that for a given function $s(t)$ we seek the actual positions to be such that

$$\mathbf{q}_i(t) \to \mathbf{q}_{d,i}\big(s(t)\big), \quad \forall i = 1, \ldots, N. \tag{9.9}$$

Similarly to the NSB algorithms in Part I, the path parameter $s(t)$ can be treated as an additional *degree of freedom* when designing the controller. Consequently, we also need to find a suitable control law for $s(t)$. For this purpose, let U_d be the desired speed of the barycenter of the formation. If the vehicles follow the path perfectly, the actual speed of the barycenter is given by

$$U(t) = \left\|\dot{\mathbf{p}}_p\big(s(t)\big)\right\| = \left\|\frac{\partial \mathbf{p}_p(s)}{\partial s}\, \dot{s}(t)\right\| = \left|\dot{s}(t)\right|. \tag{9.10}$$

The equivalence above implies that the path parameter $s(t)$ should thus be chosen such that

$$\dot{s} \to U_d. \tag{9.11}$$

9.1.3 A Centralized Solution

The problem of finding for each agent i its actuation signal $\mathbf{u}_i(t)$ that guarantees following the desired path $\mathbf{q}_{d,i}\big(s(t)\big)$ as close as possible can thanks to (9.3)–(9.5) be transformed into the problem of finding a corresponding output trajectory $\mathbf{y}_i(t)$.

In general, it is not possible to find an output trajectory $\mathbf{y}_i(t)$ such that (9.9) is satisfied, since the dynamics of the agents are constrained by both (9.2) and (9.7). This means that at any time t, there is a position error

$$\widetilde{\mathbf{q}}_i(t) = \mathbf{q}_i(t) - \mathbf{q}_{d,i}\big(s(t)\big). \tag{9.12}$$

Thanks to (9.5), we can express $\widetilde{\mathbf{q}}_i(t)$ in terms of $\mathbf{y}_i(t)$

$$\widetilde{\mathbf{q}}_i(t) = \boldsymbol{\phi}_\mathbf{q}\left(\mathbf{y}_i^{(-r'')}(t), \ldots, \mathbf{y}_i^{(r')}(t)\right) - \mathbf{q}_{d,i}\big(s(t)\big), \tag{9.13}$$

and thus solve the problem by optimizing $\mathbf{y}_i(t)$ and $s(t)$.

The problem should be cast in a receding horizon fashion to reject potential disturbances as the mission proceeds. We thus propose to formulate the *centralized*

problem of optimizing a part of the trajectory, i.e., $\{\mathbf{y}_i(t:t+T)\}$, $s(t:t+T)$, as that of solving the following constrained problem

$$\begin{aligned}\underset{\{\mathbf{y}_i(t:t+T)\},s(t:t+T)}{\text{minimize}} \sum_{i=1}^{N} \int_t^{t+T} \widetilde{\mathbf{q}}_i^{\mathrm{T}}(\tau)\, \boldsymbol{Q}_{\mathbf{p}}\, \widetilde{\mathbf{q}}_i(\tau)\mathrm{d}\tau \\ + \int_t^{t+T} Q_s\, \left(\dot{s}(\tau) - U_d\right)^2 \mathrm{d}\tau,\end{aligned} \tag{9.14}$$

with T being the prediction horizon, $\boldsymbol{Q}_{\mathbf{p}}$ and Q_s positive weight matrices, $\widetilde{\mathbf{q}}_i$ the position error as defined in (9.13), and subject to, for every agent $i = 1, \ldots, N$, to the constraints C1 to C3 below:

C1 the implicit constraint on the inputs and states, *i.e.*,

$$\mathbf{h}\left(\mathbf{y}_i^{(-r'')}(\tau), \ldots, \mathbf{y}_i^{(r')}(\tau)\right) \geq \mathbf{0}, \quad \forall \tau \in [t, t+T],$$

C2 the constraint on the initial condition of the state of the system, i.e.,

$$\boldsymbol{\phi}_{\mathbf{x}}\left(\mathbf{y}_i^{(-r'')}(t), \ldots, \mathbf{y}_i^{(r')}(t)\right) = \mathbf{x}_i(t),$$

C3 the constraint on the initial condition of the path of the agents, *i.e.*, $s(t)$. In other words, $s(t)$ is not a decision variable, while $s(t+\tau)$ for any $\tau > 0$ is.

We note that the variational problem above may not be solvable using off-the-shelf hardware with limited computing power. For this reason, it will be rewritten below.

9.1.4 A Distributed Solution

To make the centralized approach from the previous section distributed, we assume *synchronous bidirectional reliable* communication. In other words, we assume that all vehicles exchange information simultaneously and there are no packet losses. Bidirectional communication implies that the communication network can be described by an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \ldots, N\}$ correspond to the agents, and $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ represents the communication between pairs of agents. We further assume that $\mathcal{G}$ is connected. Similarly to Chapter 6, we denote the set of neighbors of agent i as $\mathcal{N}_i$.

Before doing the rewriting, we note that it is possible to make (9.14) distributed by letting the path parameter $s(t)$ be a local variable (*i.e.*, $s_i(t)$), and adding a synchronization constraint on the set of $s_i(t)$'s. This leads to the local reformulation

$$\begin{aligned}\underset{\{\mathbf{y}_i(t:t+T),s_i(t:t+T)\}}{\text{minimize}} & \int_t^{t+T} \widetilde{\mathbf{q}}_i^{\mathrm{T}}(\tau)\, \boldsymbol{Q}_{\mathbf{p}}\, \widetilde{\mathbf{q}}_i(\tau)\mathrm{d}\tau \\ & + \int_t^{t+T} Q_s\, \left(\dot{s}_i(\tau) - U_d\right)^2 \mathrm{d}\tau, \end{aligned} \tag{9.15a}$$

$$\text{subject to } h\left(\mathbf{y}_i^{(-r'')}(\tau), \ldots, \mathbf{y}_i^{(r')}(\tau)\right) \geq \mathbf{0}, \tag{9.15b}$$

$$\boldsymbol{\phi}_{\mathbf{x}}\left(\mathbf{y}_i^{(-r'')}(t), \ldots, \mathbf{y}_i^{(r')}(t)\right) = \mathbf{x}_i(t), \tag{9.15c}$$

$$s_i(\tau) = s_j(\tau), \quad \forall i \in \mathcal{V}, \forall j \in \mathcal{N}_i, \forall \tau \in [t, t+T] \tag{9.15d}$$

where j is the index of the generic neighbor of agent i. This formulation is again only an intermediate step towards the approach proposed in this chapter, as explained below.

9.2 A Distributed Spline-Based MPC Solution

The goal of this section is to show how constraining $\mathbf{y}_i$ and s_i to be splines enables rewriting the variational problem above in a way that is computationally tractable.

For the sake of readability, we will use the convention for which sans-serif fonts (*e.g.*, $\mathbf{y}$) indicate quantities relative to splines, while serif fonts (*e.g.*, $\mathbf{y}$) indicate trajectories parametrized in time as above.

9.2.1 Spline Parametrization

Let $\mathbf{b} = [\mathsf{b}_1, \ldots, \mathsf{b}_n]^{\mathrm{T}}$ be the vector of basis functions of a B-spline, and let $\mathbf{y}_i = \left[\mathbf{y}_{i,1}^{\mathrm{T}}, \ldots, \mathbf{y}_{i,n}^{\mathrm{T}}\right]^{\mathrm{T}}$ be a generic matrix and $\mathbf{s}_i = [\mathsf{s}_{i,1}, \ldots, \mathsf{s}_{i,n}]^{\mathrm{T}}$ a generic vector of spline coefficients. Assume then that the trajectories and path parameters may be expressed as B-splines, *i.e.,* as

$$\mathbf{y}_i(\tau) = \sum_{l=1}^{n} \mathbf{y}_{i,l}\, \mathsf{b}_l(\tau) = \mathbf{y}_i^{\mathrm{T}}\, \mathbf{b}(\tau), \tag{9.16}$$

$$s_i(\tau) = \sum_{l=1}^{n} \mathsf{s}_{i,l}\, \mathsf{b}_l(\tau) = \mathbf{s}_i^{\mathrm{T}}\, \mathbf{b}(\tau). \tag{9.17}$$

This assumption implies the possibility of exploiting the convex hull property

$$\mathbf{y}_i \geq \mathbf{0} \implies \mathbf{y}(\tau) \geq \mathbf{0}, \tag{9.18}$$

that implies that any polynomial constraint on a spline can be replaced by a (stricter) constraint on the spline coefficients. In other words, by assuming the output to be a spline, we assume that there exists a function h such that

$$\mathsf{h}\left(\mathbf{y}_i\right) \geq \mathbf{0} \implies h\left(\mathbf{y}_i^{(-r'')}(\tau), \ldots, \mathbf{y}_i^{(r')}(\tau)\right) \geq \mathbf{0}. \tag{9.19}$$

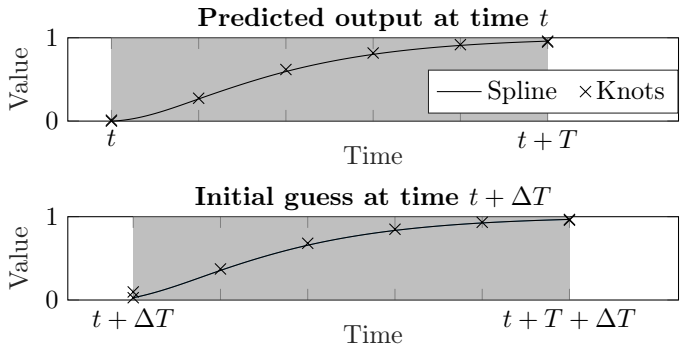


Figure 9.1: Warm-starting the optimization problem. The grey area represents the prediction horizon.

This eventually enables us to rewrite the trajectory optimization problems in Section 9.1 as corresponding spline-based MPC problems.

To do so, each agent i must locally approximate the path function and the associated rotation matrix as polynomials

$$\mathbf{p}_p(s) \approx \mathbf{p}_{p,0} + \mathbf{p}_{p,1}\, s + \ldots + \mathbf{p}_{p,m}\, s^m, \tag{9.20}$$

$$\mathbf{R}_p(s) \approx \mathbf{R}_{p,0} + \mathbf{R}_{p,1}\, s + \ldots + \mathbf{R}_{p,m}\, s^m, \tag{9.21}$$

over an interval $[s_i(t), s_i(t) + s_T]$, where t is the current time and s_T is chosen such that $s_T \geq U_d T$. We then need to impose an additional constraint on the path parameter

$$s_i(t) \leq s_i(\tau) \leq s_i(t) + s_T, \quad \forall \tau \in [t, t+T]\,, \tag{9.22}$$

to ensure that the polynomial approximation is valid.

This approximation transforms the criterion from (9.15a) into a polynomial function. The optimization problem (9.15) can then be reformulated in terms of spline coefficients

$$\underset{\mathbf{y}_i, \mathbf{s}_i}{\text{minimize}}\; J_i\left(\mathbf{y}_i, \mathbf{s}_i\right), \tag{9.23a}$$

$$\text{subject to } \mathbf{y}_i \in \mathsf{Y}_i, \tag{9.23b}$$

$$\mathbf{s}_i \in \mathsf{S}_i, \tag{9.23c}$$

$$\mathbf{s}_i = \mathbf{s}_j, \quad \forall i \in \mathcal{V}, \forall j \in \mathcal{N}_i, \tag{9.23d}$$

where J_i is the objective function from (9.15a), reformulated using the spline coefficients, and Y_i and S_i are the sets of feasible coefficients given by (9.15b), (9.15c) and (9.22).

This optimization problem is then solved in discrete time-steps. Similarly to collocation-based MPC, we can use the results from the previous time-step to "warm-start" the optimization problem. We do this by extrapolating the previous results over the new horizon (see Figure 9.1).

9.2.2 ADMM

We solve the distributed optimization problem (9.23) using the alternating direction method of multipliers (ADMM). In [124], it is discussed that ADMM tends to converge to "modest accuracy" within a few iterations. Due to this property, ADMM is often used to solve distributed MPC problems. Specifically, we use the relaxed ADMM algorithm proposed in [125] to solve the problem.

Relaxed ADMM solves the optimization problem (9.23) by introducing an auxiliary variable $\mathbf{z}_{ji}$ for all $i \in \mathcal{V}, j \in \mathcal{N}_i$. The auxiliary variable $\mathbf{z}_{ji}$ represents agent i's estimate of $\mathbf{s}_j$. The optimization problem (9.23) can then be reformulated as

$$\begin{aligned}
&\underset{\mathbf{y}_i, \mathbf{s}_i, \mathbf{z}_{ji}}{\text{minimize}} && J_i\left(\mathbf{y}_i, \mathbf{s}_i\right), && \text{(9.24a)}\\
&\text{subject to} && \mathbf{y}_i \in \mathsf{Y}_i, && \text{(9.24b)}\\
& && \mathbf{s}_i \in \mathsf{S}_i, && \text{(9.24c)}\\
& && \mathbf{z}_{ji} = \mathbf{s}_j, && \text{(9.24d)}\\
& && \mathbf{z}_{ji} = \mathbf{z}_{ij}, \quad \forall i \in \mathcal{V}, \forall j \in \mathcal{N}_i, && \text{(9.24e)}
\end{aligned}$$

It is then possible to apply the so-called Peaceman-Rachford splitting [126] and solve the optimization problem iteratively. We omit the derivations, as they can be found in [125], and only show the algorithm. The problem is solved iteratively in two steps. First, we compute $\mathbf{y}_i$ and $\mathbf{s}_i$ by solving

$$\mathbf{y}_i, \mathbf{s}_i \leftarrow \underset{\mathbf{y}_i \in \mathsf{Y}_i, \mathbf{s}_i \in \mathsf{S}_i}{\arg\min} \; \mathcal{L}_i\left(\mathbf{y}_i, \mathbf{s}_i, \mathbf{z}_{ji}\right), \tag{9.25}$$

where $\mathcal{L}_i(\mathbf{y}_i, \mathbf{s}_i, \mathbf{z}_{ji})$ is the so-called augmented Lagrangian given by

$$\mathcal{L}_i(\mathbf{y}_i, \mathbf{s}_i, \mathbf{z}_{ji}) = J_i(\mathbf{y}_i, \mathbf{s}_i) - \sum_{j \in \mathcal{N}_i} \mathbf{z}_{ji}^{\mathrm{T}} \mathbf{s}_i + \frac{\rho}{2} d_i \left\|\mathbf{s}_i\right\|^2, \tag{9.26}$$

where $\rho > 0$ is a penalty weight and d_i is the cardinality of $\mathcal{N}_i$. In the second step, we update the auxiliary variables

$$\mathbf{z}_{ji} \leftarrow (1-\alpha)\mathbf{z}_{ji} + \alpha\left(2\rho\mathbf{s}_j - \mathbf{z}_{ij}\right), \tag{9.27}$$

where $0 < \alpha < 1$ is the step size. To perform this step, each agent $j \in \mathcal{N}_i$ sends a packet

$$\mathbf{w}_{ji} = 2\rho\mathbf{s}_i - \mathbf{z}_{ji}, \tag{9.28}$$

to agent i. The update law (9.27) then becomes

$$\mathbf{z}_{ji} \leftarrow (1-\alpha)\mathbf{z}_{ji} + \alpha\mathbf{w}_{ji}. \tag{9.29}$$

To further reduce the needed communication bandwidth, we only perform one ADMM iteration per MPC step. An overview of the resulting distributed MPC is shown in Algorithm 1.

Algorithm 1 ADMM for Distributed MPC

1: **Initialization:** Perform several ADMM iterations (9.25), (9.27) to converge to $\mathbf{y}_i(0), \mathbf{s}_i(0)$ and $\mathbf{z}_{ij}(0)$
2: **for** $k = 1, 2, \ldots$ **do** every ΔT
3: Use extrapolation (see Figure 9.1) to provide an initial guess for $\mathbf{y}_i(k\Delta T), \mathbf{s}_i(k\Delta T)$ and $\mathbf{z}_{ij}(k\Delta T)$
4: Perform one ADMM iteration (9.25), (9.27)
5: **end for**

9.3 Case Studies

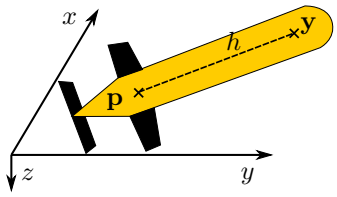
In this section, we demonstrate the proposed MPC scheme on autonomous underwater vehicles (AUVs) and differential drive robots. In both cases, we simulate six vehicles, the prediction horizon is set to 50 seconds, and the path parameter and outputs are represented by cubic splines with 11 breakpoints. Consequently, each spline is represented by 13 coefficients.

9.3.1 AUVs

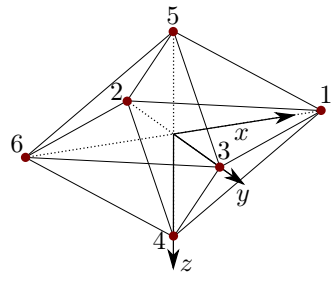
In the first case study, we consider AUVs with six degrees of freedom. Because the vehicle is underactuated (second-order nonholonomic), we cannot use the origin of the body-fixed frame $\mathbf{p}$ as the output of our system. Instead, we choose the hand position defined in Chapter 7 as the output (see Figure 9.2a). Using output-feedback linearization, we can simplify the system to a double integrator

$$\ddot{\mathbf{y}} = \mathbf{u}. \tag{9.30}$$

Remark. *In Chapters 7 and 8, we assumed that only the relative velocities of the AUV are known. Here, we assume that the absolute velocities of the AUV are known as well. This assumption implies that the ocean current* $\mathbf{V}_c$ *can either be measured or estimated. We note that there exist methods for estimating the ocean current,* e.g., *[127].*



(a) Output of the AUV model.



(b) Desired formation.

Figure 9.2: Illustration of the case study with autonomous underwater vehicles (AUVs).

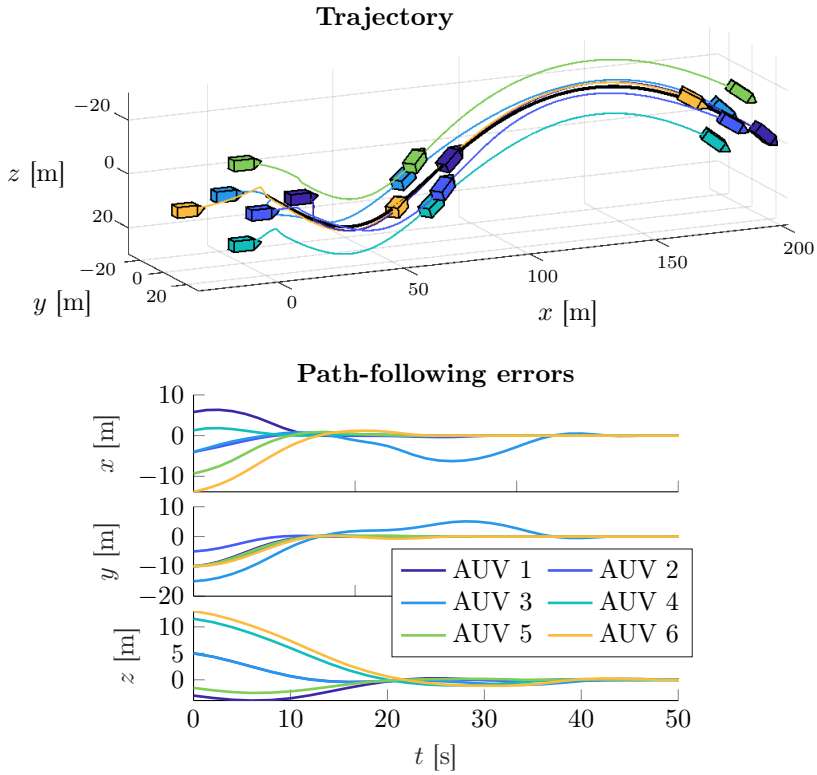


Figure 9.3: Results of numerical simulations with six marine vehicles.

Having transformed the system model into the required form, we validated the proposed method in numerical simulations. The simulations were carried out on a 6DOF model of the light autonomous underwater vehicle (LAUV). The barycenter should follow a spiral path given by

$$\mathbf{p}_p(s) = [\rho(s), a_p \cos(\rho(s)), b_p \sin(\rho(s))]^{\mathrm{T}}, \tag{9.31}$$

where $a_p = b_p = 20\,\mathrm{m}$, and $\rho(s)$ is a monotonically increasing function designed such that $\mathbf{p}_p(s)$ is a parametrization by arc length. The desired formation is shaped like an octahedron; the relative position vectors are given by

$$\begin{bmatrix} \mathbf{p}^f_{f,1} & \cdots & \mathbf{p}^f_{f,6} \end{bmatrix} = \begin{bmatrix} a_f & 0 & 0 & 0 & 0 & -a_f \\ 0 & b_f & -b_f & 0 & 0 & 0 \\ 0 & 0 & 0 & c_f & -c_f & 0 \end{bmatrix}, \tag{9.32}$$

where $a_f = 15\,\mathrm{m}$, and $b_f = c_f = 10\,\mathrm{m}$. The adjacency matrix of the communica-

tions graph is given by

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}. \tag{9.33}$$

The desired formation and the communications graph are illustrated in Figure 9.2b. The MPC and ADMM parameters are shown in Table 9.1a.

The results are shown in Figure 9.3. The top plot shows how the vehicles converge to the desired formation. The bottom plot shows the path-following errors $\widetilde{\mathbf{q}}_i$. We only show the first 75 seconds since the errors converge to zero afterwards.

9.3.2 Differential Drive Robots

In the second case study, we consider differential drive robots modeled as unicycles (see Figure 9.4a). The model is given by

$$\dot{x}_1 = u_1 \cos x_3, \tag{9.34a}$$
$$\dot{x}_2 = u_1 \sin x_3, \tag{9.34b}$$
$$\dot{x}_3 = u_2, \tag{9.34c}$$

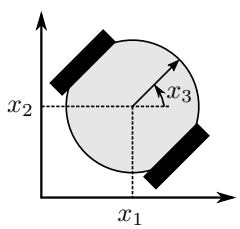
where x_1, x_2 give the position, x_3 is the orientation of the vehicle, and u_1 and u_2 are the tangential and angular velocities.

Similarly to the previous case, we could use the hand position to enable the application of the spline-based MPC. However, doing so would prevent us from imposing constraints on the inputs. Instead, we will use the procedure from [31]. First, we introduce $z = \tan \frac{x_3}{2}$ and use the following trigonometric identities

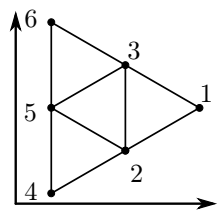
$$\cos x_3 = \frac{1 - z^2}{1 + z^2}, \qquad \sin x_3 = \frac{2z}{1 + z^2}. \tag{9.35}$$

Next, we substitute z and a modified input $\bar{u}_1 = \frac{u_1}{1+z^2}$ into the first two lines of (9.34) to obtain

$$\dot{x}_1 = \bar{u}_1 \left(1 - z^2\right), \qquad \dot{x}_2 = 2\bar{u}_1 z. \tag{9.36}$$



(a) Unicycle model.



(b) Desired formation.

Figure 9.4: Illustration of the case study with differential drive robots.

We choose $\mathbf{y} = [\bar{u}_1, z]^{\mathrm{T}}$ as the output of the system. The states and inputs can then be expressed as

$$x_1(t) = \int_0^t y_1(\tau)\left(1 - y_2^2(\tau)\right)\,\mathrm{d}\tau + x_1(0), \tag{9.37a}$$

$$x_2(t) = \int_0^t 2y_1(\tau)y_2(\tau)\,\mathrm{d}\tau + x_2(0), \tag{9.37b}$$

$$x_3(t) = 2\arctan y_2(t), \tag{9.37c}$$

$$u_1(t) = y_1(t)\left(1 + y_2^2(t)\right), \tag{9.37d}$$

$$u_2(t) = \frac{2\dot{y}_2(t)}{1 + y_2^2(t)}. \tag{9.37e}$$

Let us assume that there are no constraints on the states and the constraints on the inputs are given by

$$u_{1,\min} \leq u_1(t) \leq u_{1,\max}, \qquad u_{2,\min} \leq u_2(t) \leq u_{2,\max}, \tag{9.38}$$

From (9.37d), (9.37e), the constraints can be expressed as

$$u_{1,\min} \leq y_1(t)\left(1 + y_2^2(t)\right) \leq u_{1,\max}, \tag{9.39a}$$

$$u_{2,\min}\left(1 + y_2^2(t)\right) \leq 2\dot{y}_2(t) \leq u_{2,\max}\left(1 + y_2^2(t)\right). \tag{9.39b}$$

We have thus shown how to express the states, inputs and constraints in terms of the outputs.

Having transformed the system model into the required form, we validated the proposed method in numerical simulations. The barycenter should follow a sine wave given by

$$\mathbf{p}_p(s) = [\rho(s), a_p \sin(\rho(s))]^{\mathrm{T}}, \tag{9.40}$$

where $a_p = 15\,\mathrm{m}$, and $\rho(s)$ is a monotonically increasing function designed such that $\mathbf{p}_p(s)$ is a parametrization by arc length. The desired formation is shaped like an equilateral triangle; the relative position vectors are given by

$$\begin{bmatrix}\mathbf{p}_{f,1}^f & \cdots & \mathbf{p}_{f,6}^f\end{bmatrix} = \begin{bmatrix}4a_f & a_f & a_f & -2a_f & -2a_f & -2a_f \\ 0 & -b_f & b_f & -2b_f & 0 & 2b_f\end{bmatrix}, \tag{9.41}$$

where $a_f = \frac{5\sqrt{3}}{3}\,\mathrm{m}$, and $b_f = 5\,\mathrm{m}$. The adjacency matrix of the communications graph is given by

$$\mathbf{A} = \begin{bmatrix}0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0\end{bmatrix}. \tag{9.42}$$

The desired formation and the communications graph are illustrated in Figure 9.4b. The MPC and ADMM parameters are shown in Table 9.1b.

The results of numerical simulations are shown in Figure 9.5. Due to the numerical inaccuracies caused by (9.37) and arising primarily from the multiplication and division of splines, the MPC time-step ΔT must be shorter than in the previous case-study.

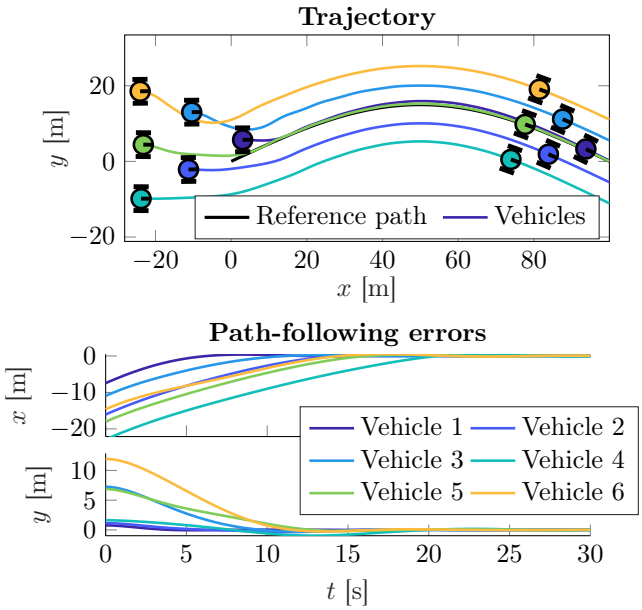


Figure 9.5: Results of numerical simulations with six differential drive robots.

Table 9.1: Simulation parameters

(a) Marine vehicles

Parameter	Value
ΔT	1
$\boldsymbol{Q}_p$	$\mathbf{I}_3$
Q_s	10
ρ	10
α	0.6
h	1
$\mathbf{V}_c$	$\begin{bmatrix} 0.15 \\ 0.1 \\ -0.05 \end{bmatrix}$

(b) Differential drive robots

Parameter	Value
ΔT	0.1
$\boldsymbol{Q}_p$	$\mathbf{I}_2$
Q_s	10
ρ	10
α	0.6
$u_{1,\min}$	-1
$u_{1,\max}$	2
$u_{2,\min}$	$-\pi/8$
$u_{2,\max}$	$\pi/8$

Chapter 10

Control of AUVs Under Hard and Soft Constraints

This chapter investigates the tracking-in-formation problem for a group of underactuated autonomous marine vehicles interconnected over a directed topology. The agents are subject to *hard* inter-agent constraints, *i.e.,* connectivity maintenance and collision avoidance, and *soft* constraints, specifically on the non-negativity of the surge velocity, as well as to constant disturbances in the form of unknown ocean currents. The control approach is based on two concepts: the 3D hand position output linearization presented in Chapter 7, and the edge-based framework for multi-agent consensus under constraints. We establish *almost-everywhere* uniform asymptotic stability of the output dynamics with guaranteed respect of the constraints. Numerical and high-fidelity simulations are provided to illustrate the effectiveness of our approach. The contents of this chapter are based on [77, 88].

The chapter is organized as follows. In Section 10.1 we present the model of the multi-agent system and the problem formulation. For clarity of exposition, in Section 10.2 we present the control design when considering only the hard constraints, followed by the stability analysis. Then, in Section 10.3 we present the control design adding the soft constraints. Finally, the results of numerical simulations are presented in Section 10.4.

10.1 Model and Problem Formulation

10.1.1 Model of the Marine Vehicle

We consider underactuated AUVs with six DOFs, and apply the hand position transformation from Chapter 7. Recalling (7.11), the dynamics of the transformed system are

$$\dot{\mathbf{x}}_1 = \mathbf{x}_2 + \mathbf{V}_c, \tag{10.1a}$$

$$\dot{\mathbf{x}}_2 = \boldsymbol{\mu}, \tag{10.1b}$$

$$\dot{\mathbf{R}} = \mathbf{R}\mathbf{S}(\boldsymbol{\omega}), \tag{10.1c}$$

$$\begin{aligned}\dot{\boldsymbol{\omega}} = \bar{\boldsymbol{\ell}} \times \left(\mathbf{R}^{\mathrm{T}}\boldsymbol{\mu} + \mathcal{D}_{\boldsymbol{v}}(\boldsymbol{\nu}) + \mathcal{C}_{\boldsymbol{v}}(\boldsymbol{\nu}) - \boldsymbol{\omega} \times \mathbf{R}^{\mathrm{T}}\mathbf{x}_2\right) \\ - \left(\bar{\boldsymbol{\ell}}\boldsymbol{\ell}^{\mathrm{T}}\right)\left(\mathcal{D}_{\boldsymbol{\omega}}(\boldsymbol{\nu}) + \mathbf{M}'_{22}\left(W z_{gb}\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right)\right).\end{aligned} \tag{10.1d}$$

10.1.2 Problem Statement

We consider a multi-agent system composed of N marine vehicles modeled by (10.1). The states of the vehicles are denoted by double subscripts (*e.g.,* the hand position of vehicle i is $\mathbf{x}_{1,i}$). The interaction between the AUVs is given by a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ which has either a spanning tree or a cycle. Moreover, we consider that the multi-agent system is subject to inter-agent output constraints. For one part, these constraints may come from embedded relative-measurements devices, which are reliable only if used within a limited range. Hence, the vehicles must remain within a limited distance from their neighbors in order to maintain the connectivity of the graph. Furthermore, to ensure the safety of the system, the agents must avoid collisions among themselves, that is, they must always guarantee a minimal distance with respect to their neighbors. These connectivity and collision-avoidance constraints may be defined as a set of restrictions on $\mathbf{x}_1$. Also, such constraints may be considered as *hard* constraints since they are fundamental for ensuring the safety of the system and for reaching the control goal.

More precisely, define the relative output

$$\mathbf{z}_{1,k} = \mathbf{x}_{1,i} - \mathbf{x}_{1,j}, \quad \forall k \leq M, \quad e_k = (i,j) \in \mathcal{E}, \tag{10.2}$$

where M is the cardinality of $\mathcal{E}$. For each $k \leq M$, let δ_k and Δ_k be, the minimal and maximal distances between agents i and j so that collisions are avoided and that the communication through edge e_k is reliable, respectively. Then, the set of inter-agent output constraints is defined as

$$\mathcal{D}_k = \left\{ \mathbf{z}_{1k} \in \mathbb{R}^3 \,|\, \delta_k < \|\mathbf{z}_{1k}\| < \Delta_k \right\}, \quad \forall\, k \leq M. \tag{10.3}$$

Coming back to each individual agent, it is important to note that, in practice, marine vehicles are not optimized for moving backwards. However, backwards motion is not prevented from the dynamical model (10.1). Therefore, in order to let the vehicles evolve in an optimal way, besides the inter-agent connectivity and collision-avoidance constraints, we could formulate the additional constraints

$$u_{r,i}(t) > 0, \quad \forall\, i \leq N, \quad \forall t \geq 0. \tag{10.4}$$

However, in some cases, the constraints (10.4) could conflict with the constraints defined by the set (10.3). Indeed, there might exist situations when the only way to avoid a collision or avoid losing connectivity is to move backwards. Moreover, although not optimized to, marine vehicles *can* move backwards. The latter fact motivates us to reformulate the constraints (10.4) as *soft* constraints, that is, to impose a positive surge velocity as long as this does not interfere with the satisfaction of the hard constraints (10.3). We formulate these soft constraints as follows:

$$u_{r,i}(t) + \rho_i(t) > 0, \quad \forall\, i \leq N, \quad \forall t \geq 0, \tag{10.5}$$

where $\rho_i : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ will be defined later, such that $\rho_i(t) \cong 0$ when there are no conflicts with the hard constraints and $\rho_i(t) > 0$ otherwise, allowing $u_{r,i}(t)$ to

become negative. Akin to (10.3), we may define the set of soft constraints as

$$\mathcal{C}_i = \left\{ u_{r,i} \in \mathbb{R} \,|\, u_{r,i} > -\rho_i(t) \right\}, \quad \forall\, i \leq N. \tag{10.6}$$

Now, let $\mathbf{x}_{1,o}$, $\mathbf{x}_{2,o}$, and $\boldsymbol{\mu}_o$ define the position, velocity, and acceleration of a virtual target, and let its dynamics be modeled as a second-order integrator

$$\dot{\mathbf{x}}_{1,o} = \mathbf{x}_{2,o}, \quad \dot{\mathbf{x}}_{2,o} = \boldsymbol{\mu}_o(t). \tag{10.7}$$

Moreover, assume the following.

Assumption 10.1. *For all t, there exist positive constants $\underline{x}_{2,o}$, $\overline{x}_{2,o}$, and $\overline{\mu}_o$ such that $\|\mathbf{V}_c\| < \underline{x}_{2,o} \leq \|\mathbf{x}_{2,o}(t)\| \leq \overline{x}_{2,o}$, and $\|\boldsymbol{\mu}_o(t)\| \leq \overline{\mu}_o$.*

Then, the control goal is for the N marine vehicles to achieve the desired formation and track the target modeled by (10.7), all while guaranteeing that the hard constraints given by the set $\mathcal{D}_k$ in (10.3) and the soft constraints (10.5) are respected.

For the tracking part of the problem, we consider the case that only one agent, labeled $i = 1$ without loss of generality, has access to the state of the target, and knows the upper bound $\overline{\mu}_o$ on the target's acceleration. Let $\mathbf{z}_{1,o}^d \in \mathbb{R}^3$ be the desired displacement with respect to the target. Then, we define the tracking error states as

$$\tilde{\mathbf{z}}_{1,o} = \mathbf{x}_{1,1} - \mathbf{x}_{1,o} - \mathbf{z}_{1,o}^d, \quad \mathbf{z}_{2,o} = \mathbf{x}_{2,1} - \mathbf{x}_{2,o}. \tag{10.8}$$

To address the formation part, we rely on the edge-agreement framework [78] where instead of considering the states of each individual agent (the nodes of the graph), we consider the variables $\mathbf{z}_{1,k}$ defined in (10.2) which correspond to the edges in the graph. Hence, let us denote by $\mathbf{E} \in \mathbb{R}^{N \times M}$ the incidence matrix of graph $\mathcal{G}$, where its (i,k)th entry is defined as follows: $[\mathbf{E}]_{ik} = -1$ if i is the terminal node of edge e_k, $[\mathbf{E}]_{ik} = 1$ if i is the initial node of edge e_k, and $[\mathbf{E}]_{ik} = 0$ otherwise. Let $\mathbf{x}_1^{\mathrm{T}} = \begin{bmatrix} \mathbf{x}_{1,1}^{\mathrm{T}} & \cdots & \mathbf{x}_{1,N}^{\mathrm{T}} \end{bmatrix}$ be the collection of the hand-position coordinates of all the agents of the system, and let $\mathbf{z}_1^{\mathrm{T}} = [\mathbf{z}_{1,1}^{\mathrm{T}} \cdots \mathbf{z}_{1,M}^{\mathrm{T}}]^{\mathrm{T}}$ be the collection of all the relative positions between the pairs of neighboring agents. Then, we can express the edge states in the following compact form

$$\mathbf{z}_1 = [\mathbf{E}^{\mathrm{T}} \otimes \mathbf{I}_3]\mathbf{x}_1. \tag{10.9}$$

The formation error, in turn, is defined as

$$\tilde{\mathbf{z}}_1 = [\mathbf{E}^{\mathrm{T}} \otimes \mathbf{I}_3]\mathbf{x}_1 - \mathbf{z}_1^d, \qquad \mathbf{z}_1^{d,\mathrm{T}} = \begin{bmatrix} \mathbf{z}_{1,1}^{d\mathrm{T}} \cdots \mathbf{z}_{1,M}^{d\mathrm{T}} \end{bmatrix} \tag{10.10}$$

where $\mathbf{z}_{1,k}^d \in \mathbb{R}^3$ denotes the desired relative position between a pair of neighboring agents over edge e_k. In the same way, let $\mathbf{x}_2^{\mathrm{T}} = \begin{bmatrix} \mathbf{x}_{2,1}^{\mathrm{T}} & \cdots & \mathbf{x}_{2,N}^{\mathrm{T}} \end{bmatrix}$ be the collection of the hand-position velocities. Then, the relative hand-position velocities in the edge coordinates, $\mathbf{z}_2$, are given by

$$\mathbf{z}_2 = [\mathbf{E}^{\mathrm{T}} \otimes \mathbf{I}_3]\mathbf{x}_2. \tag{10.11}$$

Let us also define the collection of the control inputs $\boldsymbol{\mu}^{\mathrm{T}} = \begin{bmatrix}\boldsymbol{\mu}_1^{\mathrm{T}} & \cdots & \boldsymbol{\mu}_N^{\mathrm{T}}\end{bmatrix}$. Then, mathematically, the tracking-in-formation problem translates into designing a distributed controller $\boldsymbol{\mu}$ such that

$$\lim_{t\to\infty} \tilde{\mathbf{z}}_{1,o}(t) = \mathbf{0} \qquad \lim_{t\to\infty} \mathbf{z}_{2,o}(t) = \mathbf{0} \tag{10.12a}$$

$$\lim_{t\to\infty} \tilde{\mathbf{z}}_{1}(t) = \mathbf{0} \qquad \lim_{t\to\infty} \mathbf{z}_{2}(t) = \mathbf{0}. \tag{10.12b}$$

Now, as observed in [128], using an appropriate labeling of the edges, the incidence matrix can be expressed as $\mathbf{E} = \begin{bmatrix}\mathbf{E}_t & \mathbf{E}_c\end{bmatrix}$, where $\mathbf{E}_t \in \mathbb{R}^{N\times(N-1)}$ denotes the full column-rank incidence matrix corresponding to an arbitrary spanning tree $\mathcal{G}_t \subset \mathcal{G}$ and $\mathbf{E}_c \in \mathbb{R}^{N\times(M-N+1)}$ represents the incidence matrix corresponding to the remaining edges in $\mathcal{G} \setminus \mathcal{G}_t$. Similarly, the error edge states may be split as $\mathbf{z}_\iota = \begin{bmatrix}\mathbf{z}_{\iota,t}^{\mathrm{T}} & \mathbf{z}_{\iota,c}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$, $\iota = \{1,2\}$, where $\mathbf{z}_{\iota,t} \in \mathbb{R}^{3(N-1)}$ are the states corresponding to the edges of $\mathcal{G}_t$ and $\mathbf{z}_{\iota,c} \in \mathbb{R}^{n(M-N+1)}$ denote the states of the remaining edges, $\mathcal{G}\setminus\mathcal{G}_t$. Note that the spanning tree and the error variables associated with its edges are sufficient to describe the errors of the multi-agent system. Indeed, defining the following transformation matrix

$$\mathcal{R} = \begin{bmatrix}\mathbf{I}_{N-1} & \mathbf{T}\end{bmatrix}, \quad \mathbf{T} = \left(\mathbf{E}_t^{\mathrm{T}}\mathbf{E}_t\right)^{-1}\mathbf{E}_t^{\mathrm{T}}\mathbf{E}_c, \tag{10.13}$$

we get the following identities

$$\mathbf{E} = \mathbf{E}_t\mathcal{R}, \quad \mathbf{z}_\iota = \left[\mathcal{R}^{\mathrm{T}} \otimes \mathbf{I}_3\right]\tilde{\mathbf{z}}_{\iota,t}, \ \ \iota \in \{1,2\}. \tag{10.14}$$

We have thus shown that is is possible to obtain the incidence matrix and the error variables corresponding to the original graph from its spanning tree.

Then, using (10.14) and denoting $\mathbf{z}_{1,t}^d \in \mathbb{R}^{n(N-1)}$ as the vector of desired relative displacements corresponding to $\mathcal{G}_t$, a reduced-order model for the external dynamics in terms of the edges of a spanning tree is given by

$$\dot{\mathbf{z}}_{1,o} = \mathbf{z}_{2,o} + \mathbf{V}_c \tag{10.15a}$$

$$\dot{\tilde{\mathbf{z}}}_{1,t} = \mathbf{z}_{2,t} \tag{10.15b}$$

$$\dot{\mathbf{z}}_{2,o} = \boldsymbol{\mu}_1 - \boldsymbol{\mu}_o(t) \tag{10.15c}$$

$$\dot{\mathbf{z}}_{2,t} = \left[\mathbf{E}_t^{\mathrm{T}} \otimes \mathbf{I}_3\right]\boldsymbol{\mu}. \tag{10.15d}$$

In these coordinates, the control objective as defined in (10.12) is achieved if the origin of system (10.15) is asymptotically stabilized. More precisely, we consider the following problem.

Tracking-in-formation under hard and soft constraints: Consider a system of N autonomous marine vehicles with dynamics given by (10.1), interacting over a directed graph which has either a spanning tree or a cycle. Assume, in addition, that the agents are subject to the hard inter-agent output constraints given by the set defined in (10.3) and the soft constraints given by (10.6). Under these conditions, find distributed controllers $\boldsymbol{\mu}_i$, $i \leq N$, that asymptotically stabilize the origin of (10.15) and render the sets (10.3) and (10.6) forward invariant, *i.e.*, $\mathbf{z}_{1,k}(t_0) \in \mathcal{D}_k$ ($u_{r,i}(t_0) \in \mathcal{C}_i$) implies that $\mathbf{z}_{1,k}(t) \in \mathcal{D}_k$ ($u_{r,i}(t) \in \mathcal{C}_i$), $\forall k \leq M$ ($\forall i \leq N$) and $\forall t \geq t_0$.

10.2 Designing a Tracking Controller under Proximity and Safety Constraints

For clarity of exposition, in this section we begin by presenting the control design considering *only* the hard constraints, *i.e.,* connectivity maintenance and collision avoidance. The inclusion of the soft constraints is addressed in Section 10.3.

We will show how the tracking-in-formation problem, with the previously formulated inter-agent constraints, can be solved following a backstepping approach which is well adapted to the normal form of the external dynamics (10.15). We start by defining a virtual control law for (10.15a)-(10.15b) with $\mathbf{z}_{2,o}$ and $\mathbf{z}_{2,t}$ as inputs. In order to account for the output constraints, a good choice of control design for the virtual inputs consists in using the gradient of a barrier Lyapunov function (BLF) [129].

10.2.1 On Barrier Lyapunov Functions

BLFs are reminiscent of Lyapunov functions in that they are positive definite, but their domain is restricted by design to open subsets of the Euclidean space and they grow unbounded as their argument approaches the boundary of their domain. We define them as follows, *c.f.,* [129].

Definition 10.1 (BLF)**.** *Consider the system $\dot{x} = f(x)$ and let $\mathcal{M} \subset \mathbb{R}^n$ be an open set containing the origin. A BLF is a positive definite function $V : \mathcal{M} \mapsto \mathbb{R}_{\geq 0}$ that satisfies*

$$\nabla V(x)^{\mathrm{T}} f(x) = \frac{\partial V(x)}{\partial x}^{\mathrm{T}} f(x) \leq 0,$$

and $V(x) \to \infty$ and $\|\nabla V(x)\| \to \infty$ as $x \to \partial\mathcal{M}$.

Akin to (10.3), the inter-agent constraints in terms of the formation error are given, for all $k \leq M$, by the set

$$\tilde{\mathcal{D}}_k = \{\tilde{\mathbf{z}}_{1,k} \in \mathbb{R}^3 \,|\, \delta_k < \left\|\tilde{\mathbf{z}}_{1,k} + \mathbf{z}^d_{1,k}\right\| < \Delta_k\}. \tag{10.16}$$

Then, for each $k \leq M$, we define a candidate BLF $W_k : \tilde{\mathcal{D}}_k \mapsto \mathbb{R}_{\geq 0}$, of the form

$$W_k(\tilde{\mathbf{z}}_{1,k}) = \frac{1}{2}\left[\|\tilde{\mathbf{z}}_{1,k}\|^2 + B_k(\tilde{\mathbf{z}}_{1,k} + \mathbf{z}^d_{1,k})\right], \tag{10.17}$$

where

$$B_k(\mathbf{z}_{1,k}) = \kappa_{1,k}\left[\ln\left(\frac{\Delta_k^2}{\Delta_k^2 - \|\mathbf{z}_{1,k}\|^2}\right) - \ln\left(\frac{\Delta_k^2}{\Delta_k^2 - \left\|\mathbf{z}^d_{1,k}\right\|^2}\right)\right]$$
$$+ \kappa_{2,k}\left[\ln\left(\frac{\|\mathbf{z}_{1,k}\|^2}{\|\mathbf{z}_{1,k}\|^2 - \delta_k^2}\right) - \ln\left(\frac{\left\|\mathbf{z}^d_{1,k}\right\|^2}{\left\|\mathbf{z}^d_{1,k}\right\|^2 - \delta_k^2}\right)\right],$$

$$\kappa_{1,k} = \frac{\delta_k^2}{\left\|\mathbf{z}^d_{1,k}\right\|^2\left(\left\|\mathbf{z}^d_{1,k}\right\|^2 - \delta_k^2\right)}, \quad \kappa_{2,k} = \frac{1}{\Delta_k^2 - \left\|\mathbf{z}^d_{1,k}\right\|^2}.$$

Note that B_k is a non-negative function that satisfies: $B_k(\mathbf{z}_{1,k}^d)=0$, $\nabla B_k(\mathbf{z}_{1,k}^d)=\mathbf{0}$, and $B_k(\tilde{\mathbf{z}}_{1,k}+\mathbf{z}_{1,k}^d)\to\infty$ as either $\left\|\tilde{\mathbf{z}}_{1,k}+\mathbf{z}_{1,k}^d\right\|\to\Delta_k$ or $\left\|\tilde{\mathbf{z}}_{1,k}+\mathbf{z}_{1,k}^d\right\|\to\delta_k$. Therefore, the candidate BLF (10.17) satisfies: $W_k(\tilde{\mathbf{z}}_{1,k})\to\infty$ as either $\left\|\tilde{\mathbf{z}}_{1,k}+\mathbf{z}_{1,k}^d\right\|\to\Delta_k$ or $\left\|\tilde{\mathbf{z}}_{1,k}+\mathbf{z}_{1,k}^d\right\|\to\delta_k$.

Remark 10.1. *The function in* (10.17) *is reminiscent of scalar potential functions in constrained environments. Hence, the appearance of multiple critical points is inevitable [130]. Indeed, the gradient of the BLF* (10.17)*, $\nabla W_k(\tilde{\mathbf{z}}_{1,k})$, vanishes at the origin and at an isolated saddle point separated from the origin. Therefore, when using the gradient of* (10.17)*, the closed-loop system has multiple equilibria. We address such technicalities using tools for multi-stable systems [131, 132].*

Now, we define a BLF for the multi-agent system as

$$W(\tilde{\mathbf{z}}_1)=\sum_{k\leq M}\varrho_k W_k(\tilde{\mathbf{z}}_{1,k}),\quad \varrho_k>0, \tag{10.18}$$

and, in light of Remark 10.1, let $\tilde{\mathbf{z}}_1^*\in\mathbb{R}^{3M}$ denote the vector containing the saddle points of the BLF for each edge (10.17). Moreover, let us define the disjoint set

$$\mathcal{W}=\{\mathbf{0}\}\cup\{\tilde{\mathbf{z}}_1^*\}, \tag{10.19}$$

which corresponds to the critical points of W in (10.17). Then, W satisfies

$$\frac{a_1}{2}\left\|\tilde{\mathbf{z}}_1\right\|_{\mathcal{W}}^2\leq W(\tilde{\mathbf{z}}_1)\leq a_2\left\|\nabla W(\tilde{\mathbf{z}}_1)\right\|^2, \tag{10.20}$$

where $a_1,a_2>0$ and $\|\tilde{\mathbf{z}}_1\|_{\mathcal{W}}=\min\left\{\left\|\tilde{\mathbf{z}}_1\right\|,\left\|\tilde{\mathbf{z}}_1-\tilde{\mathbf{z}}_1^*\right\|\right\}$.

10.2.2 Control Design for Systems over Directed Graphs

Let us define the so-called in-incidence matrix $\mathbf{E}_\odot\in\mathbb{R}^{N\times M}$, whose elements are defined as follows: $[\mathbf{E}_\odot]_{i,k}=-1$ if i is the terminal node of edge e_k and $[\mathbf{E}_\odot]_{ik}=0$ otherwise. Then, in the edge-agreement framework, the virtual controllers are

$$\begin{aligned}\mathbf{z}_{2,t}^*&=[\mathbf{E}_t^{\mathrm{T}}\otimes\mathbf{I}_3]\mathbf{x}_2^*\\ \mathbf{x}_2^*&=-c_1[\mathbf{E}_\odot\otimes\mathbf{I}_3]\nabla W(\tilde{\mathbf{z}}_{1,t})-c_1[\mathbf{C}\otimes\mathbf{I}_3]\tilde{\mathbf{z}}_{1,o}-\hat{\mathbf{V}}_c,\end{aligned} \tag{10.21}$$

where c_1 is a positive gain, $\hat{\mathbf{V}}_c$ is a vector of estimates of the ocean current for each agent, and $\mathbf{C}^{\mathrm{T}}=\begin{bmatrix}1 & \mathbf{0}_{N-1}^{\mathrm{T}}\end{bmatrix}$. To avoid a cumbersome notation we write $\nabla W(\tilde{\mathbf{z}}_{1,t})$ in place of the more appropriate spelling $\nabla W\left(\left[\mathcal{R}^{\mathrm{T}}\otimes\mathbf{I}_3\right]\tilde{\mathbf{z}}_{1,t}\right)$.

Defining velocity errors $\tilde{\mathbf{z}}_{2,t}=\mathbf{z}_{2,t}-\mathbf{z}_{2,t}^*$, $\tilde{\mathbf{z}}_{2,o}=\tilde{\mathbf{z}}_{2,1}-\mathbf{z}_{2,o}$, and using (10.21), the subsystem (10.15a)–(10.15b) becomes

$$\begin{aligned}\begin{bmatrix}\dot{\tilde{\mathbf{z}}}_{1,o}\\ \dot{\tilde{\mathbf{z}}}_{1,t}\end{bmatrix}=&-c_1\left[\begin{bmatrix}1 & \mathbf{C}^{\mathrm{T}}\mathbf{E}_t\\ \mathbf{E}_t^{\mathrm{T}}\mathbf{C} & \mathbf{E}_t^{\mathrm{T}}\mathbf{E}_\odot\end{bmatrix}\otimes\mathbf{I}_3\right]\begin{bmatrix}\tilde{\mathbf{z}}_{1,o}\\ \nabla W(\tilde{\mathbf{z}}_1)\end{bmatrix}\\ &+\left[\begin{bmatrix}\mathbf{C}^{\mathrm{T}}\\ \mathbf{E}_t^{\mathrm{T}}\end{bmatrix}\otimes\mathbf{I}_3\right]\tilde{\mathbf{V}}_c+\begin{bmatrix}\tilde{\mathbf{z}}_{2,o}\\ \tilde{\mathbf{z}}_{2,t}\end{bmatrix}\end{aligned} \tag{10.22}$$

where the estimation error $\tilde{\mathbf{V}}_c$ is defined as

$$\tilde{\mathbf{V}}_c = \bar{\mathbf{V}}_c - \hat{\mathbf{V}}_c, \quad \text{with } \bar{\mathbf{V}}_c = \mathbf{1}_N \otimes \mathbf{V}_c. \tag{10.23}$$

With the aim of making $\tilde{\mathbf{V}}_c \to \mathbf{0}$, we design the adaptation law

$$\dot{\hat{\mathbf{V}}}_c = c_v \left(\mathbf{x}_1 - \boldsymbol{\varphi}\right), \quad c_v > 0 \tag{10.24a}$$

$$\dot{\boldsymbol{\varphi}} = \mathbf{x}_2 + \hat{\mathbf{V}}_c. \tag{10.24b}$$

Using (10.24) and (10.1a)–(10.1b), the derivative of (10.23) becomes

$$\dot{\tilde{\mathbf{V}}}_c = -c_v \left(\mathbf{x}_2 + \bar{\mathbf{V}}_c - \mathbf{x}_2 - \hat{\mathbf{V}}_c\right) = -c_v \tilde{\mathbf{V}}_c. \tag{10.25}$$

In the error coordinates $\tilde{\mathbf{z}}_{2,o}$ and $\tilde{\mathbf{z}}_{2,t}$, we have

$$\dot{\tilde{\mathbf{z}}}_{2,o} = \boldsymbol{\mu}_1 - \dot{\mathbf{x}}^*_{2,1} \tag{10.26a}$$

$$\dot{\tilde{\mathbf{z}}}_{2,t} = \left[\mathbf{E}_t^{\mathrm{T}} \otimes \mathbf{I}_3\right]\boldsymbol{\mu} - \dot{\mathbf{z}}^*_{2,t}. \tag{10.26b}$$

Hence, we design the tracking-in-formation control law as

$$\begin{aligned}\boldsymbol{\mu} = &- c_2\left[\mathbf{E}_\odot \mathcal{R}^{\mathrm{T}} \otimes \mathbf{I}_3\right]\tilde{\mathbf{z}}_{2,t} - c_2[\mathbf{C} \otimes \mathbf{I}_3]\tilde{\mathbf{z}}_{2,o} + \dot{\mathbf{x}}^*_2 \\ &- \gamma \mathrm{sign}\left(\left[\mathbf{E}_\odot \mathcal{R}^{\mathrm{T}} \otimes \mathbf{I}_3\right]\tilde{\mathbf{z}}_{2,\mathrm{t}} + [\mathbf{C} \otimes \mathbf{I}_3]\tilde{\mathbf{z}}_{2,\mathrm{o}}\right) + \boldsymbol{\mu}^*(\mathrm{t})\end{aligned} \tag{10.27}$$

where $c_2, \gamma > 0$, $\bar{\mathbf{x}}^*_2 = -c_1[\mathbf{E}_\odot \otimes \mathbf{I}_3]\nabla W(\tilde{\mathbf{z}}_1) - c_1[\mathbf{C} \otimes \mathbf{I}_3]\tilde{\mathbf{z}}_{1,o}$. The signal $\boldsymbol{\mu}^*(t)^{\mathrm{T}} = [\boldsymbol{\mu}^*_1(t) \ \cdots \ \boldsymbol{\mu}^*_N(t)]$, satisfying $\|\boldsymbol{\mu}^*(t)\| \leq \bar{\mu}^*$ for a constant $\bar{\mu}^*$, is given by

$$\boldsymbol{\mu}^*_i(t) = \mathbf{R}_i^{\mathrm{T}} f^*_{u,i} \mathbf{e}_1, \tag{10.28}$$

where $f^*_{u,i}$ is an additional bounded control input that will be used to deal with the soft constraints, *c.f.,* Section 10.3.

10.2.3 Closed-Loop Analysis

First, let us define $\boldsymbol{\varsigma}_1^{\mathrm{T}} = \left[\tilde{\mathbf{z}}_{1,o}^{\mathrm{T}} \ \ \tilde{\mathbf{z}}_{1,t}^{\mathrm{T}}\right]$, and $\boldsymbol{\varsigma}_2^{\mathrm{T}} = \left[\tilde{\mathbf{z}}_{2,o}^{\mathrm{T}} \ \ \tilde{\mathbf{z}}_{2,t}^{\mathrm{T}}\right]$. We note that the control goal defined in (10.12) is equivalent to $\lim_{t\to\infty} \boldsymbol{\varsigma}_1 = \mathbf{0}$, $\lim_{t\to\infty} \boldsymbol{\varsigma}_2 = \mathbf{0}$. Consequently, $\boldsymbol{\varsigma}_1$ and $\boldsymbol{\varsigma}_2$ are valid error variables. From (10.22), (10.26), and (10.27), the closed-loop dynamics are

$$\dot{\boldsymbol{\varsigma}}_1 = - c_1\mathcal{L}_1\bar{\boldsymbol{\varsigma}}_1 + \boldsymbol{\varsigma}_2 + \mathcal{T}_1\tilde{\mathbf{V}}_c, \tag{10.29a}$$

$$\dot{\boldsymbol{\varsigma}}_2 = - c_2\mathcal{L}_2\boldsymbol{\varsigma}_2 + c_v\mathcal{T}_1\tilde{\mathbf{V}}_c + \mathcal{T}_1[\boldsymbol{\mu}^*(t) - \mathbf{1}_N \otimes \boldsymbol{\mu}_o(t)] - \gamma\mathcal{T}_1 \mathrm{sign}\left(\mathcal{T}_2^{\mathrm{T}}\boldsymbol{\varsigma}_2\right), \tag{10.29b}$$

$$\dot{\tilde{\mathbf{V}}}_c = - c_v\tilde{\mathbf{V}}_c, \tag{10.29c}$$

where

$$\bar{\boldsymbol{\varsigma}}_1^{\mathrm{T}} = \left[\tilde{\mathbf{z}}_{1,o} \ \ \nabla W(\tilde{\mathbf{z}}_1)^{\mathrm{T}}\right], \tag{10.30a}$$

$$\mathcal{L}_1 = \begin{bmatrix} 1 & \mathbf{C}^{\mathrm{T}}\mathbf{E}_t \\ \mathbf{E}_t^{\mathrm{T}}\mathbf{C} & \mathbf{E}_t^{\mathrm{T}}\mathbf{E}_\odot \end{bmatrix} \otimes \mathbf{I}_3, \qquad \mathcal{T}_1 = \begin{bmatrix} \mathbf{C}^{\mathrm{T}} \\ \mathbf{E}_t^{\mathrm{T}} \end{bmatrix} \otimes \mathbf{I}_3 \tag{10.30b}$$

$$\mathcal{L}_2 = \begin{bmatrix} 1 & \mathbf{C}^{\mathrm{T}}\mathbf{E}_t \\ \mathbf{E}_t^{\mathrm{T}}\mathbf{C} & \mathbf{E}_t^{\mathrm{T}}\mathbf{E}_\odot\mathcal{R}^{\mathrm{T}} \end{bmatrix} \otimes \mathbf{I}_3, \qquad \mathcal{T}_2 = \begin{bmatrix} \mathbf{C}^{\mathrm{T}} \\ \mathcal{R}\mathbf{E}_\odot^{\mathrm{T}} \end{bmatrix} \otimes \mathbf{I}_3. \tag{10.30c}$$

The first part of the main result is stated as follows:

Proposition 10.1. *Consider N AUVs, each described by the model* (10.1)*, and interconnected over a directed graph which has either a spanning tree or a cycle. Then, under Assumption 10.1 and*

$$\gamma \geq \bar{\mu}^* + \sqrt{2N}\bar{\mu}_o, \tag{10.31}$$

the controller (10.27) *renders the constraints set* (10.3) *forward invariant and guarantees the achievement of the tracking-in-formation objective* (10.12) *for almost all initial conditions such that $\mathbf{z}_{1,k}(t_0) \in \mathcal{D}_k$, for all $k \leq M$.*

Moreover, let us define $\boldsymbol{\xi}_{2,d} = \mathbf{x}_{2,o}$ and $a_x, \bar{\alpha}_y, \bar{\alpha}_z$ in accordance with (7.27)*. The internal dynamics are ultimately bounded for almost all initial conditions if $a_x, \bar{\alpha}_y, \bar{\alpha}_z > 0$.*

Proof. We begin by analyzing the external dynamics (10.29). First define the function

$$W_1(\boldsymbol{\varsigma}_1) = \frac{1}{2}\left\|\tilde{\mathbf{z}}_{1,o}\right\|^2 + W(\tilde{\mathbf{z}}_{1,t}), \tag{10.32}$$

where, with a slight abuse of notation, $\tilde{\mathbf{z}}_{1,t} \mapsto W(\tilde{\mathbf{z}}_{1,t})$ is defined in (10.18). The derivative of W_1 along the trajectories of (10.29a) yields

$$\dot{W}_1(\boldsymbol{\varsigma}_1) = -c_1\bar{\boldsymbol{\varsigma}}_1^{\mathrm{T}}\mathcal{L}_1\bar{\boldsymbol{\varsigma}}_1 + \bar{\boldsymbol{\varsigma}}_1^{\mathrm{T}}\boldsymbol{\varsigma}_2 + \bar{\boldsymbol{\varsigma}}_1^{\mathrm{T}}\mathcal{T}_1\tilde{\mathbf{V}}_c. \tag{10.33}$$

Note that for any directed graph containing a spanning tree, $-\mathbf{E}_t^{\mathrm{T}}\mathbf{E}_{\odot}$ is Hurwitz (*c.f.,* [128, Proposition 1]). Consequently, from (10.30b), we can conclude that there exist $c_1', c_v' > 0$ such that

$$\dot{W}_1(\boldsymbol{\varsigma}_1) \leq -c_1'\left\|\bar{\boldsymbol{\varsigma}}_1\right\|^2 + \left\|\bar{\boldsymbol{\varsigma}}_1\right\|\left\|\boldsymbol{\varsigma}_2\right\| + c_v'\left\|\bar{\boldsymbol{\varsigma}}_1\right\|\left\|\tilde{\mathbf{V}}_c\right\|. \tag{10.34}$$

Moreover, for any directed graph containing a spanning tree, it follows that $-\mathbf{E}_t^{\mathrm{T}}\mathbf{E}_{\odot}\mathcal{R}^{\mathrm{T}}$ is Hurwitz (*c.f.,* [133]). Consequently, we can define the following candidate Lyapunov function

$$W_2(\boldsymbol{\varsigma}_2) = \frac{1}{2}\boldsymbol{\varsigma}_2^{\mathrm{T}}\mathbf{P}\boldsymbol{\varsigma}_2 \tag{10.35}$$

where $\mathbf{P}$ is a positive definite such that for any positive definite $\mathbf{Q}$, it holds that $\mathcal{L}_2^{\mathrm{T}}\mathbf{P} + \mathbf{P}\mathcal{L}_2 = \mathbf{Q}$. Then, the derivative of (10.35) along the trajectories of (10.29b), is defined by the differential inclusion $\boldsymbol{\varsigma}_2 \in F_2(t, \boldsymbol{\varsigma}_2)$, where

$$F_2(t,\boldsymbol{\varsigma}_2) = \begin{cases} (10.29\text{b}), & \text{if } \mathcal{T}_2^{\mathrm{T}}\boldsymbol{\varsigma}_2 \neq \mathbf{0}, \\ -c_2\mathcal{L}_2\boldsymbol{\varsigma}_2 + c_v\mathcal{T}_1\tilde{\mathbf{V}}_c - \gamma\lambda + \mathcal{T}_1[\boldsymbol{\mu}^*(t) - \mathbf{1}_N \otimes \boldsymbol{\mu}_o(t)], & \text{if } \mathcal{T}_2^{\mathrm{T}}\boldsymbol{\varsigma}_2 = \mathbf{0}, \end{cases}$$

and $\lambda \in [-1, 1]$. Thus, using (10.31) and the fact that $\|s\|_1 = s^{\mathrm{T}}\mathrm{sign}(s)$, the derivative of W_2 is

$$\begin{aligned} \dot{W}_2(\boldsymbol{\varsigma}_2) &= -c_2\boldsymbol{\varsigma}_2^{\mathrm{T}}\mathbf{P}\mathcal{L}_2\boldsymbol{\varsigma}_2 + c_v\boldsymbol{\varsigma}_2^{\mathrm{T}}\mathbf{P}\mathcal{T}_1\tilde{\mathbf{V}}_c - \gamma\boldsymbol{\varsigma}_2^{\mathrm{T}}\mathbf{P}\mathcal{T}_1\mathrm{sign}\left(\mathcal{T}_2^{\mathrm{T}}\boldsymbol{\varsigma}_2\right) \\ &\quad + \boldsymbol{\varsigma}_2^{\mathrm{T}}\mathbf{P}\mathcal{T}_1\left[\boldsymbol{\mu}^*(t) - \mathbf{1}_N \otimes \boldsymbol{\mu}_o(t)\right] \\ &\leq -c_2'\left\|\boldsymbol{\varsigma}_2\right\|^2 + c_v''\left\|\boldsymbol{\varsigma}_2\right\|\left\|\tilde{\mathbf{V}}_c\right\|, \end{aligned} \tag{10.36}$$

where c_2' and c_v'' are positive constants.

Next, let us define $\boldsymbol{\varsigma}^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{\varsigma}_1^{\mathrm{T}} & \boldsymbol{\varsigma}_2^{\mathrm{T}} & \tilde{\mathbf{V}}_c^{\mathrm{T}} \end{bmatrix}$ and define the candidate Lyapunov function

$$W_{\varsigma}(\boldsymbol{\varsigma}) = W_1(\boldsymbol{\varsigma}_1) + \kappa_1 W_2(\boldsymbol{\varsigma}_2) + \frac{\kappa_2}{2} \left\| \tilde{\mathbf{V}}_c \right\|^2, \tag{10.37}$$

where κ_1 and κ_2 are positive constants. From (10.34), (10.36), and (10.29c), we have

$$\begin{aligned} \dot{W}_{\varsigma}(\boldsymbol{\varsigma}) \leq & -c_1' \left\| \bar{\boldsymbol{\varsigma}}_1 \right\|^2 - \kappa_1 c_2' \left\| \boldsymbol{\varsigma}_2 \right\|^2 - \kappa_2 c_v \left\| \tilde{\mathbf{V}}_c \right\|^2 + \left\| \bar{\boldsymbol{\varsigma}}_1 \right\| \left\| \boldsymbol{\varsigma}_2 \right\| \\ & + c_v' \left\| \boldsymbol{\varsigma}_1 \right\| \left\| \tilde{\mathbf{V}}_c \right\| + \kappa_1 c_v'' \left\| \boldsymbol{\varsigma}_2 \right\| \left\| \tilde{\mathbf{V}}_c \right\|. \end{aligned} \tag{10.38}$$

Setting κ_1, κ_2 large enough, we can find $\bar{c}_1, \bar{c}_2, \bar{c}_3, \bar{c} > 0$ such that

$$\dot{W}_{\varsigma}(\boldsymbol{\varsigma}) \leq -\bar{c}_1 \left\| \bar{\boldsymbol{\varsigma}}_1 \right\|^2 - \bar{c}_2 \left\| \boldsymbol{\varsigma}_2 \right\|^2 - \bar{c}_v \left\| \tilde{\mathbf{V}}_c \right\|^2 \leq -\bar{c}\, W_{\varsigma}(\boldsymbol{\varsigma}). \tag{10.39}$$

Now, let $\mathcal{W}_{\varsigma}$ be the set of the equilibria of the closed-loop system (10.29). Recalling Remark 10.1, $\mathcal{W}_{\varsigma}$ is given by

$$\mathcal{W}_{\varsigma} = \{\mathbf{0}\} \times \mathcal{W} \times \{\mathbf{0}\} \times \{\mathbf{0}\}^{2(N-1)} \times \{\mathbf{0}\}^{2N} \tag{10.40}$$

where $\mathcal{W}$ is defined in (10.19). Then, from (10.20) we have

$$\dot{W}_{\varsigma}(\boldsymbol{\varsigma}) \leq -\,\bar{c}' \left\| \boldsymbol{\varsigma} \right\|_{\mathcal{W}_{\varsigma}}^2. \tag{10.41}$$

Thus, the closed-loop system (10.29) is uniformly asymptotically multi-stable at $\mathcal{W}_{\varsigma}$, *c.f.,* [131]. Furthermore, the critical point $\tilde{\mathbf{z}}_1^*$ of the barrier Lyapunov function is a saddle point. After [132, Proposition 1], it follows that the region of attraction of the unstable equilibrium $\tilde{\mathbf{z}}_1^*$ has zero Lebesgue measure. Therefore, we conclude that the origin of (10.29) is *almost-everywhere* uniformly asymptotically stable in $\mathbb{D} = \mathbb{R}^3 \times \tilde{\mathcal{D}} \times \mathbb{R}^{3M} \times \mathbb{R}^{3N}$, except for a zero-measure set of initial conditions.

In order to establish forward invariance of the set $\tilde{\mathcal{D}}$ we proceed by contradiction. Assume that there exists $T > 0$ such that $\tilde{\mathbf{z}}_{1,k}(t) \in \tilde{\mathcal{D}}_k$ for all $t \in [t_0, t_0 + T)$, but $\tilde{\mathbf{z}}_{1,k}(t_0 + T) \notin \tilde{\mathcal{D}}_k$ for at least one $k \leq M$. In other words, we have $\left\| \mathbf{z}_{1,k}(t) \right\| \to \Delta_k$ or $\left\| \mathbf{z}_{1,k}(t) \right\| \to \delta_k$ as $t \to t_0 + T$ for at least one $k \leq M$. From the definition of $\tilde{\mathbf{z}}_{1,t} \mapsto W(\tilde{\mathbf{z}}_{1,t})$ in (10.18) and $\tilde{\mathbf{z}}_{1,k} \mapsto W_k(\tilde{\mathbf{z}}_{1,k})$ in (10.17), this implies that $W_{\varsigma}(\boldsymbol{\varsigma}(t)) \to \infty$ as $t \to t_0 + T$ which is in contradiction with (10.39). We can therefore conclude that $W_{\varsigma}(\boldsymbol{\varsigma}(t))$ is bounded for all initial conditions such that $\tilde{\mathbf{z}}_1(t_0) \in \tilde{\mathcal{D}}$, therefore, $W_{\varsigma}(\boldsymbol{\varsigma}(t)) \leq W_{\varsigma}(\boldsymbol{\varsigma}(t_0)) < \infty$ for all $\boldsymbol{\varsigma}(t_0) \in \mathbb{D}$ and all $t \geq t_0$. The respect of the inter-agent constraints follows from the forward invariance of $\tilde{\mathcal{D}}$.

Since (10.29) is asymptotically stable at the origin with domain of attraction $\mathbb{D}$ it follows that for almost all initial conditions $\boldsymbol{\varsigma}(t_0) \in \mathbb{D}$, there exist small positive constants $\underline{\epsilon}(\boldsymbol{\varsigma}(t_0))$ and $\bar{\epsilon}(\boldsymbol{\varsigma}(t_0))$ such that $\tilde{\mathbf{z}}_{1,k}(t) \in \tilde{\mathcal{D}}_{\epsilon k}$, where

$$\tilde{\mathcal{D}}_{\epsilon k} = \left\{ \tilde{\mathbf{z}}_{1,k} \in \mathbb{R}^3 \,|\, \delta_k + \underline{\epsilon} \leq \left\| \tilde{\mathbf{z}}_{1,k} + \mathbf{z}_{1,k}^d \right\| \leq \Delta_k - \bar{\epsilon} \right\}, \ \forall k \leq M. \tag{10.42}$$

Moreover, for any $\tilde{\mathbf{z}}_1 \in \tilde{\mathcal{D}}_{\epsilon}$, with $\tilde{\mathcal{D}}_{\epsilon} = \bigcap_{k \leq M} \tilde{\mathcal{D}}_{\epsilon k}$, we have that the BLF W in (10.18) satisfies

$$\frac{a_1}{2} \left\| \tilde{\mathbf{z}}_{1.t} \right\|_{\mathcal{W}}^2 \leq W(\tilde{\mathbf{z}}_{1,t}) \leq \frac{a_2'}{2} \left\| \tilde{\mathbf{z}}_{1,t} \right\|_{\mathcal{W}}^2. \tag{10.43}$$

Therefore, from (10.43) and (10.39), we conclude that for almost all initial conditions $\boldsymbol{\varsigma}(t_0) \in \mathbb{D}$, the trajectories $\boldsymbol{\varsigma}(t)$ of the external dynamics converge to the origin exponentially.

The ultimate boundedness of the internal dynamics can then be proven using Lemma 7.2. We showed that for almost all initial conditions $\boldsymbol{\varsigma}(t_0) \in \mathbb{D}$, there exist small positive constants $\underline{\epsilon}(\boldsymbol{\varsigma}(t_0))$ and $\overline{\epsilon}(\boldsymbol{\varsigma}(t_0))$ such that $\tilde{\mathbf{z}}_{1,k}(t) \in \tilde{\mathcal{D}}_{\epsilon k}$, with $\tilde{\mathcal{D}}_{\epsilon k}$ defined in (10.42). Consequently, for almost all initial conditions $\boldsymbol{\varsigma}(t_0) \in \mathbb{D}$, the input $\boldsymbol{\mu}$ defined in (10.27) is bounded. Therefore, if $a_x, \bar{\alpha}_y, \bar{\alpha}_z > 0$, then all assumptions of Lemma 7.2 are satisfied, and the internal dynamics are ultimately bounded. □

10.3 Control Design for Tracking with Hard and Soft Constraints

In this section, we build on the results of Section 10.2 to include the soft constraints defined in (10.5) that act on the surge velocity of the marine vehicles. For this purpose we explicitly design the additional control input $f_{u,i}^*$ introduced in Eq. (10.28) and we analyze the closed-loop system in terms of the barrier function framework.

Consider the surge velocity subsystem (2.22a) with an additional control input. That is,

$$\dot{u}_{r,i} = F_u(\boldsymbol{\nu}_{r,i}) + f_{u,i} + f_{u,i}^*. \tag{10.44}$$

In order to satisfy the soft constraints, we design the additional input $f_{u,i}^*$ as the gradient of a barrier function. For each agent i, define the barrier function

$$U_i(t, u_{r,i}) := -\ln\left(\frac{u_{r,i} + \rho_i(t)}{u_{r,i} + \rho_i(t) + 1}\right). \tag{10.45}$$

Note that if $u_{r,i} + \rho_i(t) > 0$, then $U_i(t, u_{r,i}) > 0$ for all $t \geq 0$, and $U_i(t, u_{r,i}) \to \infty$ as $u_{r,i} + \rho_i(t) \to 0$. Then, we set the additional control input to

$$f_{u,i}^* = -\kappa_u \nabla U_i(t, u_{r,i}) = -\kappa_u \frac{\partial U_i(t, u_{r,i})}{\partial u_{r,i}}, \quad i \leq N, \tag{10.46}$$

with $\kappa_u > 0$ and

$$\dot{\rho}_i = -\kappa_\rho \rho_i + \frac{1}{2}\left[1 - \operatorname{sign}\left(\sigma - |F_u(\boldsymbol{\nu}_{r,i}) + f_{u,i}|\right)\right] |F_u(\boldsymbol{\nu}_{r,i}) + f_{u,i}|, \tag{10.47}$$

where $\kappa_\rho,\ \sigma > 0$ are design constants. Initially, we set $\rho_i(t_0) = 0$.

Remark 10.2. *Note that under* (10.47) *and the initial condition* $\rho_i(t_0) = 0$*, we have that* $\rho_i(t) \geq 0$*, for all* $t \geq t_0$*. To see this, note that second term on the right-hand side of* (10.47) *is always positive. Therefore,* $\dot{\rho}_i(t) \geq -\kappa_\rho \rho_i(t)$*, which means that the set* $\mathcal{C}_\rho = \{\rho_i \in \mathbb{R} \,|\, \rho_i \geq 0\}$ *is forward invariant.*

Remark 10.3. *The definition of* (10.47) *is loosely inspired by the framework developed in [134] to deal with hard and soft constraints in the setting of prescribed-performance control of single-agent systems. The signal* $\rho_i(t)$ *adjusts the soft constraints whenever the hard constraints become conflicting. Note that when the term*

$|F_u + f_{u,i}|$ is less than or equal to a given positive constant σ, it means that the edges connected to vehicle i are far from the border of the set $\tilde{\mathcal{D}}_k$, since under the barrier-function-based law (10.27) *the controller $f_{u,i}$ grows unbounded as $\mathbf{z}_{1,k} \to \partial\tilde{\mathcal{D}}_k$ for any $k \leq M$. In this case, the second term on the right-hand side of* (10.47) *is equal to zero. Hence, assuming that in an interval $t \in [t_0, t_0 + T]$, $|F_u + f_{u,i}| \leq \sigma$, then* (10.5)*, with $\rho(t) = 0$, corresponds to a positive-velocity constraint. Conversely, when $|F_u + f_{u,i}| > \sigma$, the right-hand side of* (10.47) *becomes positive and ρ_i grows. Hence, $u_{r,i}$ may take negative values,* i.e., *$u_{r,i} > -\rho_i(t)$, avoiding possible conflicts between the constraints. Then, as the vehicles move away from the border of the set $\tilde{\mathcal{D}}_k$, $|F_u + f_{u,i}| \leq \sigma$ again and $\rho_i(t) \to 0$ exponentially fast, recovering the non-negativity constraint.*

Then, the second part of our main result is stated as follows:

Proposition 10.2. *Consider N AUVs, each described by the model* (10.1)*, and interconnected over a directed graph which has either a spanning tree or a cycle. Then, under the same assumptions as in Proposition 10.1, and with initial conditions such that $\mathbf{z}_{1,k}(t_0) \in \mathcal{D}_k$ for all $k \leq M$ and $u_{r,i}(t_0) \in \mathcal{C}_i$ for all $i \leq N$, the controller* (10.27)*, with $f^*_{u,i}$ given by* (10.46)*, achieves the tracking-in-formation objective* (10.12) *almost everywhere and renders the constraints sets* (10.3) *and* (10.6) *forward invariant. Moreover, the internal dynamics are ultimately bounded.*

Proof. In Proposition 10.1 we established that the controller (10.27) with the additional bounded input $f^*_{u,i}$ renders the hard-constraints set (10.3) forward invariant and guarantees the achievement of the tracking-in-formation objective (10.12) for almost all initial conditions such that $\mathbf{z}_{1,k}(t_0) \in \mathcal{D}_k$, for all $k \leq M$. Therefore, to prove Proposition 10.2, what remains is to show that $f^*_{u,i}$ given by (10.46) is bounded and guarantees the satisfaction of the soft constraints (10.4). For that purpose, consider the barrier function (10.45), whose derivative along the trajectories of (10.44) in $\mathcal{C}_i$ yields

$$\dot{U}_i(t, u_{r,i}) = \nabla U_i(t, u_{r,i}) \left(-\kappa_u \nabla U_i(t, u_{r,i}) + F_u(\boldsymbol{\nu}_{r,i}) + f_{u,i} + \dot{\rho}_i(t)\right). \tag{10.48}$$

Now, in view of (10.47), we split the analysis into two cases.
Case 1 ($|F_u + f_{u,i}| \leq \sigma$): in this case (10.48) becomes

$$\begin{aligned}\dot{U}_i(t, u_{r,i}) \leq& -\kappa_u \left|\nabla U_i(t, u_{r,i})\right|^2 + \left|\nabla U_i(t, u_{r,i})\right| \left[\sigma - \kappa_\rho \rho_i(t)\right] \\ \leq& -\kappa'_u \left|\nabla U_i(t, u_{ri})\right|^2 + \lambda_\sigma \sigma^2 + \frac{\kappa_\rho \rho_i(t)}{(u_{r,i} + \rho_i(t))(u_{r,i} + \rho_i(t) + 1)},\end{aligned} \tag{10.49}$$

with $\kappa'_u,\ \lambda_\sigma > 0$. Since $\rho_i(t)$ is non-negative in $\mathcal{C}_i$ for all $t \geq t_0$, *c.f.,* Remark 10.2, the last term on the right-hand side of (10.49) is bounded by a constant $\lambda_\rho > 0$. Therefore, we have

$$\dot{U}_i(t, u_{r,i}) \leq -\kappa_u \left|\nabla U_i(t, u_{r,i})\right|^2 + \lambda_\sigma \sigma^2 + \lambda_\rho. \tag{10.50}$$

Case 2 ($|F_u + f_{u,i}| > \sigma$): for $u_{r,i} \in \mathcal{C}_i$, (10.48) becomes

$$\begin{aligned}\dot{U}_i(t, u_{r,i}) \leq& \left|\nabla U_i(t, u_{r,i})\right| \left[|F_u + f_{u,i}| - \kappa_\rho \rho_i(t) - |F_u + f_{u,i}|\right] \\ & - \kappa_u \left|\nabla U_i(t, u_{r,i})\right|^2 \\ \leq& -\kappa_u \left|\nabla U_i(t, u_{r,i})\right|^2 + \lambda_\rho.\end{aligned} \tag{10.51}$$

From (10.50)-(10.51) we conclude that for $u_{r,i} \in \mathcal{C}_i$, the function $U_i(t, u_{r,i})$ is bounded along the trajectories. Therefore, akin to the forward invariance of $\tilde{\mathcal{D}}$ established in the proof of Proposition 10.1, it is straightforward to show forward invariance of $\mathcal{C}_i$. Hence, there exists a constant $\bar{f}^*$ such that $|f_{u,i}(t)| \leq \bar{f}^*$ for all $t \geq t_0$.

Note that the additional input $f^*_{u,i}$ does not affect the internal dynamics. Hence, the analysis of the internal dynamics presented in the proof of Proposition 10.1 still holds under the action of $f^*_{u,i}$. □

10.4 Simulations

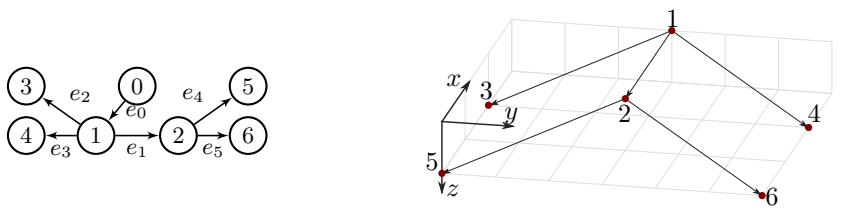
In this section we illustrate the performance of the controller (10.27) through simulations in MATLAB and DUNE [89]. The MATLAB simulation enables us to validate the closed-loop behavior under ideal conditions. The Unified Navigation Environment (DUNE) is a software platform designed to run on autonomous underwater vehicles. It also contains a high-fidelity AUV simulator, allowing us to validate the proposed control algorithm, reproducing as closely as possible a laboratory experiment.

The simulation case consists in the tracking-in-formation problem for six AUVs subject to hard (proximity and collision-avoidance) and soft (positive surge velocity) constraints. We assume that the vehicles are interconnected at the initial time and that the controller must preserve this connectivity. We further assume that the AUVs interact over a directed spanning tree illustrated in Figure 10.1a and that only AUV 1 has knowledge of the (relative) state of the target (labeled 0).

10.4.1 MATLAB Example

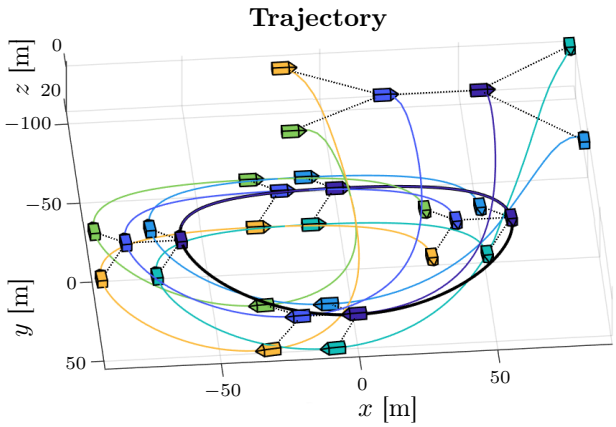
Here we simulate six light autonomous underwater vehicles (LAUVs). The ocean current velocity is set to $\mathbf{V}_c = [0.05 \quad -0.08 \quad -0.03]^{\mathrm{T}}$. The desired formation is illustrated in Figure 10.1b. The desired relative positions $\mathbf{z}^d_{1,k}$ are given by

$$\begin{bmatrix} \mathbf{z}^d_{1,1} & \cdots & \mathbf{z}^d_{1,5} \end{bmatrix} = \begin{bmatrix} 20 & 10 & 10 & 10 & 10 \\ 0 & 15 & -15 & 15 & -15 \\ 0 & -5 & -5 & -5 & -5 \end{bmatrix} \tag{10.52}$$

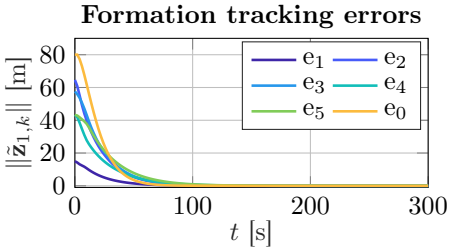


(a) Interaction topology. **(b)** Desired formation shape.

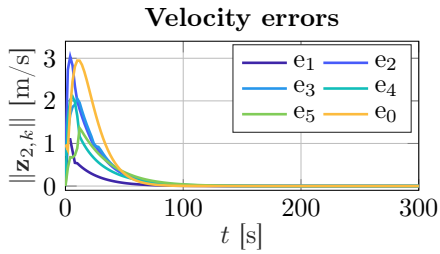
Figure 10.1: Illustrations of the simulated example.



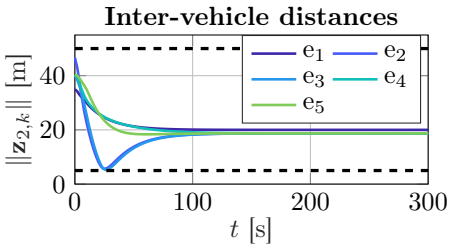
(a) The 3D trajectory of the AUVs. The black line represents the virtual target.



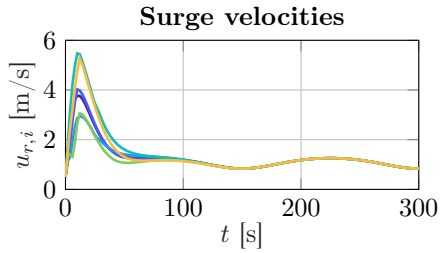
(b) Norms of the formation-keeping errors, $\tilde{\mathbf{z}}_{1,k}$, and the tracking error, $\tilde{\mathbf{z}}_{1,o}$.



(c) Norms of the velocity errors $\mathbf{z}_{2,k}$ and $\mathbf{z}_{2,o}$.



(d) Inter-vehicle distances $\|\mathbf{z}_{1,k}\|$. The dashed lines represent the limits δ_k and Δ_k.



(e) Surge velocities of the AUVs.

Figure 10.2: Results of numerical simulations in MATLAB.

The trajectory of the virtual target is

$$\mathbf{x}_{1,o} = \left[a\cos(\omega_o t), b\sin(\omega_o t), c\sin(\omega_o t)^2\right]^{\mathrm{T}}, \tag{10.53}$$

where $a = 60$, $b = 40$, $c = 10$, and $\omega_o = \frac{\pi}{150}$.

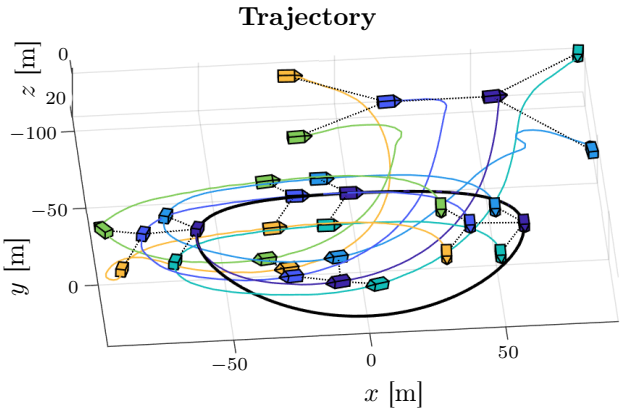
The maximal and minimal distance parameters are $\Delta_k = 50\,\mathrm{m}$ and $\delta_k = 5\,\mathrm{m}$. The hand position length is chosen as $h = 5\,\mathrm{m}$, and the control gains are set to $c_1 = 0.1$, $c_2 = 0.5$, $\gamma = 0.25$, $c_v = 0.2$, $\kappa_u = 0.1$, $\kappa_\rho = 4$, and $\sigma = 0.3$. Furthermore, in order to avoid discontinuities in the control, the non-smooth sign(s) function in (10.27) and (10.47) is replaced by the smooth approximation $\tanh(c_a\, s)$, with $c_a = 10^3$.

Figure 10.2 presents the results of the simulation scenario. Specifically, Figure 10.2a shows the 3D trajectories of the AUVs. We can see that the vehicles successfully reach the desired formation while following the target, as is also evidenced from the formation and tracking errors in Figure 10.2b and the velocity errors in Figure 10.2c. Furthermore, note that the hard connectivity and collision-avoidance constraints, shown as dashed black lines in Figure 10.2d, are always respected. The soft constraints are satisfied as well, and the surge velocities are kept non-negative, as shown in Figure 10.2e.

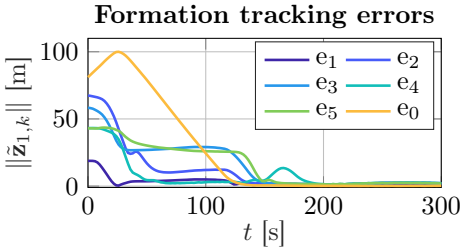
10.4.2 DUNE simulations

Here, we simulate a formation of six LAUVs using DUNE. The parameters of the simulation are chosen identically to the MATLAB example.

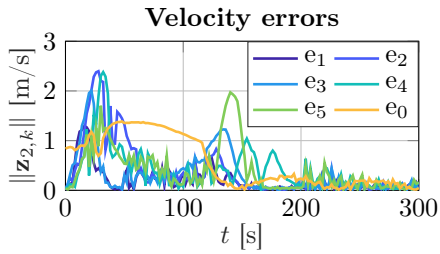
Figure 10.3 presents the results of the simulation. Specifically, Figure 10.3a shows the 3D trajectories of the AUVs. We can see that the vehicles manage to reach the desired formation. However, the transient behavior is different from the one under the ideal conditions of the MATLAB simulation. One reason behind these differences is that in the DUNE simulation, the torque produced by the fins depends on the speed of the vehicle. Consequently, the AUVs cannot turn if their speed is too low. In addition, the surge thrust of the AUVs is limited. Consequently, the AUVs can only reach a surge velocity of $1.8\,\mathrm{m\,s^{-1}}$, as shown in Figure 10.3e. Unlike the MATLAB simulations, the soft constraints cannot always be satisfied, and the surge velocities of AUVs 2 and 5 briefly drop below zero. Having a negative surge velocity is necessary to satisfy the hard connectivity and collision-avoidance constraints shown as dashed black lines in Figure 10.3d. The formation-tracking errors and the velocity errors are shown in Figures 10.3b and 10.3c, respectively. We can see that these errros do not converge to zero but rather to a small area around zero. These nonzero steady-state errors are caused by two factors: the uncertainty of the navigation system, and the delay in the actuators.



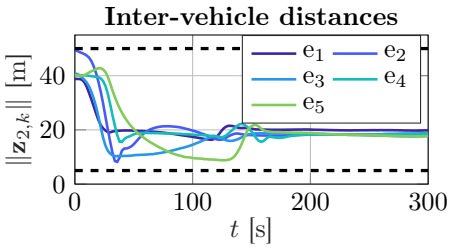
(a) The 3D trajectory of the AUVs. The black line represents the virtual target.



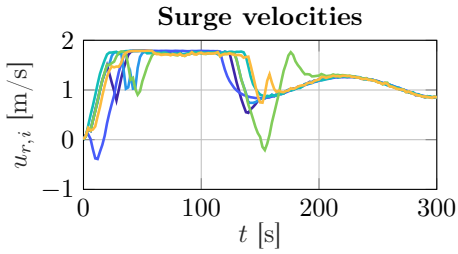
(b) Norms of the formation-keeping errors, $\tilde{\mathbf{z}}_{1,k}$, and the tracking error, $\tilde{\mathbf{z}}_{1,o}$.



(c) Norms of the velocity errors $\mathbf{z}_{2,k}$ and $\mathbf{z}_{2,o}$.



(d) Inter-vehicle distances $\|\mathbf{z}_{1,k}\|$. The dashed lines represent the limits δ_k and Δ_k.



(e) Surge velocities of the AUVs.

Figure 10.3: Results of numerical simulations in DUNE.

Chapter 11

Combining NSB with the Hand Position Approach

This chapter presents an extended null-space-behavioral (NSB) algorithm for the formation control of fleets of underactuated autonomous underwater vehicles. The NSB controller is developed to work directly with second-order integrator systems, handling the double integrator dynamics in task space. The method is applied to the formation path-following problem of a fleet of underactuated autonomous underwater vehicles. The nonlinear six-degrees-of-freedom models of the vehicles are transformed into second-order integrator systems using the 3D hand position output linearizing approach presented in Chapter 7. The behavioral controller implements a hierarchy of path-following, formation-keeping, and collision-avoidance tasks. The closed-loop system is proven uniformly globally asymptotically stable, and the proposed method is validated through numerical simulations. The contents of this chapter are based on [85].

This chapter is organized as follows. Section 11.1 presents the extended NSB algorithm which is applicable for general double-integrator systems. Section 11.2 presents a case study of this NSB algorithm applied to a fleet of AUVs.

11.1 The NSB Algorithm for Double Integrators

The NSB method enables the creation of multiple tasks in a hierarchical manner, ensuring that low-priority tasks do not interfere with high-priority ones. In this section, we extend this method to second-order systems with double integrator dynamics. This modified NSB algorithm provides the acceleration input $\boldsymbol{\mu}$ to the following system

$$\dot{\mathbf{p}} = \mathbf{v}, \tag{11.1a}$$

$$\dot{\mathbf{v}} = \boldsymbol{\mu}. \tag{11.1b}$$

For each task, we design a task variable $\boldsymbol{\sigma}_m \in \mathbb{R}^{n_m}$ as a function of $\mathbf{p}$

$$\boldsymbol{\sigma}_m = \mathbf{f}_m(\mathbf{p}). \tag{11.2}$$

The first and second time-derivatives of $\boldsymbol{\sigma}_i$ are

$$\dot{\boldsymbol{\sigma}}_m = \mathbf{J}_m(\mathbf{p})\mathbf{v}, \tag{11.3a}$$

$$\ddot{\boldsymbol{\sigma}}_m = \mathbf{J}_m(\mathbf{p})\dot{\mathbf{v}} + \dot{\mathbf{J}}_m\mathbf{v}, \tag{11.3b}$$

where $\mathbf{J}_m = \partial\mathbf{f}_m/\partial\mathbf{p}$ is the task Jacobian. We denote the desired value of the task variable by $\boldsymbol{\sigma}_{d,m}$.

In Section 2.5, we introduced the closed-loop inverse kinematics (CLIK) equation as a method for finding the desired velocity associtated with a given task. Recalling (2.77), the desired velocity of task m, $\mathbf{v}_m$, is given by

$$\mathbf{v}_m = \mathbf{J}_m^{\dagger}\big(\dot{\boldsymbol{\sigma}}_{d,m} - \boldsymbol{\Lambda}_m\tilde{\boldsymbol{\sigma}}_m\big), \tag{11.4}$$

where $\boldsymbol{\Lambda}_m$ is a positive definite gain matrix, $\tilde{\boldsymbol{\sigma}}_m = \boldsymbol{\sigma}_m - \boldsymbol{\sigma}_{d,i}$ is the task error, and $\mathbf{J}_m^{\dagger}$ is the pseudo-inverse of the task Jacobian. To achieve second-order differential control, we instead propose the second-order closed-loop inverse kinematics (SOCLIK) equation inspired by robotic manipulators [135]

$$\dot{\mathbf{v}}_m = \mathbf{J}_m^{\dagger}\big(\ddot{\boldsymbol{\sigma}}_{d,m} - \boldsymbol{\Lambda}_{p,m}\tilde{\boldsymbol{\sigma}}_m - \boldsymbol{\Lambda}_{d,m}\dot{\tilde{\boldsymbol{\sigma}}}_m - \dot{\mathbf{J}}_m\mathbf{v}\big), \tag{11.5}$$

where $\boldsymbol{\Lambda}_{p,m}$ and $\boldsymbol{\Lambda}_{d,m}$ are positive definite gain matrices.

In first-order systems, there exists a subspace of velocities that do not conflict with a given task. Similarly, in second-order systems, there exists a subspace of non-conflicting accelerations. Let $\dot{\mathbf{v}}_m$ be the SOCLIK solution to task m, and let $\dot{\mathbf{v}}_{\text{add}}$ be some additional desired acceleration. Then, the following control input

$$\dot{\mathbf{v}} = \dot{\mathbf{v}}_m + \mathbf{N}_m\dot{\mathbf{v}}_{\text{add}}, \tag{11.6}$$

where $\mathbf{N}_m$ is the null space projector of the task, guarantees the desired behavior of the task.

Similarly to first-order NSB methods, we can combine the tasks by projecting the inputs from the lower-priority tasks onto the null space of the higher-priority tasks. The desired acceleration is then given by

$$\boldsymbol{\mu} = \dot{\mathbf{v}}_1 + \sum_{m=2}^{M}\bar{\mathbf{N}}_{m-1}\dot{\mathbf{v}}_m, \tag{11.7}$$

where M is the number of tasks and $\bar{\mathbf{N}}_m$ is the null space projector of the augmented Jacobian (2.80). With this choice of acceleration, the highest-priority task is always fulfilled, whereas the lower-priority tasks are fulfilled as well as possible in the subspace that does not conflict with higher-priority tasks.

11.1.1 Stability Analysis

In this section, we investigate the stability of an NSB algorithm consisting of two tasks. The proof is based on [103], but extended to a double integrator system. The proof utilizes the concepts of independence and orthogonality defined in Section 2.5.1.

Lemma 11.1. *Consider two independent and orthogonal tasks labeled 1 and 2. Let* $\tilde{\mathbf{X}}^{\mathrm{T}} = \left[\tilde{\boldsymbol{\sigma}}_1^{\mathrm{T}}, \tilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}, \dot{\tilde{\boldsymbol{\sigma}}}_1^{\mathrm{T}}, \dot{\tilde{\boldsymbol{\sigma}}}_2^{\mathrm{T}}\right]$ *be the stacked error vector (*$\tilde{\boldsymbol{\sigma}}_m = \boldsymbol{\sigma}_m - \boldsymbol{\sigma}_{d,m}$*).*

The control input defined in (11.7) *ensures that* $\tilde{\mathbf{X}} = \mathbf{0}$ *is a globally exponentially stable (GES) equilibrium point.*

Proof. First, let us find the closed-loop expressions for $\ddot{\tilde{\boldsymbol{\sigma}}}_1$ and $\ddot{\tilde{\boldsymbol{\sigma}}}_2$. From (11.3b) and (11.7), we get

$$\ddot{\tilde{\boldsymbol{\sigma}}}_1 = \mathbf{J}_1\dot{\boldsymbol{\nu}}_1 + \mathbf{J}_1\mathbf{N}_1\dot{\boldsymbol{\nu}}_2 + \dot{\mathbf{J}}_1\boldsymbol{\nu} - \ddot{\boldsymbol{\sigma}}_{d,1}, \tag{11.8a}$$

$$\ddot{\tilde{\boldsymbol{\sigma}}}_2 = \mathbf{J}_2\dot{\boldsymbol{\nu}}_1 + \mathbf{J}_2\mathbf{N}_1\dot{\boldsymbol{\nu}}_2 + \dot{\mathbf{J}}_2\boldsymbol{\nu} - \ddot{\boldsymbol{\sigma}}_{d,2}, \tag{11.8b}$$

Note that thanks to the independence and orthogonality assumptions, it follows that $\mathbf{J}_2\mathbf{N}_1\mathbf{J}_2^{\dagger} = \mathbf{I}$. Consequently, substituting (11.5) into (11.8) and using (11.3), the time-derivative of $\tilde{\mathbf{X}}$ is given by

$$\dot{\tilde{\mathbf{X}}} = \mathbf{M}\tilde{\mathbf{X}}, \qquad \mathbf{M} = \begin{bmatrix} \mathbf{O} & \mathbf{O} & \mathbf{I} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{I} \\ -\boldsymbol{\Lambda}_{p,1} & \mathbf{O} & -\boldsymbol{\Lambda}_{d,1} & \mathbf{O} \\ \mathbf{O} & -\boldsymbol{\Lambda}_{p,2} & \mathbf{O} & -\boldsymbol{\Lambda}_{d,2} \end{bmatrix}. \tag{11.9}$$

Since the gain matrices are positive definite by design, the matrix $\mathbf{M}$ is Hurwitz, and the closed-loop system is GES. □

11.2 Case Study: Formation Path Following of AUVs

The following sections present a case study of the proposed second-order NSB algorithm applied to a fleet of underactuated AUVs equipped with the hand-position-based controller defined in Chapter 7. The control objective of the fleet is to follow a predefined path while keeping formation and avoiding obstacles.

The vehicle model under the hand position controller is presented in Section 11.2.1 and the formation path following problem is formulated in Section 11.2.2. The NSB tasks are detailed in Section 11.2.3, Section 11.2.3 details the obstacle avoidance method, Section 11.2.4 analyzes the stability properites, and Section 11.2.5 presents a simulation study.

11.2.1 AUV Model

We consider a 6DOF model of an AUV exposed to an unknown constant irrotational ocean current, and apply the 3D hand position transform from Chapter 7. Recalling (7.11), the transformed dynamics of the AUV are

$$\dot{\mathbf{x}}_1 = \mathbf{x}_2 + \mathbf{V}_c, \tag{11.10a}$$

$$\dot{\mathbf{x}}_2 = \boldsymbol{\mu}, \tag{11.10b}$$

$$\dot{\mathbf{R}} = \mathbf{R}\mathbf{S}(\boldsymbol{\omega}), \tag{11.10c}$$

$$\begin{aligned} \dot{\boldsymbol{\omega}} = {} & \bar{\boldsymbol{\ell}} \times \left(\mathbf{R}^{\mathrm{T}}\boldsymbol{\mu} + \mathcal{D}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}) + \mathcal{C}_{\boldsymbol{\upsilon}}(\boldsymbol{\nu}) - \boldsymbol{\omega} \times \mathbf{R}^{\mathrm{T}}\mathbf{x}_2\right) \\ & - \left(\bar{\boldsymbol{\ell}}\boldsymbol{\ell}^{\mathrm{T}}\right)\left(\mathcal{D}_{\boldsymbol{\omega}}(\boldsymbol{\nu}) + \mathbf{M}'_{22}\left(W z_{gb}\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right)\right). \end{aligned} \tag{11.10d}$$

11.2.2 Formation Path Following

The formation path-following problem considered in this chapter is analogous to the one defined in Section 2.3. We consider a fleet of N AUVs and define the stacked position and velocity vectors $\mathbf{x}_1 = [\mathbf{x}_{1,1}^{\mathrm{T}}, \ldots, \mathbf{x}_{1,N}^{\mathrm{T}}]^{\mathrm{T}}$ and $\mathbf{x}_2 = [\mathbf{x}_{2,1}^{\mathrm{T}}, \ldots, \mathbf{x}_{2,N}^{\mathrm{T}}]^{\mathrm{T}}$, respectively. We also define the stacked ocean current vector $\mathcal{V}_c = \mathbf{1}_N \otimes \mathbf{V}_c$.

The desired path is parametrized by a smooth function $\mathbf{p}_p : \mathbb{R} \mapsto \mathbb{R}^3$ that is assumed to be $\mathcal{C}^\infty$ and regular. For every point $\mathbf{p}_p(s)$, there exists a path-tangential coordinate frame with a corresponding rotation matrix $\mathbf{R}_p$ (see Section 2.2). The path-following error $\mathbf{p}_b^p$ is defined in terms of the barycenter of the fleet in the path-tangential coordinate frame

$$\mathbf{p}_b^p = \mathbf{R}_p^{\mathrm{T}} \left(\mathbf{p}_b - \mathbf{p}_p(s) \right), \quad \mathbf{p}_b = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_{1,i}. \tag{11.11}$$

The goal of path following is to control the vehicles so that $\mathbf{p}_b^p \to \mathbf{0}_3$.

The vehicles should converge to a dynamic formation that rotates with the desired path (see Section 2.3.2 for details). Let $\mathbf{p}_{f,1}^f, \ldots, \mathbf{p}_{f,N}^f$ be the position vectors that represent the desired formation. The objective is to control the vehicles so that

$$\mathbf{x}_{1,i} - \mathbf{p}_b \to \mathbf{R}_p \mathbf{p}_{f,i}^f, \qquad \forall i \in \{1, \ldots, N\}. \tag{11.12}$$

11.2.3 NSB Controller

We let the control system consist of three tasks in decreasing order of priority: inter-vehicle collision avoidance, formation keeping, and path following. The following sections will detail the chosen task variables and SOCLIK solution for each task.

Inter-Vehicle Collision Avoidance

The highest-priority task is inter-vehicle collision avoidance (COLAV). The task variable is given by a stacked vector $\boldsymbol{\sigma}_1 = [\sigma_{1,1}^{\mathrm{T}}, \ldots, \sigma_{1,l}^{\mathrm{T}}]^{\mathrm{T}}$ of relative distances between vehicles closer than the activation distance d_{COLAV}:

$$\begin{aligned} \sigma_{1,k} = \|\mathbf{x}_{1,i} - \mathbf{x}_{1,j}\|, \quad & \forall i, j \in \{1, \ldots, N\} : j > i, \\ & \|\mathbf{x}_{1,i} - \mathbf{x}_{1,j}\| < d_{COLAV}. \end{aligned} \tag{11.13}$$

The task size varies depending on the number of vehicles that are within the activation distance, and it is empty under nominal conditions. The desired values of the task are given by

$$\boldsymbol{\sigma}_{d,1} = d_{COLAV} \mathbf{1}_l. \tag{11.14}$$

We note that $\ddot{\boldsymbol{\sigma}}_{d,1} = \dot{\boldsymbol{\sigma}}_{d,1} = \mathbf{0}$.

The task Jacobian is given by the stacked partial derivatives for each active

collision

$$\mathbf{J}_1 = \left[\left(\frac{\partial \sigma_{1,1}}{\partial \mathbf{x}_1} \right)^{\mathrm{T}}, \ldots, \left(\frac{\partial \sigma_{1,l}}{\partial \mathbf{x}_1} \right)^{\mathrm{T}} \right]^{\mathrm{T}}, \tag{11.15a}$$

$$\frac{\partial \sigma_{1,k}}{\partial \mathbf{x}_{1,i}} = \frac{(\mathbf{x}_{1,i} - \mathbf{x}_{1,j})^{\mathrm{T}}}{\|\mathbf{x}_{1,i} - \mathbf{x}_{1,j}\|}, \qquad \frac{\partial \sigma_{1,k}}{\partial \mathbf{x}_{1,j}} = -\frac{(\mathbf{x}_{1,i} - \mathbf{x}_{1,j})^{\mathrm{T}}}{\|\mathbf{x}_{1,i} - \mathbf{x}_{1,j}\|}. \tag{11.15b}$$

The derivative of the task Jacobian is given by

$$\dot{\mathbf{J}}_1 = \left[\left(\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial \sigma_{1,1}}{\partial \mathbf{x}_1} \right)^{\mathrm{T}}, \ldots, \left(\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial \sigma_{1,l}}{\partial \mathbf{x}_1} \right)^{\mathrm{T}} \right]^{\mathrm{T}}, \tag{11.16a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial \sigma_{1,k}}{\partial \mathbf{x}_{1,i}} = \left(\frac{\mathbf{I}_3}{\|\mathbf{x}_{1,i} - \mathbf{x}_{1,j}\|} - \frac{\big(\mathbf{x}_{1,i} - \mathbf{x}_{1,j}\big)\big(\mathbf{x}_{1,i} - \mathbf{x}_{1,j}\big)^{\mathrm{T}}}{\|\mathbf{x}_{1,i} - \mathbf{x}_{1,j}\|^3} \right) \big(\mathbf{x}_{2,i} - \mathbf{x}_{2,j}\big), \tag{11.16b}$$

The resulting SOCLIK equation for the task is

$$\dot{\mathbf{v}}_1 = -\mathbf{J}_1^{\dagger} \big(\mathbf{\Lambda}_{p,1} \tilde{\boldsymbol{\sigma}}_1 + \mathbf{\Lambda}_{d,1} \dot{\boldsymbol{\sigma}}_1 + \dot{\mathbf{J}}_1 (\mathbf{x}_2 + \mathcal{V}_c) \big), \tag{11.17}$$

with $\dot{\boldsymbol{\sigma}}_1 = \mathbf{J}_1(\mathbf{x}_2 + \mathcal{V}_c)$. Note that due to the structure of the task Jacobian, it follows that $\mathbf{J}_1 \mathcal{V}_c = \dot{\mathbf{J}}_1 \mathcal{V}_c = \mathbf{0}$. Consequently, $\dot{\mathbf{v}}_1$ is independent of the ocean current velocity.

Formation-Keeping Task

The formation-keeping task moves the vehicles into a predefined formation in the formation-centered frame. The task variable is given by

$$\boldsymbol{\sigma}_2^{\mathrm{T}} = \left[\boldsymbol{\sigma}_{2,1}^{\mathrm{T}}, ..., \boldsymbol{\sigma}_{2,N-1}^{\mathrm{T}} \right], \quad \boldsymbol{\sigma}_{2,i} = \mathbf{x}_{1,i} - \mathbf{p}_b. \tag{11.18}$$

The desired values are

$$\boldsymbol{\sigma}_{d,2}^{\mathrm{T}} = \left[\big(\mathbf{R}_p \mathbf{p}_{f,1}^f \big)^{\mathrm{T}}, ..., \big(\mathbf{R}_p \mathbf{p}_{f,N-1}^f \big)^{\mathrm{T}} \right]. \tag{11.19}$$

The Jacobian is given by

$$\mathbf{J}_2 = \left(\begin{bmatrix} \mathbf{I}_{N-1} & \mathbf{0}_{N-1} \end{bmatrix} - \frac{\mathbf{1}_{(N-1)\times N}}{N} \right) \otimes \mathbf{I}_3. \tag{11.20}$$

Because $\dot{\mathbf{J}}_2 = \mathbf{0}$, the SOCLIK equation reduces to

$$\dot{\mathbf{v}}_2 = \mathbf{J}_2^{\dagger} \big(\ddot{\boldsymbol{\sigma}}_{d,2} - \mathbf{\Lambda}_{p,2} \tilde{\boldsymbol{\sigma}}_2 - \mathbf{\Lambda}_{d,2} \dot{\tilde{\boldsymbol{\sigma}}}_2 \big). \tag{11.21}$$

The nominal task acceleration (11.21) may saturate the actuators if the formation error is large. The NSB controller may also lead to a loss of controllability if the formation-keeping velocities exactly cancel out the path-following velocities. Therefore, similarly to Chapter 5, we introduce saturated task acceleration

$$\dot{\mathbf{v}}_2 = \mathbf{J}_2^{\dagger} \big(\ddot{\boldsymbol{\sigma}}_{d,2} - v_{2_{\max}} \mathrm{sat}\big(\mathbf{\Lambda}_{p,2} \tilde{\boldsymbol{\sigma}}_2 \big) - \mathbf{\Lambda}_{d,2} \dot{\tilde{\boldsymbol{\sigma}}}_2 \big), \tag{11.22}$$

where $v_{2_{\max}}$ is a positive constant and sat is a saturation function given by

$$\mathrm{sat}\big(\mathbf{x}\big) = \mathbf{x}\frac{\tanh\|\mathbf{x}\|}{\|\mathbf{x}\|}. \tag{11.23}$$

With the saturated task acceleration, we further require that the product of the gain matrices $\mathbf{\Lambda}_{p,2}\mathbf{\Lambda}_{d,2}$ is symmetric positive definite. Similarly to the inter-vehicle collision avoidance task, this task is independent of the ocean current.

Path-Following Task

The path following task concerns moving the barycenter along the predefined path. Moreover, we want the formation to move at a desired constant path-following speed U_{LOS}. We apply the same acceleration to all vehicles to ensure that the barycenter moves without changing the relative formation.

We apply the coupled line-of-sight (LOS) guidance algorithm defined in Section 2.4, and modify it to work with double-integrator systems. We denote the components of the path following error $\mathbf{p}_b^p$ as x_b^p, y_b^p, and z_b^p. Similarly to Chapter 5, we let $\Delta(\mathbf{p}_b^p)$ be the error-dependent look-ahead distance of the LOS guidance law given by

$$\Delta(\mathbf{p}_b^p) = \sqrt{\Delta_0^2 + (x_b^p)^2 + (y_b^p)^2 + (z_b^p)^2}, \tag{11.24}$$

where Δ_0 is a positive constant. The LOS velocity is then given by

$$\mathbf{v}_{\mathrm{LOS},d} = \mathbf{R}_p\left[\Delta(\mathbf{p}_b^p), -y_b^p, -z_b^p\right]^{\mathrm{T}}\frac{U_{\mathrm{LOS}}}{D}, \tag{11.25}$$

where $D = \sqrt{\Delta(\cdot)^2 + (y_b^p)^2 + (z_b^p)^2}$ is a normalization term.

Since the second-order NSB algorithm requires the desired acceleration, we need to find the time-derivative of the line-of-sight velocity (11.25)

$$\begin{aligned}\dot{\mathbf{v}}_{\mathrm{LOS},d} = {}& \dot{\mathbf{R}}_p\left[\Delta(\mathbf{p}_b^p), -y_b^p, -z_b^p\right]^{\mathrm{T}}\frac{U_{\mathrm{LOS}}}{D}\\ &+ \mathbf{R}_p\left[\dot{\Delta}(\mathbf{p}_b^p, \dot{\mathbf{p}}_b^p), -\dot{y}_b^p, -\dot{z}_b^p\right]^{\mathrm{T}}\frac{U_{\mathrm{LOS}}}{D}\\ &- \mathbf{R}_p\left[\Delta(\mathbf{p}_b^p), -y_b^p, -z_b^p\right]^{\mathrm{T}}\frac{U_{\mathrm{LOS}}}{D^2}\dot{D}.\end{aligned} \tag{11.26}$$

We want to eliminate the error caused by the constant unknown ocean current at this stage of the control hierarchy, as all higher-priority tasks are independent of it. To this end, we introduce the virtual integral state $\mathbf{p}_v$ defined by

$$\dot{\mathbf{p}}_v = \mathbf{v}_{\mathrm{LOS},d}, \tag{11.27}$$

and define the following task acceleration

$$\dot{\mathbf{v}}_{\mathrm{LOS}} = \dot{\mathbf{v}}_{\mathrm{LOS},d} + \mathbf{\Lambda}_{p,3}(\mathbf{v}_{\mathrm{LOS},d} - \mathbf{v}_b) + \mathbf{\Lambda}_{i,3}(\mathbf{p}_v - \mathbf{p}_b), \tag{11.28}$$

where $\mathbf{v}_b = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_{2,i}$.

Lemma 11.2. *Let $\mathbf{\Lambda}_{p,3}$ and $\mathbf{\Lambda}_{i,3}$ be two symmetric positive definite matrices. The relative barycenter velocity $\mathbf{v}_b$ converges exponentially to the relative LOS velocity $\mathbf{v}_{\mathrm{LOS},d} - \mathbf{V}_c$ under controller* (11.7)*, with the path-following task-acceleration given by* (11.28)*.*

Proof. We define the following error variables

$$\tilde{\mathbf{p}}_{\mathrm{LOS}} = \mathbf{p}_b - \mathbf{p}_v - \mathbf{\Lambda}_{i,3}^{-1}\mathbf{\Lambda}_{p,3}\mathbf{V}_c, \tag{11.29a}$$

$$\tilde{\mathbf{v}}_{\mathrm{LOS}} = \mathbf{v}_b + \mathbf{V}_c - \mathbf{v}_{\mathrm{LOS},d}. \tag{11.29b}$$

The time-derivatives of these variables are

$$\begin{bmatrix}\dot{\tilde{\mathbf{p}}}_{\mathrm{LOS}} \\ \dot{\tilde{\mathbf{v}}}_{\mathrm{LOS}}\end{bmatrix} = \mathbf{M}_{\mathrm{LOS}} \begin{bmatrix}\tilde{\mathbf{p}}_{\mathrm{LOS}} \\ \tilde{\mathbf{v}}_{\mathrm{LOS}}\end{bmatrix}, \qquad \mathbf{M}_{\mathrm{LOS}} = \begin{bmatrix}\mathbf{O} & \mathbf{I} \\ -\mathbf{\Lambda}_{i,3} & -\mathbf{\Lambda}_{p,3}\end{bmatrix} \tag{11.30}$$

The matrices $\mathbf{\Lambda}_{i,3}$ and $\mathbf{\Lambda}_{p,3}$ are positive definite by design. Consequently, the matrix $\mathbf{M}_{\mathrm{LOS}}$ is Hurwitz, and $\left[\tilde{\mathbf{p}}_{\mathrm{LOS}}^{\mathrm{T}}, \tilde{\mathbf{v}}_{\mathrm{LOS}}^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$ is a GES equilibrium of (11.30). From (11.29), we conclude that if $\tilde{\mathbf{v}}_{\mathrm{LOS}}$ exponentially converges to zero, then $\mathbf{v}_b$ exponentially converges to $\mathbf{v}_{\mathrm{LOS},d} - \mathbf{V}_c$. □

The desired acceleration of the path-following task is then given by

$$\dot{\mathbf{v}}_3 = \mathbf{1}_N \otimes \dot{\mathbf{v}}_{\mathrm{LOS}}. \tag{11.31}$$

Similarly to Chapter 5, the update of the path-parameter s is used as an extra degree of freedom to stability of the along-track error

$$\dot{s} = U_{\mathrm{LOS}} \left\| \frac{\partial \mathbf{p}_p(s)}{\partial s} \right\|^{-1} \left(\frac{\Delta(\mathbf{p}_b^p)}{D} + k_s \frac{x_b^p}{\sqrt{1 + (x_b^p)^2}} \right). \tag{11.32}$$

In Section 5.3, we showed that the LOS guidance law (11.25) guarantees uniform semiglobal exponential stability (USGES) of the path-following task.

Obstacle Avoidance

We implement an obstacle avoidance method that enables the fleet to avoid external obstacles while keeping the formation. This approach mitigates the issue of vehicles straying out of communication range while evading obstacles. We modify the collision cones method from Chapter 5 to be compatible with double integrator dynamics and focus on obstacle avoidance in the xy-plane.

We assume a constant velocity model for the obstacle. Its position and velocity vectors are denoted by $\mathbf{p}_o = [x_o, y_o, z_o]^{\mathrm{T}}$ and $\mathbf{v}_o = [\dot{x}_o, \dot{y}_o, \dot{z}_o]^{\mathrm{T}}$. We define an obstacle avoidance radius r_o that is large enough to account for both the size of the obstacle and the AUV. The formation radius r_f is defined as the maximum distance between any vehicle in the fleet and the formation center, and it is assumed to be constant throughout the avoidance maneuver. We further define $\mathbf{p}_{rel} = [x_b - x_o, y_b - y_o]^{\mathrm{T}}$, $\mathbf{v}_{rel} = [\dot{x}_{\mathrm{LOS},d} - \dot{x}_o, \dot{y}_{\mathrm{LOS},d} - \dot{y}_o]^{\mathrm{T}}$, and $\dot{\mathbf{v}}_{rel} = [\ddot{x}_{\mathrm{LOS},d}, \ddot{y}_{\mathrm{LOS},d}]^{\mathrm{T}}$. Note that $\mathbf{v}_{rel}$ is defined in terms of the desired LOS velocity (11.25), so $\dot{\mathbf{p}}_{rel} \neq \mathbf{v}_{rel}$.

Obstacle avoidance is guaranteed if we ensure that

$$\|\mathbf{p}_{rel}\| \geq r_o + r_f \tag{11.33}$$

throughout the avoidance maneuver (see Section 5.2.5). A conflict between the AUVs and the obstacle arises if $\mathbf{v}_{rel}$ lies in the collision cone, *i.e.,* if

$$|\angle(\mathbf{p}_{rel}, -\mathbf{v}_{rel})| \leq \alpha, \quad \alpha = \arcsin\left(\frac{r_o + r_f}{\|\mathbf{p}_{rel}\|}\right). \tag{11.34}$$

The obstacle avoidance task is activated if the cone angle satisfies $\alpha > \alpha_{\min}$. When the task is active, the x- and y-components of $\mathbf{v}_{\mathrm{LOS},d}$ and $\dot{\mathbf{v}}_{\mathrm{LOS},d}$ given by (11.25) and (11.26) are replaced with $\mathbf{v}_{OA,d}$ and $\dot{\mathbf{v}}_{OA,d}$, given by

$$\mathbf{v}_{OA,d} = \|\mathbf{v}_{rel}\| \left[\cos(\psi_{OA}), \sin(\psi_{OA})\right]^{\mathrm{T}} + [\dot{x}_o, \dot{y}_o]^{\mathrm{T}}, \tag{11.35}$$

$$\begin{aligned}\dot{\mathbf{v}}_{OA,d} = &\left(\tfrac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{v}_{rel}\|\right)\left[\cos(\psi_{OA}), \sin(\psi_{OA})\right]^{\mathrm{T}} \\ &+ \|\mathbf{v}_{rel}\|\left[-\sin(\psi_{OA})\dot{\psi}_{OA}, \cos(\psi_{OA})\dot{\psi}_{OA}\right]^{\mathrm{T}},\end{aligned} \tag{11.36}$$

where

$$\psi_{OA} = \arctan_2(y_o - y_b, x_o - x_b) \pm \alpha, \tag{11.37}$$

$$\dot{\psi}_{OA} = \frac{\det\left([\mathbf{p}_{rel}\ \dot{\mathbf{p}}_{rel}]\right)}{\|\mathbf{p}_{rel}\|^2} \pm \dot{\alpha}, \tag{11.38}$$

$$\dot{\alpha} = \frac{r_o + r_f}{\|\mathbf{p}_{rel}\|^2\sqrt{\|\mathbf{p}_{rel}\|^2 - (r_o + r_f)^2}}\mathbf{p}_{rel}^{\mathrm{T}}\dot{\mathbf{p}}_{rel}, \tag{11.39}$$

before entering into (11.27) and (11.28).

11.2.4 Closed-Loop Analysis

In this section, we analyze the closed-loop stability of the external states and the boundedness of the internal states. We assume that the inter-vehicle collision avoidance task is inactive for the analysis. As discussed in Section 2.3, the path-following and formation-keeping tasks are orthogonal. Therefore, the null-space projection $\mathbf{N}_2$ from the formation-keeping task will not affect the path-following acceleration $\mathbf{v}_3$

$$\dot{\mathbf{v}} = \dot{\mathbf{v}}_2 + \dot{\mathbf{v}}_3. \tag{11.40}$$

Let $\dot{\mathbf{v}}_{2,i}$ and $\dot{\mathbf{v}}_{3,i}$ denote the desired accelerations for the individual vehicles, *i.e.,*

$$\dot{\mathbf{v}}_2^{\mathrm{T}} = \left[\dot{\mathbf{v}}_{2,1}^{\mathrm{T}}, \ldots, \dot{\mathbf{v}}_{2,N}^{\mathrm{T}}\right], \qquad \dot{\mathbf{v}}_3^{\mathrm{T}} = \left[\dot{\mathbf{v}}_{3,1}^{\mathrm{T}}, \ldots, \dot{\mathbf{v}}_{3,N}^{\mathrm{T}}\right]. \tag{11.41}$$

Because the tasks are independent, the following two relations hold

$$\ddot{\boldsymbol{\sigma}}_2 = \mathbf{J}_2\dot{\mathbf{v}}_2 + \mathbf{J}_2\dot{\mathbf{v}}_3 = \mathbf{J}_2\dot{\mathbf{v}}_2, \tag{11.42}$$

$$\dot{\mathbf{v}}_b = \frac{1}{N}\sum_{i=1}^{N}(\dot{\mathbf{v}}_{2,i} + \dot{\mathbf{v}}_{3,i}) = \frac{1}{N}\sum_{i=1}^{N}\dot{\mathbf{v}}_{3,i}, \tag{11.43}$$

and the closed-loop properties of each task can thus be analyzed separately.

Stability of the Formation-Keeping Task

The closed-loop dynamics of the formation-keeping error $\tilde{\boldsymbol{\sigma}}_2$ under the saturated formation-keeping acceleration, (11.22), are given by the system

$$\ddot{\tilde{\boldsymbol{\sigma}}}_2 = -v_{2_{\max}} \mathrm{sat}\big(\boldsymbol{\Lambda}_{p,2}\tilde{\boldsymbol{\sigma}}_2\big) - \boldsymbol{\Lambda}_{d,2}\dot{\tilde{\boldsymbol{\sigma}}}_2. \tag{11.44}$$

Theorem 11.1. *Let $\boldsymbol{\Lambda}_{p,2}$, $\boldsymbol{\Lambda}_{d,2}$ be two symmetric positive definite matrices so that the product $\boldsymbol{\Lambda}_{p,2}\boldsymbol{\Lambda}_{d,2}$ is symmetric positive definite. Then, the point $\big[\dot{\tilde{\boldsymbol{\sigma}}}_2^{\mathrm{T}}, \tilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}\big] = \mathbf{0}^{\mathrm{T}}$ is a uniformly globally asymptotically stable (UGAS) equilibrium of the closed-loop system* (11.44).

Proof. Consider the Lyapunov function

$$V(\tilde{\boldsymbol{\sigma}}_2, \dot{\tilde{\boldsymbol{\sigma}}}_2) = v_{2,\max} \log\left(\cosh\|\boldsymbol{\Lambda}_{p,2}\tilde{\boldsymbol{\sigma}}_2\|\right) + \frac{1}{2}\dot{\tilde{\boldsymbol{\sigma}}}_2^{\mathrm{T}}\boldsymbol{\Lambda}_{p,2}\dot{\tilde{\boldsymbol{\sigma}}}_2. \tag{11.45}$$

Substituting (11.44), the time-derivative of V is given by

$$\begin{aligned}\dot{V} &= v_{2,\max}\mathrm{sat}\left(\boldsymbol{\Lambda}_{p,2}\tilde{\boldsymbol{\sigma}}_2\right)^{\mathrm{T}}\boldsymbol{\Lambda}_{p,2}\dot{\tilde{\boldsymbol{\sigma}}}_2 - \dot{\tilde{\boldsymbol{\sigma}}}_2^{\mathrm{T}}\boldsymbol{\Lambda}_{p,2}\left(v_{2,\max}\mathrm{sat}\left(\boldsymbol{\Lambda}_{p,2}\tilde{\boldsymbol{\sigma}}_2\right) + \boldsymbol{\Lambda}_{d,2}\dot{\tilde{\boldsymbol{\sigma}}}_2\right), \\ &= -\dot{\tilde{\boldsymbol{\sigma}}}_2^{\mathrm{T}}\boldsymbol{\Lambda}_{p,2}\boldsymbol{\Lambda}_{d,2}\dot{\tilde{\boldsymbol{\sigma}}}_2. \end{aligned} \tag{11.46}$$

Let $S = \{\big[\dot{\tilde{\boldsymbol{\sigma}}}_2^{\mathrm{T}}, \tilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}\big]^{\mathrm{T}} \in \mathbb{R}^{6(N-1)} | \dot{V} = 0\}$. Because of the dynamics (11.44), no other solution can stay identically in S, other than the trivial solution $\big[\dot{\tilde{\boldsymbol{\sigma}}}_2^{\mathrm{T}}, \tilde{\boldsymbol{\sigma}}_2^{\mathrm{T}}\big]^{\mathrm{T}} \equiv \mathbf{0}$. Thus, the origin is globally asymptotically stable according to [120, Corollary 4.2]. Furthermore, because (11.44) is time-invariant, the equilibrium is UGAS. □

Stability of the Path-Following Task

Let $\tilde{\mathbf{p}}_{\mathrm{LOS}}$ and $\tilde{\mathbf{v}}_{\mathrm{LOS}}$ be given by (11.29). Using the definition

$$\mathbf{p}_b^p = \mathbf{R}_p^{\mathrm{T}}(\mathbf{p}_b - \mathbf{p}_p), \tag{11.47}$$

we get the following error system

$$\begin{aligned}\dot{\tilde{\mathbf{p}}}_{\mathrm{LOS}} &= \tilde{\mathbf{v}}_{\mathrm{LOS}}, \\ \dot{\tilde{\mathbf{v}}}_{\mathrm{LOS}} &= -\boldsymbol{\Lambda}_{p,3}\tilde{\mathbf{v}}_{\mathrm{LOS}} - \boldsymbol{\Lambda}_{i,3}\tilde{\mathbf{p}}_{\mathrm{LOS}},\end{aligned} \tag{11.48a}$$

$$\begin{aligned}\dot{\mathbf{p}}_b^p &= \mathbf{R}_p^{\mathrm{T}}(\mathbf{v}_b + \mathbf{V}_c - \dot{\mathbf{p}}_p) + \big(\mathbf{S}(\omega_p\dot{s})\big)^{\mathrm{T}}\mathbf{R}_p^{\mathrm{T}}(\mathbf{p}_b - \mathbf{p}_p), \\ &= \mathbf{R}_p^{\mathrm{T}}(\mathbf{v}_{\mathrm{LOS},d} - \dot{\mathbf{p}}_p) - \mathbf{S}(\omega_p\dot{s})\mathbf{p}_b^p + \mathbf{R}_p^{\mathrm{T}}\tilde{\mathbf{v}}_{\mathrm{LOS}}.\end{aligned} \tag{11.48b}$$

Theorem 11.2. *Let $\boldsymbol{\Lambda}_{p,3}$, $\boldsymbol{\Lambda}_{i,3}$ be positive definite matrices. Then, the point $\big[\tilde{\mathbf{p}}_{\mathrm{LOS}}^{\mathrm{T}}, \tilde{\mathbf{v}}_{\mathrm{LOS}}^{\mathrm{T}}, (\mathbf{p}_b^p)^{\mathrm{T}}\big] = \mathbf{0}^{\mathrm{T}}$ is a USGES equilibrium of the system* (11.48a)-(11.48b).

Proof. Note that the error system is in a cascaded form where the velocity error $\tilde{\mathbf{v}}_{\mathrm{LOS}}$ from (11.48a) perturbs the system (11.48b). The dynamics of the perturbing system (11.48a) are GES according to Lemma 11.2. The nominal system (11.48b) with $\tilde{\mathbf{v}}_{\mathrm{LOS}} = \mathbf{0}$ was proved to be USGES in Section 5.3 using the following Lyapunov function

$$V(\mathbf{p}_b^p) = \frac{1}{2}\left(\mathbf{p}_b^p\right)^{\mathrm{T}}\mathbf{p}_b^p. \tag{11.49}$$

Therefore, if the two assumptions of Proposition 2.2 ([70, Proposition 9]) hold, the origin of the cascade is USGES.

Because $\|\partial V/\partial \mathbf{p}_b^p\| = \|\mathbf{p}_b^p\|$, the first assumption (2.111) is satisfied with $c_1 = 2$, $c_2 = \eta$ for any $\eta \in \mathbb{R}_{\geq 0}$.

The second assumption (2.112) holds with $\alpha_1(\|\tilde{\mathbf{v}}_{\mathrm{LOS}}\|) = \|\tilde{\mathbf{v}}_{\mathrm{LOS}}\|, \alpha_2(\|\tilde{\mathbf{v}}_{\mathrm{LOS}}\|) = 0$, because the norm of the perturbing term is given by $\|\mathbf{R}_p^{\mathrm{T}}\tilde{\mathbf{v}}_{\mathrm{LOS}}\| = \|\tilde{\mathbf{v}}_{\mathrm{LOS}}\|$. As a result, all assumptions of [70, Proposition 9] are satisfied, and the origin of the closed-loop path-following system (11.48a)-(11.48b) is USGES. □

Boundedness of Internal States

Let $\mathbf{p}_{\mathrm{d},i} = \mathbf{p}_p(s) + \mathbf{R}_p\mathbf{p}_{f,i}^f$ denote the desired position of vehicle i. Note that because the path function is $\mathcal{C}^\infty$ and thanks to the choice of the path parameter update law (11.32), the time-derivative of $\mathbf{p}_{\mathrm{d},i}$ is bounded. In the previous section, we proved the stability of the external dynamics. Consequently, the hand position of vehicle i, $\mathbf{x}_{1,i}$, converges to $\mathbf{p}_{\mathrm{d},i}$, and the relative hand velocity of vehicle i, $\mathbf{x}_{2,i}$, converges to $\dot{\mathbf{p}}_{\mathrm{d},i} - \mathbf{V}_c$.

Proposition 11.1. *Let us define* $\mathbf{x}_{2,d,i} = \dot{\mathbf{p}}_{\mathrm{d},i} - \mathbf{V}_c$, *and* $a_{x,i}$, $\bar{\alpha}_{y,i}$, *and* $\bar{\alpha}_{z,i}$ *for each vehicle* $i = 1, \ldots, N$ *in accordance with* (7.27). *The internal dynamics of the vehicles are ultimately bounded if* $a_{x,i}, \bar{\alpha}_{y,i}, \bar{\alpha}_{z,i} > 0$ *for all* $i \in \{1, \ldots, N\}$.

Proof. In the previous sections, we showed that in the nominal case, the external dynamics are UGAS. Moreover, for a given set of initial conditions, the control input $\boldsymbol{\mu}$ defined in (11.7) is bounded. Consequently, if $a_{x,i}, \bar{\alpha}_{y,i}, \bar{\alpha}_{z,i} > 0$ for all $i \in \{1, \ldots, N\}$, then all assumptions of Lemma 7.2 are satisfied, and the angular rate dynamics are ultimately bounded. □

11.2.5 Simulation Results

To validate the theoretical results, we perform simulations where the proposed algorithm is applied to a fleet of three light autonomous underwater vehicles (LAUVs) [12]. In the simulated scenario, the barycenter should follow a spiral path while avoiding collision with a stationary cylindrical-shaped obstacle with radius 10 m, located at $[x_o, y_o] = [100, -10]$. All position variables are here given in meters. The spiral is given by

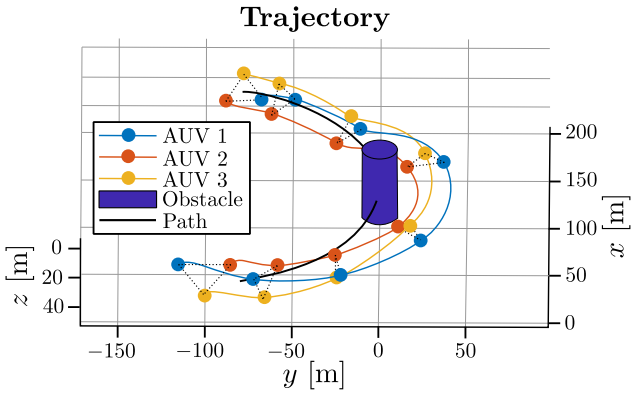
$$\mathbf{p}_p(s) = \mathbf{p}_{p,0} + \left[s, -40\cos(\tfrac{\pi}{100}s), 20\sin(\tfrac{\pi}{100}s)\right]^{\mathrm{T}}, \tag{11.50}$$

where

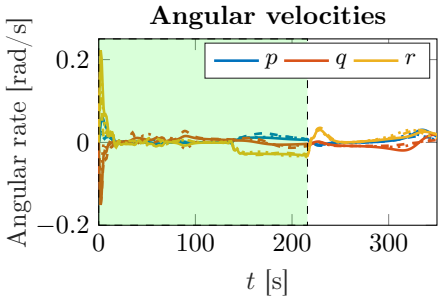
$$\mathbf{p}_{p,0} = \left[0, -40, 35\right]^{\mathrm{T}}. \tag{11.51}$$

The barycenter relative formation is given by

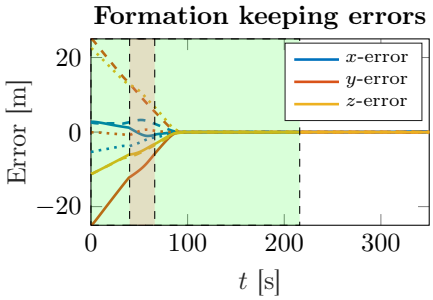
$$\mathbf{p}_{f,1}^f = \begin{bmatrix} 0 \\ 10 \\ 5 \end{bmatrix}, \quad \mathbf{p}_{f,2}^f = \begin{bmatrix} 0 \\ -10 \\ 5 \end{bmatrix}, \quad \mathbf{p}_{f,3}^f = \begin{bmatrix} 0 \\ 0 \\ -10 \end{bmatrix}, \tag{11.52}$$



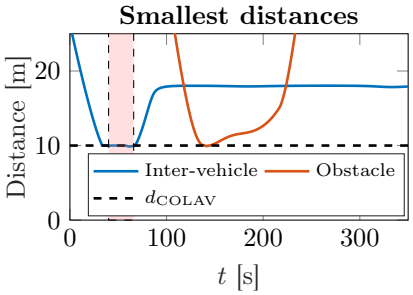
(a) The trajectory of the vehicles. The markers represent the vehicle positions every 50 seconds.



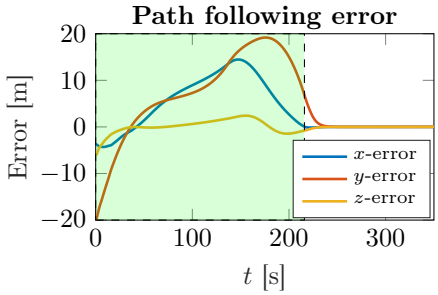
(b) The angular velocities of the vehicles.



(c) The formation keeping errors.



(d) The minimum inter-vehicle and vehicle to obstacle distance.



(e) The barycenter path-following error.

Figure 11.1: Simulation results of the path-following algorithm proposed in Section 11.2. The full, dashed, and dotted lines correspond to the three different vehicles. The green and red rectangles represent when obstacle avoidance and inter-vehicle COLAV is active.

and we want the collision avoidance task to ensure a safe distance of 10 m, both between vehicles in the fleet and external obstacles. Therefore, the avoidance radius of the cylinder, r_o, is 20 m. The vehicles are subject to an ocean current

$$\mathbf{V}_c = \begin{bmatrix} 0 & 0.25 & 0.05 \end{bmatrix}^{\mathrm{T}} \text{ m/s}. \tag{11.53}$$

The resulting trajectory of the mission is shown in Figure 11.1a. The vehicles avoid the obstacle with a margin and return to the desired path. Figure 11.1b shows that the angular velocities remain bounded, in accordance with Proposition 11.1. Figure 11.1c shows that the fleet converges to the desired formation while the obstacle avoidance mode is active. Except for during the inter-vehicle collision avoidance, the convergence seems linear, which can be expected because the task velocity is saturated by $v_{2,\max}$. Figure 11.1d shows that the inter-vehicle COLAV task activates when the distance between vehicles is below d_{COLAV}, and the distance does not decrease further. Because the obstacle avoidance radius r_o was chosen 10 m wider than the obstacle, the obstacle is avoided with a 10 m margin. Figure 11.1e shows that the path-following error initially increases as the fleet avoids the obstacle because the x- and y-components of $\mathbf{v}_{\mathrm{LOS},d}$ and $\dot{\mathbf{v}}_{\mathrm{LOS},d}$ are replaced with $\mathbf{v}_{OA,d}$ and $\dot{\mathbf{v}}_{OA,d}$ given by (11.35), (11.36). As expected from Theorem 11.2, the error converges to zero after the obstacle is passed when the LOS task is activated again.

Chapter 12

Conclusions and Future Work

We know everything, that is,
that we know nothing.

L. Smoljak, "Vyšetřování ztráty třídní knihy," 1967 (translated from Czech).

In this thesis, we presented and analyzed multiple control algorithms. In Chapter 3, we proposed a method that unifies reactive collision avoidance and control allocation. Then, in Part I (Chapters 4–6), we presented three different types of null-space-behavioral algorithms. Finally, in Part II (Chapters 7–11), we introduced the hand position concept to underactuated underwater vehicles and presented four applications. We conclude the thesis by presenting some remarks for each chapter and suggestions for future work.

Chapter 3: Unifying Reactive Collision Avoidance and Control Allocation

In Chapter 3, we proposed a method for integrating a collision avoidance (COLAV) scheme into control allocation through the use of control barrier functions (CBFs). We demonstrated its effectiveness on two models of autonomous surface vehicles (ASVs), where it significantly improved the safety. The proposed method can be readily implemented on vehicles that already use optimization-based control allocation by simply including the constraints given by the control barrier functions (CBFs) in the optimization.

We note that the performance of the proposed scheme depends on the choice of the parameters, such as the class-$\mathcal{K}_\infty$ function γ, and the weight matrices $\mathbf{Q}$, $\mathbf{R}_{\text{abs}}$, and $\mathbf{R}_{\text{rel}}$. Finding a systematic method for choosing the parameter values that guarantee safety for a given vehicle model is a topic for future work.

Chapter 4: Formation Path-Following Control of 5DOF Underactuated AUVs

In Chapter 4, we proposed a formation path-following method for an arbitrary number of AUVs, proved the stability of the path-following part, and verified its

effectiveness in simulations.

Because the proposed algorithm is centralized, our method can only be used in scenarios where all the vehicles can communicate with each other. A distributed version of the algorithm is introduced later in Chapter 6.

In the simulations, the formation-keeping error shows exponential convergence to zero. However, the stability of the formation-keeping task has not been theoretically proven. A modified version of the algorithm with provable stability of the formation-keeping task is presented in Chapter 5.

Chapter 5: Formation Path-Following Control of 6DOF Underactuated AUVs

This chapter extended the formation path-following NSB algorithm to underactuated 6DOF vehicles while adding obstacle avoidance and depth-limiting capabilities. Both the path-following and formation-keeping parts were proven to be stable. In the proofs, we assumed that the avoidance and depth-limiting tasks are not active. An analysis of the closed-loop system with active avoidance and depth-limiting tasks is left for future work.

Chapter 6: A Distributed NSB Algorithm for Formation Path Following

In Chapter 6, we discussed how to combine null-space-behavioral control with consensus, and in this way solve the formation path-following problem in a fully distributed fashion. We also found that it is possible to implement this concept in two different ways; using a continuous-time consensus algorithm, or a discrete-time one (the latter being more suitable for real-life implementations). Using Lyapunov analysis, we showed that in the special case of straight-line paths, the continuous-time version achieves uniform semiglobal exponential stability. The discrete-time version is based on event-triggered paradigms, to account for practical limitations in the way agents exchange information.

Both versions were then verified in simulations. Comparing the discrete-time version to a similar cooperative path-following algorithm presented in [25], we found that the proposed algorithm requires fewer transmissions between the vehicles, while having similar steady-state error. Finally, we demonstrated the real-life effectiveness of the discrete-time algorithm in field experiments with a fleet of light autonomous underwater vehicles.

Future work includes extending the stability proofs to the more general case of curved paths and more complex vehicle dynamics, as well as investigating the effects of the event-triggered scheme on the performance of the algorithm. In addition, we plan to perform additional experiments with more vehicles and underwater communications.

Chapter 7: Hand Position for Underactuated Underwater Vehicles

In this section, we extended the hand position concept to 6DOF underactuated underwater vehicles. By choosing the hand position as the output of our system, we could apply output feedback linearization to simplify the underactuated

6DOF vehicle dynamics to a double integrator without introducing any singularities. Then, we derived sufficient conditions under which the internal states are ultimately bounded.

As mentioned in the Introduction, the hand position concept and its ability to transform a nonlinear underactuated model to a double integrator without singularities present an opportunity to utilize numerous control strategies that could otherwise not be used on nonholonomic or underactuated vehicles. Examples of such controllers were presented in Chapters 8–11.

Chapter 8: Trajectory Tracking and Path Following using the Hand Position Concept

In this chapter, we showed how the hand position transformation combined with a simple PID-based controller can be used to solve both the trajectory-tracking and path-following control problems. Using Lyapunov analysis, we proved the exponential stability of the external dynamics. Moreover, in the special case of straight-line trajectories and paths, we could modify the controllers and prove the exponential stability of the total system. The proposed controllers were tested both in numerical simulations and experiments.

Chapter 9: Distributed MPC for Formation Path-Following of Multi-Vehicle Systems

In this chapter, we proposed a distributed spline-based MPC scheme for the formation path-following problem. We showed that using splines makes the distributed control problem computationally tractable. Compared to collocation, the spline parametrization allows us to represent a longer prediction horizon using fewer variables. This is also beneficial for the communication, and thus makes it easier to do distributed control in environments where the communication bandwidth is limited (*e.g.,* underwater).

One might argue that restricting the output to splines limits the subspace of feasible trajectories. However, simulation results show that cubic splines provide a good approximation of many curves. Another limiting factor is the need for differential flatness. However, it is often possible to simplify the structure of the model to guarantee differential flatness. The proposed spline-based MPC scheme can thus be seen as a trade-off between lower computational requirements and more restrictive assumptions on the model.

Chapter 10: Control of AUVs Under Hard and Soft Constraints

In this chapter, we addressed the tracking-in-formation control problem of cooperative autonomous underwater vehicles interacting over directed graphs and under *hard* inter-agent constraints (proximity and collision avoidance) and *soft* constraints (positive surge velocity). We proposed a distributed control law that solves this problem and that guarantees, simultaneously, connectivity preservation and inter-agent collision avoidance. With respect to the stability analysis, it is important to emphasize that, beyond mere convergence properties as usually established in the literature of multi-agent systems, we establish almost-everywhere

uniform asymptotic stability with exponential convergence of the tracking errors. Current research focuses on validating the results experimentally.

Chapter 11: Combining NSB with the Hand Position Approach

In this chapter, we proposed an extended NSB method for double-integrator systems. The method was proved to provide globally exponentially stable task error dynamics. The method was demonstrated in a case study of formation path-following with underactuated AUVs. We defined the second-order kinematic tasks for collision avoidance, formation-keeping, and path-following. Compared to the methods proposed in Chapters 4–6 where the NSB algorithm is perturbed by the low-level controller, the second-order NSB method handles vehicle dynamics directly in the task space. The closed-loop behavior of the task variables can be interpreted as a mass-spring-damper system. Consequently, the control gains can be chosen such that the closed-loop system is critically damped.

To guarantee a bounded velocity, we introduced a saturation term to the acceleration of the formation-keeping task. The closed-loop formation-error system with the reformulated formation-keeping acceleration was proved to be uniformly globally asymptotically stable, and the closed-loop path-following system was proved to be uniformly semiglobally exponentially stable. Simulation results demonstrate the effectiveness of our approach. Possible future work includes verifying the presented results through experiments.

Appendix A

AUV Model

This appendix lists the equations that are too extensive to be included in the main body of the thesis.

A.1 The Component Form

A.1.1 6DOF Model

The functions F_u, X_v, Y_v, Z_v, X_w, Y_w, Z_w, G, F_p, F_q, and F_r in (2.22) in Section 2.1.5 are given by

$$F_u(\cdot) = -\frac{m_{35}q^2 + m_{33}w_r q - m_{26}r^2 - m_{22}v_r r + d_{11}u_r}{m_{11}}, \tag{A.1a}$$

$$X_v(\cdot) = -\frac{d_{26}m_{66} - d_{66}m_{26} + u_r\left(m_{11}m_{66} - m_{26}^2\right)}{m_{22}m_{66} - m_{26}^2}, \tag{A.1b}$$

$$Y_v(\cdot) = -\frac{d_{22}m_{66} - d_{62}m_{26} + u_r\left(m_{11}m_{26} - m_{22}m_{26}\right)}{m_{22}m_{66} - m_{26}^2}, \tag{A.1c}$$

$$Z_v(\cdot) = \frac{p\left(m_{26}m_{35} + m_{33}m_{66}\right)}{m_{22}m_{66} - m_{26}^2}, \tag{A.1d}$$

$$X_w(\cdot) = -\frac{d_{35}m_{55} - d_{55}m_{35} - u_r\left(m_{11}m_{55} - m_{36}^2\right)}{m_{33}m_{55} - m_{35}^2}, \tag{A.1e}$$

$$Y_w(\cdot) = -\frac{d_{33}m_{55} - d_{53}m_{35} - u_r\left(m_{11}m_{35} - m_{33}m_{35}\right)}{m_{33}m_{55} - m_{35}^2}, \tag{A.1f}$$

$$Z_w(\cdot) = -\frac{p\left(m_{26}m_{35} + m_{22}m_{55}\right)}{m_{33}m_{55} - m_{35}^2}, \tag{A.1g}$$

$$G(\cdot) = -\frac{m_{35}}{m_{33}m_{55} - m_{35}^2} W z_{gb}\left[0,1,0\right]\left(\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right) \tag{A.1h}$$

$$\begin{aligned} F_p(\cdot) = &-\frac{d_{44}p - m_{55}qr + m_{66}qr + m_{26}qv_r + m_{35}qv_r - m_{26}rw_r}{m_{44}} \\ &+ \frac{m_{35}rw_r + m_{22}v_r w_r - m_{33}v_r w_r + W z_{gb}\left[1,0,0\right]\left(\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}}\mathbf{e}_3\right)}{m_{44}}, \end{aligned} \tag{A.1i}$$

$$F_q(\cdot) = \frac{m_{33}^2 u_r w_r + d_{35} m_{35} q - d_{55} m_{33} q + d_{33} m_{35} w_r - d_{53} m_{33} w_r}{m_{33} m_{55} - {m_{35}}^2} \tag{A.1j}$$
$$+ \frac{m_{26} m_{35} pr - m_{33} m_{44} pr + m_{33} m_{66} pr - m_{11} m_{35} q u_r}{m_{33} m_{55} - {m_{35}}^2}$$
$$+ \frac{m_{22} m_{35} p v_r + m_{26} m_{33} p v_r + m_{33} m_{35} q u_r}{m_{33} m_{55} - {m_{35}}^2}$$
$$+ \frac{m_{11} m_{33} u_r w_r + m_{33} W z_{gb} \left[0, 1, 0\right] \left(\mathbf{e}_3 \times \mathbf{R}^{\mathrm{T}} \mathbf{e}_3\right)}{m_{33} m_{55} - {m_{35}}^2},$$

$$F_r(\cdot) = -\frac{m_{22}^2 u_r v_r - d_{26} m_{26} r + d_{66} m_{22} r - d_{22} m_{26} v_r}{m_{22} m_{66} - {m_{26}}^2} \tag{A.1k}$$
$$- \frac{d_{62} m_{22} v_r + m_{26} m_{35} pq - m_{22} m_{44} pq + m_{22} m_{55} pq}{m_{22} m_{66} - {m_{26}}^2}$$
$$- \frac{m_{22} m_{26} r u_r - m_{11} m_{26} r u_r + m_{22} m_{35} p w_r}{m_{22} m_{66} - {m_{26}}^2}$$
$$- \frac{m_{26} m_{33} p w_r - m_{11} m_{22} u_r v_r}{m_{22} m_{66} - {m_{26}}^2}.$$

A.1.2 3DOF Model

The functions F_u, X_v, Y_v, and F_r in (2.43) are given by

$$F_u(\cdot) = \frac{m_{23} r^2 + m_{22} v_r r - d_{11} u_r}{m_{11}}, \tag{A.2a}$$

$$X_v(\cdot) = \frac{(m_{23}^2 + m_{11} m_{33}) u_r + d_{33} m_{23} - d_{23} m_{33}}{m_{22} m_{33} - m_{23}^2}, \tag{A.2b}$$

$$Y_v(\cdot) = \frac{d_{22} m_{33} - d_{32} m_{23} + (m_{11} m_{23} - m_{22} m_{23}) u_r}{m_{22} m_{33} - m_{23}^2}, \tag{A.2c}$$

$$F_r(\cdot) = -\frac{m_{22}^2 u_r v_r - d_{23} m_{23} r + d_{33} m_{22} r - d_{22} m_{23} v_r + d_{32} m_{22} v_r}{m_{22} m_{33} - m_{23}^2} \tag{A.2d}$$
$$- \frac{m_{22} m_{23} r u_r - m_{11} m_{23} r u_r - m_{11} m_{22} u_r v_r}{m_{22} m_{33} - m_{23}^2}.$$

Appendix B

Proofs of Lemmas from Chapter 4

B.1 Derivation of Closed-Loop Barycenter Kinematics

We begin by taking $\dot{y}_b^p$ from (4.31b).

$$\dot{y}_b^p = \frac{1}{n}\sum_{i=1}^{n} U_i \cos(\gamma_i) \sin(\chi_i - \psi_p) - \dot{\xi}\, \iota\, x_b^p. \tag{B.1}$$

Now, consider the term $\sin(\chi_i - \psi_p)$. The course of the vessel is given by

$$\chi_i = \psi_i + \beta_i, \qquad \beta_i = \arcsin\left(\frac{v_i}{U_i}\right). \tag{B.2}$$

After substituting and applying some trigonometric identities, we get

$$\begin{aligned}
\sin(\chi_i - \psi_p) &= \sin(\psi_i + \beta_i - \psi_p) && \text{(B.3a)}\\
&= \cos(\psi_i - \psi_p) \sin(\beta_i) + \sin(\psi_i - \psi_p) \cos(\beta_i) && \text{(B.3b)}\\
&= \cos(\psi_i - \psi_p) \frac{v_i}{U_i} + \sin(\psi_i - \psi_p) \frac{\sqrt{u_i^2 + w_i^2}}{U_i}. && \text{(B.3c)}
\end{aligned}$$

Consequently, the term $U_i \cos(\gamma_i) \sin(\chi_i - \psi_p)$ is equivalent to

$$\begin{aligned}
U_i \cos(\gamma_i) \sin(\chi_i - \psi_p) = \cos(\gamma_i) \Big(&\cos(\psi_i - \psi_p)\, v_i \\
&+ \sin(\psi_i - \psi_p) \sqrt{u_i^2 + w_i^2} \Big).
\end{aligned} \tag{B.4}$$

Now, consider a term $\sin(\psi_i + \beta_{d,i} - \psi_p)$. Using a similar procedure, we get

$$\sin(\psi_i + \beta_{d,i} - \psi_p) = \cos(\psi_i - \psi_p) \frac{v_i}{U_{d,i}} + \sin(\psi_i - \psi_p) \frac{\sqrt{u_{d,i}^2 + w_i^2}}{U_{d,i}}. \tag{B.5}$$

Combining (B.4) and (B.5), we get

$$\begin{aligned}
U_i \cos(\gamma_i) \sin(\chi_i - \psi_p) = U_{d,i} &\cos(\gamma_i) \sin(\psi_i + \beta_{d,i} - \psi_p) \\
&+ \cos(\gamma_i) \sin(\psi_i - \psi_p) \left(\sqrt{u_i^2 + w_i^2} - \sqrt{u_{d,i}^2 + w_i^2} \right).
\end{aligned} \tag{B.6}$$

Note that the following holds for the angles

$$\begin{aligned}\psi_i + \beta_{d,i} - \psi_p &= \psi_{d,i} + \tilde{\psi}_i + \beta_{d,i} - (\psi_{d,i} + \beta_{d,i} + \beta_{\mathrm{LOS}}) = \tilde{\psi}_i - \beta_{\mathrm{LOS}},\\ \beta_{\mathrm{LOS}} &= \arctan\left(\frac{y_b^p}{\Delta(\mathbf{p}_b^p)}\right).\end{aligned} \tag{B.7}$$

Therefore, their sine is given by

$$\sin(\psi_i + \beta_{d,i} - \psi_p) = \sin\left(\tilde{\psi}_i\right) \frac{\Delta(\mathbf{p}_b^p)}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2}} - \cos\left(\tilde{\psi}_i\right) \frac{y_b^p}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2}}. \tag{B.8}$$

Furthermore, note that the following holds for the flight-path angle

$$\gamma_i = \theta_i - \alpha_i = \tilde{\theta}_i + \theta_{d,i} - \alpha_i = \tilde{\theta}_i + \gamma_{\mathrm{LOS}} + \alpha_{d,i} - \alpha_i. \tag{B.9}$$

Consequently, the cosine of the flight-path angle is equal to

$$\begin{aligned}\cos(\gamma_i) = &\cos(\gamma_{\mathrm{LOS}}) \cos\left(\tilde{\theta}_i\right) \cos(\alpha_{d,i} - \alpha_i)\\ &- \cos(\gamma_{\mathrm{LOS}}) \sin\left(\tilde{\theta}_i\right) \sin(\alpha_{d,i} - \alpha_i)\\ &- \sin(\gamma_{\mathrm{LOS}}) \cos\left(\tilde{\theta}_i\right) \sin(\alpha_{d,i} - \alpha_i)\\ &- \sin(\gamma_{\mathrm{LOS}}) \sin\left(\tilde{\theta}_i\right) \cos(\alpha_{d,i} - \alpha_i)\end{aligned} \tag{B.10}$$

Using the equalities (B.8), (B.10), we can rewrite (B.6) as

$$\begin{aligned}U_i \cos(\gamma_i) \sin(\chi_i - \psi_p) = &-U_{d,i} \cos(\gamma_{\mathrm{LOS}}) \frac{y_b^p}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2}}\\ &+ G_{y,i}\left(\tilde{u}_i, \tilde{\psi}_i, \gamma_i, u_{d,i}, v_i, w_i, \mathbf{p}_b^p, \psi_p\right),\end{aligned} \tag{B.11}$$

where

$$\begin{aligned}G_{y,i}(\cdot) = &\cos(\gamma_i) \sin(\psi_i - \psi_p) \left(\sqrt{u_i^2 + w_i^2} - \sqrt{u_{d,i}^2 + w_i^2}\right)\\ &- U_{d,i} \cos(\gamma_i) \sin\left(\tilde{\psi}_i\right) \frac{\Delta(\mathbf{p}_b^p)}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2}}\\ &+ U_{d,i}\Big[\sin(\gamma_{\mathrm{LOS}})\Big(\cos\Big(\tilde{\theta}_i\Big)\sin(\alpha_{d,i} - \alpha_i) + \sin\Big(\tilde{\theta}_i\Big)\cos(\alpha_{d,i} - \alpha_i)\Big)\\ &\quad - \cos(\gamma_{\mathrm{LOS}}) \left(\cos\left(\tilde{\theta}_i\right) \cos(\alpha_{d,i} - \alpha_i) - 1\right)\Big] \frac{y_b^p}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2}}\end{aligned} \tag{B.12}$$

Substituting (B.11) into (B.1), we get the following

$$\begin{aligned}\dot{y}_b^p = &-\frac{1}{n} \sum_{i=1}^{n} U_{d,i} \cos(\gamma_{\mathrm{LOS}}) \frac{y_b^p}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2}} - \dot{\xi}\, \iota\, x_b^p\\ &+ G_y\Big(\tilde{u}_1, \ldots, \tilde{u}_n, \tilde{\psi}_1, \ldots, \tilde{\psi}_n, \gamma_1, \ldots, \gamma_n, u_{d,1}, \ldots, u_{d,n},\\ &\qquad v_1, \ldots, v_n, w_1, \ldots, w_n, \mathbf{p}_b^p, \psi_p\Big),\end{aligned} \tag{B.13}$$

where

$$G_y(\cdot) = \frac{1}{n}\sum_{i=1}^{n} G_{y,i}\left(\tilde{u}_i, \tilde{\psi}_i, \gamma_i, u_{d,i}, v_i, w_i, \mathbf{p}_b^p, \psi_p\right). \tag{B.14}$$

Now, we demonstrate a similar procedure for $\dot{z}_b^p$. From (4.31b), we get

$$\begin{aligned}\dot{z}_b^p &= \frac{1}{n}\sum_{i=1}^{n} U_i\left(-\cos(\theta_p)\sin(\gamma_i) + \cos(\gamma_i)\sin(\theta_p)\cos(\psi_p - \chi_i)\right) + \dot{\xi}\,\kappa\, x_b^p \\ &= \frac{1}{n}\sum_{i=1}^{n} U_i\left(-\sin(\gamma_i - \theta_p) - (1-\cos(\chi_i - \psi_p))\cos(\gamma_i)\sin(\theta_p)\right) + \dot{\xi}\,\kappa\, x_b^p.\end{aligned} \tag{B.15}$$

Once again, we consider the terms

$$\sin(\gamma_i - \theta_p) = \sin(\theta_i - \alpha_i - \theta_p) = \sin(\theta_i - \theta_p)\frac{u_i}{U_i} - \cos(\theta_i - \theta_p)\frac{w_i}{U_i}, \tag{B.16}$$

and

$$\sin(\theta_i - \alpha_{d,i} - \theta_p) = \sin(\theta_i - \theta_p)\frac{u_{d,i}}{U_{d,i}} - \cos(\theta_i - \theta_p)\frac{w_i}{U_{d,i}}, \tag{B.17}$$

which give us the following equality

$$U_i\sin(\gamma_i - \theta_p) = U_{d,i}\sin(\theta_i - \alpha_{d,i} - \theta_p) + \tilde{u}_i\sin(\theta_i - \theta_p). \tag{B.18}$$

Using a similar trick, we can write the sine as

$$\sin(\theta_i - \alpha_{d,i} - \theta_p) = \sin\left(\tilde{\theta}_i\right)\frac{\Delta(\mathbf{p}_b^p)}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (z_b^p)^2}} - \cos\left(\tilde{\theta}_i\right)\frac{(z_b^p)}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (z_b^p)^2}} \tag{B.19}$$

Consequently, we can rewrite (B.15) as

$$\begin{aligned}\dot{z}_b^p = &-\frac{1}{n}\sum_{i=1}^{n} U_{d,i}\frac{z_b^p}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(z_b^p\right)^2}} + \dot{\xi}\,\kappa\, x_b^p \\ &+ G_z\Big(\tilde{u}_1, \ldots, \tilde{u}_n, \tilde{\theta}_1, \ldots, \tilde{\theta}_n, \gamma_1, \ldots, \gamma_n, \chi_1, \ldots, \chi_n, \\ &\qquad u_{d,1}, \ldots, u_{d,n}, v_1, \ldots, v_n, w_1, \ldots, w_n, \mathbf{p}_b^p, \theta_p, \psi_p\Big),\end{aligned} \tag{B.20}$$

where

$$G_z(\cdot) = \frac{1}{n}\sum_{i=1}^{n} G_{z,i}\left(\tilde{u}_i, \tilde{\theta}_i, \gamma_i, \chi_i, u_{d,i}, v_i, w_i, \mathbf{p}_b^p, \theta_p, \psi_p\right), \tag{B.21}$$

$$\begin{aligned}G_{z,i}(\cdot) = &-U_i\left((1-\cos(\chi_i - \psi_p))\cos(\gamma_i)\sin(\theta_p)\right) - \tilde{u}_i\sin(\theta_i - \theta_p) \\ &-\left(1-\cos\left(\tilde{\theta}_i\right)\right)\frac{(z_b^p)}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (z_b^p)^2}} - U_{d,i}\sin\left(\tilde{\theta}_i\right)\frac{\Delta(\mathbf{p}_b^p)}{\sqrt{\Delta(\mathbf{p}_b^p)^2 + (z_b^p)^2}}.\end{aligned} \tag{B.22}$$

B.2 Desired Pitch and Yaw Rate

For further calculations, we need to evaluate the desired pitch ($q_{d,i}$) and yaw ($r_{d,i}$) rates of the vessels. From (4.2d), we get the following relation between the yaw rate and the derivative of the yaw angle

$$q_{d,i} = \dot{\theta}_{d,i}. \tag{B.23}$$

Now, we consider the desired pitch angle from (4.27). Since we are investigating the path following task, we substitute γ_{LOS} from (4.22) for $\gamma_{\mathrm{NSB},i}$. Differentiating (4.27) with respect to time yields

$$q_{d,i} = \dot{\theta}_p(\xi) + \frac{\Delta\left(\mathbf{p}_b^p\right)\, \dot{z}_b^p - z_b^p\, \dot{\Delta}\left(\mathbf{p}_b^p\right)}{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(z_b^p\right)^2} + \frac{u_{d,i}\, \dot{w}}{u_{d,i}^2 + w_i^2}, \tag{B.24}$$

which can be expanded to

$$\begin{aligned} q_{d,i} = \dot{\xi}\, \kappa(\xi) &+ \frac{\Delta\left(\mathbf{p}_b^p\right)\left(\frac{1}{n}\sum_{j=1}^{n} U_{d,j} \frac{\left(z_b^p\right)}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2+\left(z_b^p\right)^2}} + \dot{\xi}\, \kappa\, x_b^p + G_z(\cdot)\right)}{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(z_b^p\right)^2} \\ &+ \frac{z_b^p\left(-k_\xi \frac{\left(x_b^p\right)^2}{\sqrt{1+\left(x_b^p\right)^2}} - \frac{1}{n}\sum_{j=1}^{n} U_{d,j}\left(\frac{\cos(\gamma_{\mathrm{LOS},j})^2\left(y_b^p\right)^2}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2+\left(y_b^p\right)^2}} + \frac{\left(z_b^p\right)^2}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2+\left(z_b^p\right)^2}}\right)\right)}{\Delta\left(\mathbf{p}_b^p\right)\left(\Delta\left(\mathbf{p}_b^p\right)^2 + \left(z_b^p\right)^2\right)} \\ &+ \frac{z_b^p(y_b^p\, G_y(\cdot) + z_b^p\, G_z(\cdot))}{\Delta(\mathbf{p}_b^p)\left(\Delta(\mathbf{p}_b^p)^2+\left(z_b^p\right)^2\right)} \\ &+ u_{d,i}\frac{X_w(u_{d,i}+\tilde{u}_i, u_c)q + Y_w(u_{d,i}+\tilde{u}_i, u_c)(w_i - w_c)}{u_{d,i}^2 + w_i^2}. \end{aligned} \tag{B.25}$$

From (4.2e), we get the following relation between the yaw rate and the derivative of the yaw angle

$$r_{d,i} = \dot{\psi}_{d,i}\, \cos\left(\theta_{d,i}\right). \tag{B.26}$$

Substituting the time-derivative of (4.28), we get

$$r_{d,i} = \left(\dot{\psi}_p(\xi) - \frac{\Delta(\mathbf{p}_b^p)\, \dot{y}_b^p - y_b^p\, \dot{\Delta}\left(\mathbf{p}_b^p\right)}{\Delta(\mathbf{p}_b^p)^2 + \left(y_b^p\right)^2} - \frac{\dot{v}}{\sqrt{U_{d,i}^2 - v_i^2}}\right) \cos\left(\theta_{d,i}\right) \tag{B.27a}$$

$$
\begin{aligned}
= \Bigg[& \dot{\xi}\,\iota(\xi) - \frac{\Delta(\mathbf{p}_b^p)\left(\frac{1}{n}\sum_{j=1}^{n} U_{d,i}\frac{\cos(\gamma_{\mathrm{LOS}})(y_b^p)}{\sqrt{\Delta(\mathbf{p}_b^p)^2+(y_b^p)^2}} - \dot{\xi}\,\iota\,x_b^p + G_y(\cdot)\right)}{\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2} \\
& + \frac{y_b^p\left(-k_\xi\frac{(x_b^p)^2}{\sqrt{1+(x_b^p)^2}} - \frac{1}{n}\sum_{j=1}^{n} U_{d,i}\left(\frac{\cos(\gamma_{\mathrm{LOS}})^2(y_b^p)^2}{\sqrt{\Delta(\mathbf{p}_b^p)^2+(y_b^p)^2}} + \frac{(z_b^p)^2}{\sqrt{\Delta(\mathbf{p}_b^p)^2+(z_b^p)^2}}\right)\right)}{\Delta(\mathbf{p}_b^p)\left(\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2\right)} \\
& + \frac{y_b^p\left(y_b^p\,G_y(\cdot) + z_b^p\,G_z(\cdot)\right)}{\Delta(\mathbf{p}_b^p)\left(\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2\right)} \\
& - \frac{X\left(u_{d,i}+\tilde{u}_i, u_c\right) r + Y\left(u_{d,i}+\tilde{u}_i, u_c\right)(v_i - v_c)}{\sqrt{u_{d,i}^2 + w_i^2}}\Bigg] \cos\left(\theta_{d,i}\right).
\end{aligned}
\tag{B.27b}
$$

B.3 Proof of Lemma 4.1

In [71], it is shown that the error states (4.37a)–(4.37e) are UGES and the ocean current estimate errors (4.38a)–(4.38c) are bounded, which implies that (4.37a)–(4.38c) are forward complete. Therefore, we only need to prove that the underactuated sway and heave dynamics (4.39), (4.40) and the barycenter dynamics (4.36a)–(4.36c) are forward complete.

First, let us consider the underactuated sway dynamics. From (4.39), we get

$$
\dot{v}_i = X_v\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(\tilde{r}_i + r_{d,i}\right) + Y_v\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(v_i - v_c\right), \tag{B.28}
$$

where $\tilde{r}_i = r_i - r_{d,i}$. Now, let us consider a Lyapunov function candidate

$$
V_v(v_i) = \frac{1}{2}v_i^2. \tag{B.29}
$$

Its derivative along the trajectories of v_i is

$$
\dot{V}_v(v_i) = X_v(\tilde{u}_i + u_{d,i}, u_c)\left(\tilde{r}_i + r_{d,i}\right)v_i + Y_v(\tilde{u}_i + u_{d,i}, u_c)\left(v_i - v_c\right)v_i. \tag{B.30}
$$

From the boudedness of $\tilde{\mathbf{X}}_{2,i}$, $\kappa(\xi)$, $\iota(\xi)$, $u_{d,i}$, u_c and v_c, we can conclude that there exists some scalar $\beta_{v,0} > 0$ such that

$$
\left\| \left[\tilde{\mathbf{X}}_{2,i}^{\mathrm{T}}, \kappa(\xi), \iota(\xi), u_{d,i}, u_c, v_c\right]^{\mathrm{T}} \right\| \leq \beta_0. \tag{B.31}
$$

Moreover, from (B.27), we can conclude that there exist some positive functions $a_r(\beta_{v,0})$ and $b_r(\beta_{v,0})$ such that

$$
|r_{d,i}| \leq a_r(\beta_{v,0})\,|v_i| + b_r(\beta_{v,0}). \tag{B.32}
$$

Consequently, we can upper bound $\dot{V}_v(v_i)$ using the following expression

$$\begin{aligned}\dot{V}_v(v_i) \leq X_v\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(\tilde{r}_i\, v_i + a_r(\cdot) v_i^2 + b_r(\cdot) v_i\right) \\ + Y_v\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(v_i^2 - v_c\, v_i\right).\end{aligned} \tag{B.33}$$

Using Young's inequality, we get

$$\begin{aligned}\dot{V}_v(v_i) &\leq \left(X_v\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(2 + a_r(\cdot)\right) + 2\, Y_v\left(\tilde{u}_i + u_{d,i}, u_c\right)\right) v_i^2 \\ &\quad + X_v\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(\tilde{r}_i^2 + b_r(\cdot)^2\right) + Y_v\left(\tilde{u}_i + u_{d,i}, u_c\right) v_c^2 \qquad \text{(B.34a)} \\ &\leq \alpha_v\, V_v(v_i) + \beta_v. \qquad \text{(B.34b)}\end{aligned}$$

Using the comparison lemma, we get

$$V_v\left(v_i(t)\right) \leq \left(V_v\left(v_i(t_0)\right) + \frac{\beta_v}{\alpha_v}\right) \exp\left(\alpha_v (t - t_0)\right) - \frac{\beta_v}{\alpha_v}. \tag{B.35}$$

As $V_v(v_i)$ is defined for all $t > t_0$, it follows that v_i is also defined for all $t > t_0$. The solutions of (4.39) thus fulfill the definition of forward completeness, as defined in [110].

Now, let us consider the underactuated heave dynamics. From (4.40), we get

$$\dot{w}_i = X_w(\tilde{u}_i + u_{d,i}, u_c)\left(\tilde{q}_i + q_{d,i}\right) + Y_w(\tilde{u}_i + u_{d,i}, u_c)\left(w_i - w_c\right) + G(\theta_i), \tag{B.36}$$

where $\tilde{q}_i = q_i - q_{d,i}$. Similar to the previous paragraph, we consider a Lyapunov function candidate

$$V_w(w_i) = \frac{1}{2} w_i^2, \tag{B.37}$$

whose derivative is

$$\begin{aligned}\dot{V}_w(w_i) = X_w\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(\tilde{q}_i + q_{d,i}\right) w_i \\ + Y_w\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(w_i - w_c\right) w_i + G(\theta)\, w_i.\end{aligned} \tag{B.38}$$

From the boudedness of $\tilde{\mathbf{X}}_{2,i}$, $\kappa(\xi)$, $\iota(\xi)$, $u_{d,i}$, u_c and w_c, we can conclude that there exists some scalar $\beta_0 > 0$ such that

$$\left\| \left[\tilde{\mathbf{X}}_{2,i}^{\mathrm{T}}, \kappa(\xi), \iota(\xi), u_{d,i}, u_c, w_c\right]^{\mathrm{T}} \right\| \leq \beta_{w,0}. \tag{B.39}$$

Moreover, from (B.25), we can conclude that there exist some positive functions $a_q(\beta_{w,0})$ and $b_q(\beta_{w,0})$ such that

$$|q_{d,i}| \leq a_q(\beta_{w,0})\ |w_i| + b_q(\beta_{w,0}). \tag{B.40}$$

Consequently, we can upper bound $\dot{V}_w(w_i)$ using the following expression

$$\begin{aligned}\dot{V}_w(w_i) \leq X_w\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(\tilde{q}_i\, w_i + a_q(\cdot) w_i^2 + b_q(\cdot) w_i\right) \\ + Y_w\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(w_i^2 - w_c\, w_i\right) + G(\theta_i)\, w_i.\end{aligned} \tag{B.41}$$

Using Young's inequality, we get

$$\begin{aligned}\dot{V}_w(w_i) &\leq \left(X_w\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(2 + a_q(\cdot)\right) + 2\,Y_w\left(\tilde{u}_i + u_{d,i}, u_c\right) + 1\right)\, w_i^2 \\ &\quad + X_w\left(\tilde{u}_i + u_{d,i}, u_c\right)\left(\tilde{q}_i^2 + b_q(\cdot)^2\right) + Y_w\left(\tilde{u}_i + u_{d,i}, u_c\right)\, w_c^2 + G(\theta)^2 \\ &\leq \alpha_w\, V_w(w_i) + \beta_w.\end{aligned} \tag{B.42}$$

Using the comparison lemma, we get

$$V_w\left(w_i(t)\right) \leq \left(V_w\left(w_i(t_0)\right) + \frac{\beta_w}{\alpha_w}\right) \exp\left(\alpha_w(t - t_0)\right) - \frac{\beta_w}{\alpha_w}. \tag{B.43}$$

Using the same arguments as in the previous paragraph, we conclude that the solutions of (4.40) are forward complete.

Finally, let us consider the barycenter dynamics. We use a Lyapunov function candidate

$$V_b(\mathbf{p}_b^p) = \frac{1}{2}\left(\left(x_b^p\right)^2 + \left(y_b^p\right)^2 + \left(z_b^p\right)^2\right), \tag{B.44}$$

whose derivative along the solutions of (4.36a)–(4.36c) is

$$\begin{aligned}\dot{V}_b\left(\mathbf{p}_b^p\right) &= -k_\xi \frac{\left(x_b^p\right)^2}{\sqrt{1 + \left(x_b^p\right)^2}} + G_y(\cdot)\, y_b^p + G_z(\cdot)\, z_b^p \\ &\quad - \frac{1}{n}\sum_{i=1}^{n} U_{d,i}\left(\frac{\cos\left(\gamma_{\mathrm{LOS}}\right)^2 \left(y_b^p\right)^2}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(y_b^p\right)^2}} + \frac{\left(z_b^p\right)^2}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(z_b^p\right)^2}}\right) \\ &\leq G_y(\cdot)\, y_b^p + G_z(\cdot)\, z_b^p + \frac{1}{2}\left(x_b^p\right)^2.\end{aligned} \tag{B.45}$$

Using Young's inequality, we get

$$\begin{aligned}\dot{V}_b\left(\mathbf{p}_b^p\right) &\leq \frac{1}{2}\left(\left(x_b^p\right)^2 + \left(y_b^p\right)^2 + \left(z_b^p\right)^2\right) + \frac{1}{2}\left(G_y(\cdot)^2 + G_z(\cdot)^2\right) \\ &\leq V_b\left(\mathbf{p}_b^p\right) + \frac{1}{2}\left(G_y(\cdot)^2 + G_z(\cdot)^2\right).\end{aligned} \tag{B.46}$$

Note that from (B.12) and (B.22), we can conclude that there exist some positive function $\zeta_y(U_{d,1}, \ldots, U_{d,n})$ and $\zeta_z(U_{d,1}, \ldots, U_{d,n})$ such that

$$|G_y(\cdot)| \leq \zeta_y(\cdot)\left\|\left[\tilde{u}_1, \ldots, \tilde{u}_n, \tilde{\psi}_1, \ldots, \tilde{\psi}_n\right]^{\mathrm{T}}\right\|, \tag{B.47}$$

$$|G_z(\cdot)| \leq \zeta_z(\cdot)\left\|\left[\tilde{u}_1, \ldots, \tilde{u}_n, \tilde{\theta}_1, \ldots, \tilde{\theta}_n\right]^{\mathrm{T}}\right\|. \tag{B.48}$$

Consequently, there exists a class-$\mathcal{K}_\infty$ function $\zeta_p(\cdot)$ such that

$$\begin{aligned}\dot{V}_p\left(\mathbf{p}_b^p\right) \leq V_p\left(\mathbf{p}_b^p\right) + \zeta_p\Big(&v_1, \ldots, v_n, w_1, \ldots, w_n, \tilde{u}_1, \ldots, \tilde{u}_n, \\ &\tilde{\psi}_1, \ldots, \tilde{\psi}_n, \tilde{\theta}_1, \ldots, \tilde{\theta}_n\Big).\end{aligned} \tag{B.49}$$

Since all the arguments of $\zeta_p(\cdot)$ are forward complete, Corollary 2.11 of [110] is satisfied and the barycenter dynamics is forward complete, thus concluding the proof of Lemma 4.1.

B.4 Proof of Lemma 4.2

First, we consider the sway dynamics. We take the Lyapunov function candidate V_v from (B.29) and simplify its derivative by setting $\left[\tilde{\mathbf{X}}_1^{\mathrm{T}}, \tilde{\mathbf{X}}_2^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$.

$$\dot{V}_v(v_i) = X_v\left(u_{d,i}, u_c\right) r_{d,i}\, v_i + Y_v\left(u_{d,i}, u_c\right)\left(v_i - v_c\right) v_i. \tag{B.50}$$

Next, we find an upper bound on $r_{d,i}\, v_i$. We substitute from (B.27), set $\left[\tilde{\mathbf{X}}_1^{\mathrm{T}}, \tilde{\mathbf{X}}_2^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$ and collect all terms that grow linearly with v_i to obtain the following expression

$$\begin{aligned} r_{d,i}\, v_i = &\left(v_i\left(1+\tfrac{\Delta(\mathbf{p}_b^p)\, x_b^p}{\Delta(\mathbf{p}_b^p)^2+\left(x_b^p\right)^2}\right)\iota(s)\tfrac{1}{n}\sum_{j=1}^{n} U_j \Omega_x(\gamma_j, \theta_p, \chi_j, \psi_p) + \tfrac{Y_v(u_{d,i}, u_c)}{\sqrt{u_{d,i}^2+w_i^2}} v_i^2\right)\cos(\theta_{d,i}) \\ &+ F_v(u_{d,i}, u_c, v_c, v_i, w_i, r_i, \theta_{d,i}), \end{aligned} \tag{B.51}$$

where

$$F_v(\cdot) = \tfrac{X_v(u_{d,i}, u_c)\, r_i - Y_v(u_{d,i}, u_c)\, v_c}{\sqrt{u_{d,i}^2+w_i^2}} v_i\, \cos(\theta_{d,i}). \tag{B.52}$$

We can bound this expression as

$$\begin{aligned} |r_{d,i}\, v_i| &\leq \frac{2}{n}\,|v_i|\;|\iota(\xi)| \sum_{j=1}^{n}\left(|u_j| + |v_j| + |w_j|\right) + |F_v(\cdot)| \\ &\leq \frac{2}{n}\,|\iota(\xi)|\; v_i^2 + \frac{2}{n}\,|v_i|\;|\iota(\xi)| \left(\sum_{j\neq i}\left(|u_j| + |v_j| + |w_j|\right) + |u_i| + |w_i|\right) \\ &\quad + \left|F_v(u_{d,i}, u_c, v_c, v_i, w_i, r_i, \theta_{d,i})\right|, \end{aligned} \tag{B.53}$$

which we can substitute to (B.50) to obtain

$$\begin{aligned} \dot{V}_v(v_i) \leq &\left(X_v\left(u_{d,i}, u_c\right)\frac{2}{n}\,|\iota(\xi)| + Y_v\left(u_{d,i}, u_c\right)\right) v_i^2 \\ &+ \left(\frac{2}{n}\,|v_i|\;|\iota(\xi)| \sum_{j\neq i}\left(|u_j| + |v_j| + |w_j|\right) + |u_i| + |w_i|\right) \\ &+ \left(|F_v(\cdot)| - Y_v\left(u_{d,i}, u_c\right)|v_c|\right)|v_i|\,. \end{aligned} \tag{B.54}$$

For a sufficiently large v_i, the quadratic term will dominate the linear term. Therefore, we can conclude that v_i is bounded if

$$X_v\left(u_{d,i}, u_c\right)\frac{2}{n}\,|\iota(\xi)| + Y_v\left(u_{d,i}, u_c\right) < 0. \tag{B.55}$$

Since Y_v is assumed to be always negative, the inequality is satisfied if

$$|\iota(\xi)| < \frac{n}{2}\left|\frac{Y_v\left(u_{d,i}, u_c\right)}{X_v\left(u_{d,i}, u_c\right)}\right|. \tag{B.56}$$

Now, we perform a similar procedure for the heave dynamics. We take the Lyapunov function candidate V_w from (B.37) and simplify its derivative by setting $\left[\tilde{\mathbf{X}}_1^{\mathrm{T}}, \tilde{\mathbf{X}}_2^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$.

$$\dot{V}_w(w_i) = X_w\left(u_{d,i}, u_c\right)\, q_{d,i}\, w_i + Y_w\left(u_{d,i}, u_c\right)\, \left(w_i - w_c\right)\, w_i + G(\theta_i)\, w_i. \tag{B.57}$$

Next, we find an upper bound on $q_{d,i}\, w_i$. We substitute from (B.25), set $\left[\tilde{\mathbf{X}}_1^{\mathrm{T}}, \tilde{\mathbf{X}}_2^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$ and collect all terms that grow linearly with w_i to obtain the following expression

$$q_{d,i}\, w_i = w_i\left(1 + \frac{\Delta(\mathbf{p}_b^p)\, x_b^p}{\Delta(\mathbf{p}_b^p)^2 + \left(x_b^p\right)^2}\right)\kappa(\xi)\frac{1}{n}\sum_{j=1}^{n} U_j\, \Omega_x(\gamma_j, \theta_p, \chi_j, \psi_p) \tag{B.58}$$

$$+ u_{d,i}\frac{Y_w(u_{d,i}, u_c)}{u_{d,i}^2 + w_i^2} w_i^2 + F_w(u_{d,i}, u_c, w_c, w_i, q_i), \tag{B.59}$$

where

$$F_w(\cdot) = u_{d,i}\frac{X_w(u_{d,i}, u_c)\, r_i - Y_w(u_{d,i}, u_c)\, w_c}{\sqrt{u_{d,i}^2 + w_i^2}} w_i. \tag{B.60}$$

We can bound this expression as

$$\begin{aligned} |q_{d,i}\, w_i| \leq \frac{2}{n}\left|\kappa(\xi)\right|\, w_i^2 + \frac{2}{n}\left|w_i\right|\, \left|\kappa(\xi)\right| \left(\sum_{j\neq i}\left(|u_j| + |v_j| + |w_j|\right) + |u_i| + |v_i|\right) \\ + \left|F_w(u_{d,i}, u_c, w_c, w_i, q_i)\right|, \end{aligned} \tag{B.61}$$

which we can substitute to (B.57) to obtain

$$\begin{aligned} \dot{V}_w(w_i) \leq \left(X_w\left(u_{d,i}, u_c\right)\frac{2}{n}\left|\kappa(\xi)\right| + Y_w\left(u_{d,i}, u_c\right)\right) w_i^2 \\ + \left(\frac{2}{n}\left|w_i\right|\, \left|\kappa(\xi)\right| \sum_{j\neq i}\left(|u_j| + |v_j| + |w_j|\right) + |u_i| + |w_i|\right) \\ + \left(\left|F(\cdot)\right| - Y_w\left(u_{d,i}, u_c\right)|v_c| + |G(\theta_i)|\right)|w_i| + G(\theta_i)\, w_i. \end{aligned} \tag{B.62}$$

For a sufficiently large w_i, the quadratic term will dominate the linear term. Therefore, we can conclude that w_i is bounded if

$$X_w\left(u_{d,i}, u_c\right)\frac{2}{n}\left|\kappa(\xi)\right| + Y_w\left(u_{d,i}, u_c\right) < 0. \tag{B.63}$$

Since Y_w is assumed to be always negative, the inequality is satisfied if

$$|\kappa(\xi)| < \frac{n}{2}\left|\frac{Y_w\left(u_{d,i}, u_c\right)}{X_w\left(u_{d,i}, u_c\right)}\right|, \tag{B.64}$$

which concludes the proof of Lemma 4.2.

B.5 Proof of Lemma 4.3

First, we consider the sway dynamics. We take the Lyapunov function candidate V_v from (B.29) and simplify its derivative by setting $\tilde{\mathbf{X}}_2 = \mathbf{0}$.

$$\dot{V}_v(v_i) = X_v\,(u_{d,i}, u_c)\; r_{d,i}\, v_i + Y_v\,(u_{d,i}, u_c)\; (v_i - v_c)\; v_i. \tag{B.65}$$

Next, we find an upper bound on $r_{d,i}\, v_i$. We substitute from (B.27), set $\tilde{\mathbf{X}}_2 = \mathbf{0}$ and collect all terms that grow linearly with v_i to obtain the following expression

$$\begin{aligned} r_{d,i}\, v_i = \Bigg(& v_i \left(1 + \frac{\Delta(\mathbf{p}_b^p)\, x_b^p}{\Delta(\mathbf{p}_b^p)^2 + \left(x_b^p\right)^2}\right) \iota(\xi) \tfrac{1}{n} \sum_{j=1}^{n} U_j\, \Omega_x(\gamma_j, \theta_p, \chi_j, \psi_p) \\ & - \frac{y_b^p\, v_i \sum_{j=1}^{n} \left(\frac{\cos(\gamma_{\mathrm{LOS}}) y_b^p}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(y_b^p\right)^2}} + \frac{z_b^p}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(z_b^p\right)^2}} \right)}{n\, \Delta(\mathbf{p}_b^p) \left(\Delta(\mathbf{p}_b^p)^2 + \left(y_b^p\right)^2 \right)} \\ & + \frac{v_i\, \Delta(\mathbf{p}_b^p) \sum_{j=1}^{n} \frac{\cos(\gamma_{\mathrm{LOS}}) y_b^p}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(y_b^p\right)^2}}}{n\, \left(\Delta(\mathbf{p}_b^p)^2 + (y_b^p)^2\right)} + \frac{Y_v(u_{d,i}, u_c)}{\sqrt{u_{d,i}^2 + w_i^2}} v_i^2 \Bigg) \cos(\theta_{d,i}) \\ & + H_v(u_{d,i}, \theta_{d,i}, u_c, v_c, v_i, w_i, r_i, \mathbf{p}_b^p, \xi), \end{aligned} \tag{B.66}$$

$$\begin{aligned} H_v(\cdot) = \Bigg(& \left(1 + \frac{\Delta(\mathbf{p}_b^p)\, x_b^p}{\Delta(\mathbf{p}_b^p)^2 + \left(x_b^p\right)^2}\right) k_\xi\, \iota(\xi) \frac{x_b^p}{\sqrt{1 + \left(x_b^p\right)^2}} \\ & + \frac{X_v(u_{d,i}, u_c)\, r_i - Y_v(u_{d,i}, u_c)\, v_c}{\sqrt{u_{d,i}^2 + w_i^2}} \\ & - \frac{y_b^p\, k_\xi\, x_b^p}{\sqrt{1 + \left(x_b^p\right)^2}\, \Delta(\mathbf{p}_b^p) \left(\Delta(\mathbf{p}_b^p)^2 + \left(y_b^p\right)^2 \right)} \Bigg) v_i\, \cos(\theta_{d,i}). \end{aligned} \tag{B.67}$$

We can bound this expression as

$$\begin{aligned} |r_{d,i}\, v_i| &\leq \left(\frac{2}{n} |\iota(\xi)| + \frac{3}{n\, \Delta(\mathbf{p}_b^p)} \right) |v_i| \sum_{j=1}^{n} \left(|u_j| + |v_j| + |w_j| \right) + |H_v(\cdot)| \\ &\leq \left(\frac{2}{n} |\iota(\xi)| + \frac{3}{n\, \Delta(\mathbf{p}_b^p)} \right) v_i^2 + |H_v(\cdot)| \\ &\quad + \left(\frac{2}{n} |\iota(\xi)| + \frac{3}{n\, \Delta(\mathbf{p}_b^p)} \right) \left(\sum_{j \neq i} \left(|u_j| + |v_j| + |w_j| \right) + |u_i| + |w_i| \right), \end{aligned} \tag{B.68}$$

which we can substitute to (B.65) to obtain

$$\begin{aligned}\dot{V}_v(v_i) \leq & \left(X_v\left(u_{d,i}, u_c\right)\left(\frac{2}{n}\left|\iota(\xi)\right| + \frac{3}{n\,\Delta(\mathbf{p}_b^p)}\right) + Y_v\left(u_{d,i}, u_c\right)\right) v_i^2 \\ & + \left(\frac{2}{n}\left|\iota(\xi)\right| + \frac{3}{n\,\Delta(\mathbf{p}_b^p)}\right)\left(\sum_{j\neq i}\left(|u_j| + |v_j| + |w_j|\right) + |u_i| + |w_i|\right) \\ & + \left(\left|H_v(\cdot)\right| - Y_v\left(u_{d,i}, u_c\right)|v_c|\right)|v_i|\,.\end{aligned} \tag{B.69}$$

For a sufficiently large v_i, the quadratic term will dominate the linear term. Therefore, we can conclude that v_i is bounded if

$$X_v\left(u_{d,i}, u_c\right)\left(\frac{2}{n}\left|\iota(\xi)\right| + \frac{3}{n\,\Delta(\mathbf{p}_b^p)}\right) + Y_v\left(u_{d,i}, u_c\right) < 0. \tag{B.70}$$

From the definition of the lookahead distance (4.20), this condition is satisfied if

$$\Delta_0 > \frac{3}{n\left|\frac{Y_v(u_{d,i},u_c)}{X_v(u_{d,i},u_c)}\right| - 2\left|\iota(\xi)\right|}. \tag{B.71}$$

Now, we perform a similar procedure for the heave dynamics. We take the Lyapunov function candidate V_w from (B.37) and simplify its derivative by setting $\tilde{\mathbf{X}}_2 = \mathbf{0}$.

$$\dot{V}_w(w_i) = X_w\left(u_{d,i}, u_c\right)\, q_{d,i}\, w_i + Y_w\left(u_{d,i}, u_c\right)\,\left(w_i - w_c\right)\, w_i + G(\theta_i)\, w_i. \tag{B.72}$$

Next, we find an upper bound on $q_{d,i}\, w_i$. We substitute from (B.25), set $\tilde{\mathbf{X}}_2 = \mathbf{0}$ and collect all terms that grow linearly with w_i to obtain the following expression

$$\begin{aligned}q_{d,i}\, w_i = & \; w_i\left(1 + \frac{\Delta(\mathbf{p}_b^p)\, x_b^p}{\Delta(\mathbf{p}_b^p)^2 + \left(x_b^p\right)^2}\right)\kappa(\xi)\,\tfrac{1}{n}\textstyle\sum_{j=1}^n U_j\,\Omega_x(\gamma_j, \theta_p, \chi_j, \psi_p) \\ & - \frac{z_b^p\, w_i\,\sum_{j=1}^n\left(\frac{\cos(\gamma_{\mathrm{LOS}})\, y_b^p}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(y_b^p\right)^2}} + \frac{z_b^p}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(z_b^p\right)^2}}\right)}{n\,\Delta(\mathbf{p}_b^p)\left(\Delta(\mathbf{p}_b^p)^2 + \left(z_b^p\right)^2\right)} \\ & + \frac{w_i\,\Delta(\mathbf{p}_b^p)\sum_{j=1}^n \frac{z_b^p}{\sqrt{\Delta\left(\mathbf{p}_b^p\right)^2 + \left(z_b^p\right)^2}}}{n\,\left(\Delta(\mathbf{p}_b^p)^2 + (z_b^p)^2\right)} + u_{d,i}\,\frac{Y_w(u_{d,i}, u_c)}{u_{d,i}^2 + w_i^2} w_i^2 \\ & + H_w(u_{d,i}, u_c, v_c, w_i, v_i, q_i, \mathbf{p}_b^p, \xi),\end{aligned} \tag{B.73}$$

where

$$\begin{aligned}H_w(\cdot) = & \left(\left(1 + \frac{\Delta(\mathbf{p}_b^p)\, x_b^p}{\Delta(\mathbf{p}_b^p)^2 + \left(x_b^p\right)^2}\right) k_\xi\,\kappa(\xi)\frac{x_b^p}{\sqrt{1 + \left(x_b^p\right)^2}}\right. \\ & - \frac{y_b^p\, k_\xi\, x_b^p}{\sqrt{1 + \left(x_b^p\right)^2}\,\Delta(\mathbf{p}_b^p)\left(\Delta(\mathbf{p}_b^p)^2 + \left(y_b^p\right)^2\right)} \\ & \left. + u_{d,i}\,\frac{X_w(u_{d,i}, u_c)\, r_i - Y_w(u_{d,i}, u_c)\, v_c}{u_{d,i}^2 + w_i^2}\right) w_i.\end{aligned} \tag{B.74}$$

We can bound this expression as

$$\begin{aligned}|q_{d,i}\, w_i| &\leq \left(\frac{2}{n}\,|\kappa(\xi)| + \frac{3}{n\,\Delta(\mathbf{p}_b^p)}\right)|w_i| \sum_{j=1}^{n}\left(|u_j| + |v_j| + |w_j|\right) + |H_w(\cdot)| \\ &\leq \left(\frac{2}{n}\,|\kappa(\xi)| + \frac{3}{n\,\Delta(\mathbf{p}_b^p)}\right) w_i^2 + |H_w(\cdot)| \\ &\quad + \left(\frac{2}{n}\,|\kappa(\xi)| + \frac{3}{n\,\Delta(\mathbf{p}_b^p)}\right)\left(\sum_{j\neq i}\big(|u_j| + |v_j| + |w_j|\big) + |u_i| + |w_i|\right),\end{aligned} \tag{B.75}$$

which we can substitute to (B.72) to obtain

$$\begin{aligned}\dot{V}_w(w_i) \leq &\left(X_w\left(u_{d,i}, u_c\right)\left(\frac{2}{n}\,|\kappa(\xi)| + \frac{3}{n\,\Delta(\mathbf{p}_b^p)}\right) + Y_w\left(u_{d,i}, u_c\right)\right) w_i^2 \\ &+ \left(\frac{2}{n}\,|\kappa(\xi)| + \frac{3}{n\,\Delta(\mathbf{p}_b^p)}\right)\left(\sum_{j\neq i}\big(|u_j| + |v_j| + |w_j|\big) + |u_i| + |w_i|\right) \\ &+ \left(|H_w(\cdot)| - Y_w\left(u_{d,i}, u_c\right)|v_c|\right)|w_i|\,.\end{aligned} \tag{B.76}$$

For a sufficiently large w_i, the quadratic term will dominate the linear term. Therefore, we can conclude that w_i is bounded if

$$X_w\left(u_{d,i}, u_c\right)\left(\frac{2}{n}\,|\kappa(\xi)| + \frac{3}{n\,\Delta(\mathbf{p}_b^p)}\right) + Y_w\left(u_{d,i}, u_c\right) < 0. \tag{B.77}$$

From the definition of the lookahead distance (4.20), this condition is satisfied if

$$\Delta_0 > \frac{3}{n\left|\frac{Y_w(u_{d,i},u_c)}{X_w(u_{d,i},u_c)}\right| - 2\,|\kappa(\xi)|}. \tag{B.78}$$

Appendix C

Derivations from Chapter 5

C.1 Bounds on the NSB Velocity

Recall the definition of $\boldsymbol{\omega}_{\mathbf{v}_{\mathrm{NSB},i}}$ in (5.41). Note that by definition, a normalized vector is always orthogonal to its derivative. Therefore, the following equality holds:

$$\left\|\boldsymbol{\omega}_{\mathbf{v}_{\mathrm{NSB},i}}\right\| = \left\|\overline{\mathbf{v}}_{\mathrm{NSB},i}\right\| \left\|\dot{\overline{\mathbf{v}}}_{\mathrm{NSB},i}\right\| = \left\|\dot{\overline{\mathbf{v}}}_{\mathrm{NSB},i}\right\|. \tag{C.1}$$

Therefore, instead of the pseudo-angular velocity, it is possible to investigate the derivative of the normalized NSB velocity. Note that according to the assumptions in Theorem 5.1, the analysis should be performed on the manifold $\left[\tilde{\boldsymbol{\sigma}}^{\mathrm{T}}, \tilde{\mathbf{X}}^{\mathrm{T}}\right] = \mathbf{0}^{\mathrm{T}}$. Substituting $\tilde{\boldsymbol{\sigma}} = \mathbf{0}$ to (5.30) yields

$$\begin{aligned} \mathbf{v}_{\mathrm{NSB},i} &= \mathbf{v}_{\mathrm{LOS}} + \dot{\mathbf{R}}_p(\xi)\mathbf{p}^f_{f,i} \\ &= U_{\mathrm{LOS}}\mathbf{R}_p(\xi)\left(\mathbf{e}_1 + \left\|\partial \mathbf{p}_p(\xi)/\partial\xi\right\|^{-1}\boldsymbol{\omega}_p(\xi) \times \mathbf{p}^f_{f,i}\right). \end{aligned} \tag{C.2}$$

For brevity, let us define

$$\boldsymbol{\kappa} = \left\|\partial \mathbf{p}_p(\xi)/\partial\xi\right\|^{-1}\boldsymbol{\omega}_p(\xi), \qquad \mathbf{e}_p = \mathbf{e}_1 + \boldsymbol{\kappa} \times \mathbf{p}^f_{f,i} \tag{C.3}$$

The normalized NSB velocity is then given by

$$\overline{\mathbf{v}}_{\mathrm{NSB},i} = \frac{\mathbf{R}_p(\xi)\mathbf{e}_p}{\|\mathbf{e}_p\|}. \tag{C.4}$$

Differentiating (C.4) with respect to time yields

$$\dot{\overline{\mathbf{v}}}_{\mathrm{NSB},i} = \frac{U_{\mathrm{LOS}}\mathbf{R}_p\left(\boldsymbol{\kappa} \times \mathbf{e}_p + \boldsymbol{\iota} \times \mathbf{p}^f_{f,i}\right)}{\|\mathbf{e}_p\|} - \frac{U_{\mathrm{LOS}}\mathbf{R}_p\mathbf{e}_p\left(\mathbf{e}_p^{\mathrm{T}}\left(\boldsymbol{\iota} \times \mathbf{p}^f_{f,i}\right)\right)}{\|\mathbf{e}_p\|^2}, \tag{C.5}$$

where $\boldsymbol{\iota} = \partial\boldsymbol{\kappa}/\partial\xi$. From (C.5), it follows that

$$\left\|\dot{\overline{\mathbf{v}}}_{\mathrm{NSB},i}\right\| \leq U_{\mathrm{LOS}}\left(\|\boldsymbol{\kappa}\| + \frac{\left\|\boldsymbol{\iota} \times \mathbf{p}^f_{f,i}\right\|\left(1 + \|\mathbf{e}_p\|\right)}{\|\mathbf{e}_p\|}\right). \tag{C.6}$$

If we assume that the second and third partial derivatives of $\mathbf{p}_p$ with respect to the path parameter are bounded, then $\boldsymbol{\iota}$ is bounded as well. Let us define

$$c_{\mathrm{NSB}} = \max_{i,\xi} \left(\|\boldsymbol{\kappa}\| + \frac{\left\| \boldsymbol{\iota} \times \mathbf{p}_{f,i}^{f} \right\| (1 + \|\mathbf{e}_p\|)}{\|\mathbf{e}_p\|} \right). \tag{C.7}$$

Substituting (5.37) and (C.7) into (C.6) gives us the following upper bound

$$\left\| \dot{\bar{\mathbf{v}}}_{\mathrm{NSB},i} \right\| \leq \frac{v_{2,\max} + \sqrt{\sum_{i=1}^{n} (v_i^2 + w_i^2) + u_{\min}^2}}{1 - k_{\mathrm{NSB}}} \, c_{\mathrm{NSB}}. \tag{C.8}$$

Note that for any two positive numbers a and b, the following inequality holds: $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$. Therefore, we can further upper-bound (C.8) with

$$\left\| \dot{\bar{\mathbf{v}}}_{\mathrm{NSB},i} \right\| \leq \underbrace{\frac{c_{\mathrm{NSB}}}{1 - k_{\mathrm{NSB}}}}_{a_{\mathrm{NSB}}} \|\boldsymbol{\upsilon}_u\| + \underbrace{\frac{v_{2,\max} + u_{\min}}{1 - k_{\mathrm{NSB}}} \, c_{\mathrm{NSB}}}_{b_{\mathrm{NSB}}}. \tag{C.9}$$

We have thus shown that there exist positive constants a_{NSB} and b_{NSB} that satisfy (5.72).

C.2 Bounds on the Linear Velocity

Note that by the assumptions of Theorem 5.1, the surge velocity of the vehicle satisfies $u_i = u_{d,i}$, and the linear velocity vector $\boldsymbol{\upsilon}_i$ thus satisfies

$$\boldsymbol{\upsilon}_i = [u_{d,i}, v_i, w_i]^{\mathrm{T}} = \left[\sqrt{\|\mathbf{v}_{\mathrm{NSB},i}\|^2 - v_i^2 - w_i^2}, v_i, w_i \right]^{\mathrm{T}}, \qquad \|\boldsymbol{\upsilon}_i\| = \|\mathbf{v}_{\mathrm{NSB},i}\|. \tag{C.10}$$

The time-derivative of a normalized vector is given by

$$\dot{\overline{\boldsymbol{\upsilon}}}_i = \frac{\dot{\boldsymbol{\upsilon}}_i}{\|\boldsymbol{\upsilon}_i\|} - \frac{\boldsymbol{\upsilon}_i \, \frac{\mathrm{d}}{\mathrm{d}t}\|\boldsymbol{\upsilon}_i\|}{\|\boldsymbol{\upsilon}_i\|^2}, \tag{C.11}$$

and the pseudo-angular velocity is thus given by

$$\boldsymbol{\omega}_{\boldsymbol{\upsilon}_i} = \overline{\boldsymbol{\upsilon}}_i \times \dot{\overline{\boldsymbol{\upsilon}}}_i = \frac{\boldsymbol{\upsilon}_i}{\|\boldsymbol{\upsilon}_i\|} \times \left(\frac{\dot{\boldsymbol{\upsilon}}_i}{\|\boldsymbol{\upsilon}_i\|} - \frac{\boldsymbol{\upsilon}_i \, \frac{\mathrm{d}}{\mathrm{d}t}\|\boldsymbol{\upsilon}_i\|}{\|\boldsymbol{\upsilon}_i\|^2} \right) = \frac{\boldsymbol{\upsilon}_i \times \dot{\boldsymbol{\upsilon}}_i}{\|\boldsymbol{\upsilon}_i\|^2}. \tag{C.12}$$

Now, let us focus on $\dot{\boldsymbol{\upsilon}}_i$. Differentiating (C.10) with respect to time yields

$$\dot{\boldsymbol{\upsilon}}_i = \begin{bmatrix} \frac{\mathbf{v}_{\mathrm{NSB},i}^{\mathrm{T}} \dot{\mathbf{v}}_{\mathrm{NSB},i} - v_i \dot{v}_i - w_i \dot{w}_i}{u_i} \\ \dot{v}_i \\ \dot{w}_i \end{bmatrix}. \tag{C.13}$$

From (5.4b), the underactuated dynamics are given by

$$\dot{v}_i = (X_{v0} + X_{v1}(u_i - u_c))\, r_i + (Y_{v0} + Y_{v1}(u_i - u_c))\, (v_i - v_c)$$

$$+ (Z_{v0} + Z_{v1} p_i)(w_i - w_c) + w_c p_i - u_c r_i, \tag{C.14a}$$

$$\dot{w}_i = (X_{w0} + X_{w1}(u_i - u_c))\, q_i + (Y_{w0} + Y_{w1}(u_i - u_c))(w_i - w_c) + (Z_{w0} + Z_{w1} p_i)(v_i - v_c) + u_c q_i - v_c p_i, \tag{C.14b}$$

where

$$X_v(u_r) = X_{v0} + X_{v1} u_r, \qquad X_w(u_r) = X_{w0} + X_{w1} u_r, \tag{C.15a}$$

$$Y_v(u_r) = Y_{v0} + Y_{v1} u_r, \qquad Y_w(u_r) = Y_{w0} + Y_{w1} u_r, \tag{C.15b}$$

$$Z_v(p) = Z_{v0} + Z_{v1} p, \qquad Z_w(p) = Z_{w0} + Z_{w1} p. \tag{C.15c}$$

Substituting (C.14) into (C.13) yields

$$\dot{\boldsymbol{v}}_i = \mathbf{A}_{\boldsymbol{\omega}_i} \boldsymbol{\omega}_i + \widehat{\boldsymbol{\omega}}_{0,i}, \tag{C.16}$$

where

$$\mathbf{A}_{\boldsymbol{\omega}_i} = \begin{bmatrix} \frac{w_i (v_c - Z_{w1} v_r) - v_i (w_c + Z_{v1} w_r)}{u_i} & -\frac{w_i (X_{w0} + X_{w1} u_r + u_c)}{u_i} & \frac{v_i (u_c - X_{v0} - X_{v1} u_r)}{u_i} \\ w_c + Z_{v1} w_r & 0 & X_{v0} + X_{v1} u_r - u_c \\ -v_c - Z_{w1} v_r & X_{w0} + X_{w1} u_r + u_c & 0 \end{bmatrix}$$

$$\widehat{\boldsymbol{\omega}}_{0,i} = \begin{bmatrix} \frac{\mathbf{v}_{\mathrm{NSB},i}^{\mathrm{T}} \dot{\mathbf{v}}_{\mathrm{NSB},i} - v_i((Y_{v0} + Y_{v1} u_r) v_r + Z_{v0} w_r) - w_i((Y_{w0} + Y_{w1} u_r) w_r + Z_{w0} v_r)}{u_i} \\ (Y_{v0} + Y_{v1} u_r)\, v_r + Z_{v0} w_r \\ (Y_{w0} + Y_{w1} u_r)\, w_r + Z_{w0} v_r \end{bmatrix}. \tag{C.17}$$

Substituting (C.16) into (C.12) yields

$$\boldsymbol{\omega}_{\boldsymbol{v}_i} = \frac{\boldsymbol{v}_i \times \left(\widehat{\mathbf{A}}_{\boldsymbol{\omega}_i} \boldsymbol{\omega}_i + \widehat{\boldsymbol{\omega}}_{0,i} \right)}{\|\boldsymbol{v}_i\|^2} = \underbrace{\frac{\mathbf{S}(\boldsymbol{v}_i)\, \widehat{\mathbf{A}}_{\boldsymbol{\omega}_i}}{\|\boldsymbol{v}_i\|^2}}_{\mathbf{A}_{\boldsymbol{\omega}_i}} \boldsymbol{\omega}_i + \underbrace{\frac{\boldsymbol{v}_i \times \widehat{\boldsymbol{\omega}}_{0,i}}{\|\boldsymbol{v}_i\|^2}}_{\boldsymbol{\omega}_{0,i}}. \tag{C.18}$$

We have thus shown that $\boldsymbol{\omega}_{\boldsymbol{v}_i}$ is affine in $\boldsymbol{\omega}_i$.

Now we investigate the determinant of $(\mathbf{I} + \mathbf{A}_{\boldsymbol{\omega}_i})$. From the definition of $\mathbf{A}_{\boldsymbol{\omega}_i}$ in (C.18), we get the following expression

$$\begin{aligned} \det(\mathbf{I} + \mathbf{A}_{\boldsymbol{\omega}_i}) = \Big(& u_i \left(u_i^2 + v_i^2 + w_i^2\right) - u_c \left(u_i^2 + v_i^2 + w_i^2\right) \\ & - (u_c u_i + v_c v_i + w_c w_i)(u_i - u_c) + X_{v0} \left(u_i^2 + v_i^2\right) \\ & - X_{w0} \left(u_i^2 + w_i^2\right) + (X_{v1} - X_{w1})\, u_i (u_i - u_c)^2 \\ & + (X_{v1} + Z_{w1})\, v_i^2 (u_i - u_c) - (X_{w1} + Z_{v1})\, w_i^2 (u_i - u_c) \\ & - X_{v0} X_{w0} u_i - X_{v0} (u_i u_c + v_i v_c) + X_{w0} (u_i u_c + w_i w_c) \\ & - X_{v0} \left(X_{w1} u_i^2 - Z_{w1} v_i^2\right) - X_{w0} \left(X_{v1} u_i^2 - Z_{v1} w_i^2\right) \\ & - X_{v1} X_{w1} u_i (u_i - u_c)^2 - (X_{v1} + Z_{w1})\, v_i v_c (u_i - u_c) \\ & + (X_{w1} + Z_{v1})\, w_i w_c (u_i - u_c) + X_{v1} Z_{w1} v_i^2 (u_i - u_c) \end{aligned}$$

$$
\begin{aligned}
&+ X_{w1}Z_{v1}w_i^2\left(u_i - u_c\right) + X_{v0}\left(X_{w1}u_iu_c - Z_{w1}v_iv_c\right) \\
&+ X_{w0}\left(X_{v1}u_iu_c - Z_{v1}w_iw_c\right) - X_{v1}Z_{w1}v_iv_c\left(u_i - u_c\right) \\
&- X_{w1}Z_{v1}w_iw_c\left(u_i - u_c\right)\Big)\frac{1}{u_i\left(u_i^2 + v_i^2 + w_i^2\right)}.
\end{aligned} \tag{C.19}
$$

It can then be shown that the determinant satisfies

$$
\det\left(\mathbf{I} + \mathbf{A}_{\boldsymbol{\omega}_i}\right) \leq 1 - k_a, \tag{C.20}
$$

where

$$
\begin{aligned}
k_a = &\frac{|u_c|}{u_{\min}} + 2\left|X_{v1} - X_{w1} - X_{v1}X_{w1}\right|\frac{u_{\min}^2 + u_c^2}{u_{\min}^2} + \frac{|X_{v0}| + |X_{w0}|}{u_{\min}} \\
&+ \frac{(|u_c| + |v_c| + |w_c|)(u_{\min} + |u_c|)}{u_{\min}^2} + \max\left\{|X_{v0}|, |X_{w0}|\right\}\frac{u_{\min}^2 + \|\mathbf{V}_c\|^2}{u_{\min}^3} \\
&+ \max\left\{\left|X_{v1} + Z_{w1} + X_{v1}Z_{w1}\right|, \left|X_{w1} + Z_{v1} + X_{w1}Z_{v1}\right|\right\}\frac{u_{\min} + |u_c|}{u_{\min}} \\
&+ \frac{|X_{v0}X_{w0}|}{u_{\min}^2} + \left|X_{v1} + Z_{w1} - X_{v1}Z_{w1}\right|\frac{|v_c|\left(u_{\max} + |u_c|\right)}{u_{\max}^2} \\
&+ \frac{|X_{v0}|\max\left\{\left|X_{w1}\right|, \left|Z_{w1}\right|\right\} + |X_{w0}|\max\left\{\left|X_{v1}\right|, \left|Z_{v1}\right|\right\}}{u_{\min}} \\
&+ \left|X_{w1} + Z_{v1} - X_{w1}Z_{v1}\right|\frac{|w_c|\left(u_{\max} + |u_c|\right)}{u_{\max}^2} \\
&+ \frac{|X_{v0}|\left(|X_{w1}u_c| + |Z_{w1}v_c|\right) + |X_{w0}|\left(|X_{v1}u_c| + |Z_{v1}w_c|\right)}{u_{\min}^2}
\end{aligned} \tag{C.21}
$$

Note that the components of the ocean current, $|u_c|$, $|v_c|$, and $|w_c|$, can be upper bounded by $\|\mathbf{V}_c\|$. We have therefore found a constant upper bound on the determinant.

Now, let us focus on $\boldsymbol{\omega}_{0,i}$. Recall the definition of $\boldsymbol{\omega}_{0,i}$ in (C.18). To find an upper bound, we will use the following inequality

$$
\|\boldsymbol{\upsilon}_i \times \widehat{\boldsymbol{\omega}}_{0,i}\| \leq \|\boldsymbol{\upsilon}_i\|\,\|\widehat{\boldsymbol{\omega}}_{0,i}\|, \qquad \Longrightarrow \qquad \|\boldsymbol{\omega}_{0,i}\| \leq \frac{\|\widehat{\boldsymbol{\omega}}_{0,i}\|}{\|\boldsymbol{\upsilon}_i\|}. \tag{C.22}
$$

Recall the definition of $\widehat{\boldsymbol{\omega}}_{0,i}$ in (C.16). To find an upper bound on this vector, we will utilize the following inequality: Consider a vector $\mathbf{x} = \left[\sum_{i=1}^{N_a} a_i, \sum_{i=1}^{N_b} b_i, \sum_{i=1}^{N_c} c_i\right]^{\mathrm{T}}$, where $a_i, b_i, c_i \in \mathbb{R}$. The following inequality holds for the Euclidean norm of $\mathbf{x}$

$$
\|\mathbf{x}\| \leq \sum_{i=1}^{N_a}|a_i| + \sum_{i=1}^{N_b}|b_i| + \sum_{i=1}^{N_c}|c_i|. \tag{C.23}
$$

Therefore, we can find an upper bound on $\|\widehat{\boldsymbol{\omega}}_{0,i}\|$ by analyzing its components.

Let us begin by investigating the term $\frac{\mathbf{v}_{\mathrm{NSB},i}^{\mathrm{T}}\dot{\mathbf{v}}_{\mathrm{NSB},i}}{u_i}$. From (C.2), $\mathbf{v}_{\mathrm{NSB},i}$ and its time-derivative are given by

$$
\mathbf{v}_{\mathrm{NSB},i} = U_{\mathrm{LOS}}\mathbf{R}_p(\xi)\mathbf{e}_p, \qquad \dot{\mathbf{v}}_{\mathrm{NSB},i} = U_{\mathrm{LOS}}\mathbf{R}_p(\xi)\left(\boldsymbol{\kappa} \times \mathbf{e}_p + \boldsymbol{\iota} \times \mathbf{p}_{f,i}^f\right). \tag{C.24}
$$

For brevity, let us define

$$\mathbf{e}_d = \boldsymbol{\kappa} \times \mathbf{e}_p + \boldsymbol{\iota} \times \mathbf{p}_{f,i}^{f}. \tag{C.25}$$

Then, the following inequality holds for the investigated term

$$\begin{aligned}\left|\frac{\mathbf{v}_{\mathrm{NSB},i}^{\mathrm{T}}\dot{\mathbf{v}}_{\mathrm{NSB},i}}{u_i}\right| &\leq \frac{\|\mathbf{v}_{\mathrm{NSB},i}\|\,\|\dot{\mathbf{v}}_{\mathrm{NSB},i}\|}{u_i} = \frac{\|\boldsymbol{\upsilon}_i\|\,U_{\mathrm{LOS}}\,\|\mathbf{e}_d\|}{u_i} \\ &\leq \|\boldsymbol{\upsilon}_i\|\,\frac{\|\mathbf{e}_d\|}{\|\mathbf{e}_p\|}\frac{\|\mathbf{v}_{\mathrm{NSB},i}\|}{u_i} \leq \frac{\|\mathbf{e}_d\|}{\|\mathbf{e}_p\|}\frac{\|\boldsymbol{\upsilon}_i\|^2}{u_{\min}}\end{aligned} \tag{C.26}$$

We can now expand the remaining terms in $\widehat{\boldsymbol{\omega}}_{0,i}$ to arrive at the following upper bound

$$\begin{aligned}\|\widehat{\boldsymbol{\omega}}_{0,i}\| \leq{}& \frac{\|\mathbf{e}_d\|}{\|\mathbf{e}_p\|}\frac{\|\boldsymbol{\upsilon}_i\|^2}{u_{\min}} + \left|\frac{Y_{v1}\,(u_i-u_c)+Y_{v0}}{u_i}\right| v_i^2 + \left|\frac{Y_{w1}\,(u_i-u_c)+Y_{w0}}{u_i}\right| w_i^2 \\ &+ \left|\frac{Z_{v0}+Z_{w0}}{u_i}v_i w_i\right| + \left|\frac{Y_{v1}u_c v_c - Y_{v0}v_c - Z_{v0}w_c - Y_{v1}u_i v_c}{u_i}v_i\right| \\ &+ \left|\frac{Y_{w1}u_c w_c - Y_{w0}w_c - Z_{w0}v_c - Y_{w1}u_i v_c}{u_i}w_i\right| \\ &+ |Y_{v0} - Y_{v1}u_c + Y_{v1}u_i|\,|v_i| + |Z_{v0}w_i| + |Z_{w0}v_i| \\ &+ |Y_{v1}u_c v_c - Z_{v0}w_c - Y_{v0}v_c - Y_{v1}u_i v_c| + |Y_{w0} - Y_{w1}u_c + Y_{w1}u_i|\,|w_i| \\ &+ |Y_{w1}u_c w_c - Z_{w0}v_c - Y_{w0}w_c - Y_{w1}u_i w_c|\,.\end{aligned} \tag{C.27}$$

Next, we use a similar strategy as in the previous section to get the following upper bound

$$\begin{aligned}\|\widehat{\boldsymbol{\omega}}_{0,i}\| \leq{}& \frac{\|\mathbf{e}_d\|}{\|\mathbf{e}_p\|}\frac{\|\boldsymbol{\upsilon}_i\|^2}{u_{\min}} + \frac{1}{2}\frac{|Z_{v0}+Z_{w0}|}{u_{\min}}\left(v_i^2+w_i^2\right) + |Y_{w1}u_c w_c| \\ &+ \max\left\{\frac{|Y_{v1}|(u_{\min}+|u_c|)+|Y_{v0}|}{u_{\min}}, \frac{|Y_{w1}|(u_{\min}+|u_c|)+|Y_{w0}|}{u_{\min}}\right\}\left(v_i^2+w_i^2\right) \\ &+ \frac{|Y_{v1}u_c v_c| + |Y_{v0}v_c| + |Z_{v0}w_c| + |Y_{v1}u_{\max}v_c|}{u_{\max}}\,|v_i| \\ &+ \left(|Y_{v0}| + |Y_{v1}u_c| + |Z_{w0}|\right)|v_i| + |Y_{v1}u_i v_i| + |Z_{w0}v_c| \\ &+ \frac{|Y_{w1}u_c w_c| + |Y_{w0}w_c| + |Z_{w0}v_c| + |Y_{w1}u_{\max}w_c|}{u_{\max}}\,|w_i| \\ &+ \left(|Y_{w0}| + |Y_{w1}u_c| + |Z_{v0}|\right)|w_i| + |Y_{w1}u_i w_i| + |Y_{w0}w_c| \\ &+ \left(|Y_{v1}v_c| + |Y_{w1}w_c|\right)|u_i| + |Y_{v1}u_c v_c| + |Z_{v0}w_c| + |Y_{v0}v_c|\end{aligned} \tag{C.28}$$

Note that the norm of $\boldsymbol{\upsilon}_i$ satisfies

$$\|\boldsymbol{\upsilon}_i\| = \|\mathbf{v}_{\mathrm{NSB},i}\| = U_{\mathrm{LOS}}\,\|\mathbf{e}_p\| \leq \frac{\|\mathbf{e}_p\|}{1-k_{\mathrm{NSB}}}\,\|\boldsymbol{\upsilon}_u\| + \frac{v_{2,\max}+u_{\min}}{1-k_{\mathrm{NSB}}}\,\|\mathbf{e}_p\|\,, \tag{C.29}$$

and the term $\left(v_i^2+w_i^2\right)$ satisfies the following two inequalities

$$v_i^2 + w_i^2 \leq \|\boldsymbol{\upsilon}_i\|^2\,, \qquad\qquad v_i^2 + w_i^2 \leq \|\boldsymbol{\upsilon}_u\|^2\,. \tag{C.30}$$

We finally arrive at the following upper bound on $\|\boldsymbol{\omega}_{0,i}\|$

$$\begin{aligned}
\|\boldsymbol{\omega}_{0,i}\| \leq & \left(\frac{\|\mathbf{e}_d\|}{u_{\min}\left(1-k_{\mathrm{NSB}}\right)} + \frac{1}{2}\frac{|Z_{v0}+Z_{w0}|}{u_{\min}} + |Y_{v1}| + |Y_{w1}| \right) \|\boldsymbol{\upsilon}_u\| \\
& + \max\left\{ \frac{|Y_{v1}|(u_{\min}+|u_c|)+|Y_{v0}|}{u_{\min}}, \frac{|Y_{w1}|(u_{\min}+|u_c|)+|Y_{w0}|}{u_{\min}} \right\} \|\boldsymbol{\upsilon}_u\| \\
& + \frac{\|\mathbf{e}_d\|(v_{2,\max}+u_{\min})}{u_{\min}\left(1-k_{\mathrm{NSB}}\right)} + \frac{|Y_{v1}u_cv_c| + |Y_{v0}v_c| + |Z_{v0}w_c| + |Y_{v1}u_{\max}v_c|}{u_{\max}} \\
& + |Y_{v1}u_c| + |Z_{w0}| + \frac{|Y_{w1}u_cw_c| + |Y_{w0}w_c| + |Z_{w0}v_c| + |Y_{w1}u_{\max}w_c|}{u_{\max}} \\
& + |Y_{v0}| + |Y_{w0}| + |Y_{w1}u_c| + |Z_{v0}| + |Y_{v1}v_c| + |Y_{w1}w_c| \\
& + \frac{|Y_{v1}u_cv_c| + |Z_{v0}w_c| + |Y_{v0}v_c| + |Y_{w1}u_cw_c| + |Z_{w0}v_c| + |Y_{w0}w_c|}{u_{\min}} \\
& \triangleq a_v \|\boldsymbol{\upsilon}_u\| + b_v
\end{aligned} \tag{C.31}$$

Similarly to the previous section, we can upper-bound $|u_c|$, $|v_c|$, and $|w_c|$ with $\|\mathbf{V}_c\|$. We have thus found positive constants a_v and b_v that satisfy (5.76).

Appendix D

Hand Position

D.1 The Coefficients in the Lyapunov Analysis

$$a_y = \frac{d_{35}m_{55} - d_{55}m_{35} + hd_{33}m_{55} - hd_{53}m_{35}}{h(m_{33}m_{55} - m_{35}^2)}, \tag{D.1a}$$

$$a_z = \frac{d_{66}m_{26} - d_{26}m_{66} + hd_{22}m_{66} - hd_{62}m_{26}}{h(m_{22}m_{66} - m_{26}^2)}, \tag{D.1b}$$

$$\begin{aligned} a_{xyz} = &\frac{m_{35}m_{44} - m_{26}m_{55} - m_{35}m_{66} + hm_{26}m_{35} + hm_{22}m_{55}}{h(m_{33}m_{55} - m_{35}^2)} \\ &- \frac{m_{26}m_{55} - m_{26}m_{44} + m_{35}m_{66} + hm_{26}m_{35} + hm_{33}m_{66}}{h(m_{22}m_{66} - m_{26}^2)}, \end{aligned} \tag{D.1c}$$

$$a_{xy} = -\frac{m_{26}m_{35} + m_{22}m_{55}}{h(m_{33}m_{55} - m_{35}^2)}, \tag{D.1d}$$

$$a_{xz} = -\frac{m_{26}m_{35} + m_{33}m_{66}}{h(m_{22}m_{66} - m_{26}^2)}, \tag{D.1e}$$

$$a_{ye} = \frac{m_{11}m_{55} - m_{35}^2 + hm_{11}m_{35} - hm_{33}m_{35}}{h(m_{33}m_{55} - m_{35}^2)}, \tag{D.1f}$$

$$a_{ze} = \frac{m_{11}m_{66} - m_{26}^2 - hm_{11}m_{26} + hm_{22}m_{26}}{h(m_{22}m_{66} - m_{26}^2)}, \tag{D.1g}$$

$$a_{ey} = \frac{m_{11}m_{35} - m_{33}m_{35}}{h(m_{33}m_{55} - m_{35}^2)}, \tag{D.1h}$$

$$a_{ez} = \frac{m_{11}m_{26} - m_{22}m_{26}}{h(m_{22}m_{66} - m_{26}^2)}, \tag{D.1i}$$

$$a_{ley} = \frac{d_{53}m_{35} - d_{33}m_{55}}{h(m_{33}m_{55} - m_{35}^2)}, \tag{D.1j}$$

$$a_{lez} = \frac{d_{22}m_{66} - d_{62}m_{26}}{h(m_{22}m_{66} - m_{26}^2)}. \tag{D.1k}$$

D.2 Components of the Jacobian Matrix

$$\Xi_{23} = \frac{d_{53}m_{35} - d_{33}m_{55} + \left\|\xi_{2_{d,r}}\right\| m_{35}\left(m_{11} - m_{33}\right)}{h\left(m_{33}m_{55} - {m_{35}}^2\right)} + \frac{k_d}{h}, \tag{D.2a}$$

$$\Xi_{32} = \frac{d_{22}m_{66} - d_{62}m_{26} + \left\|\xi_{2_{d,r}}\right\| m_{26}\left(m_{11} - m_{22}\right)}{h\left(m_{22}m_{66} - {m_{26}}^2\right)} + \frac{k_d}{h}, \tag{D.2b}$$

$$\Delta_1 = b_x, \tag{D.2c}$$

$$\Delta_2 = \frac{\left\|\xi_{2_{d,r}}\right\|}{h}\left(a_y - \frac{m_{35}\left\|\xi_{2_{d,r}}\right\|\left(m_{11} - m_{33}\right)}{m_{33}m_{55} - {m_{35}}^2}\right), \tag{D.2d}$$

$$\Delta_3 = \frac{\left\|\xi_{2_{d,r}}\right\|}{h}\left(a_z + \frac{m_{26}\left\|\xi_{2_{d,r}}\right\|\left(m_{11} - m_{22}\right)}{m_{22}m_{66} - {m_{26}}^2}\right), \tag{D.2e}$$

$$\Omega_1 = a_x, \tag{D.2f}$$

$$\Omega_2 = a_y + \left\|\xi_{2_{d,r}}\right\|\left(a_{ye} + \frac{1}{h}\right) + \frac{d_{35}m_{55} - d_{55}m_{35}}{h(m_{33}m_{55} - {m_{35}}^2)}, \tag{D.2g}$$

$$\Omega_3 = a_z + \left\|\xi_{2_{d,r}}\right\|\left(a_{ze} + \frac{1}{h}\right) + \frac{d_{26}m_{66} - d_{66}m_{26}}{h(m_{22}m_{66} - {m_{26}}^2)}. \tag{D.2h}$$

References

[1] R. Pi, P. Cieślak, P. Ridao, and P. J. Sanz, "TWINBOT: Autonomous underwater cooperative transportation," *IEEE Access*, vol. 9, 2021.

[2] E. F. Brekke, E. Eide, B. H. Eriksen, E. F. Wilthil, M. Breivik, E. Skjellaug, Ø. K. Helgesen, A. M. Lekkas, A. B. Martinsen, E. H. Thyri, *et al.*, "milliAmpere: An autonomous ferry prototype," in *Journal of Physics: Conference Series*, vol. 2311, 2022.

[3] A. Palomer, P. Ridao, and D. Ribas, "Inspection of an underwater structure using point-cloud SLAM with an AUV and a laser scanner," *Journal of Field Robotics*, vol. 36, no. 8, pp. 1333–1344, 2019.

[4] D. McLeod, "Emerging capabilities for autonomous inspection repair and maintenance," in *Proc. OCEANS*, 2010.

[5] B. Bingham, B. Foley, H. Singh, R. Camilli, K. Delaporta, R. Eustice, A. Mallios, D. Mindell, C. Roman, and D. Sakellariou, "Robotic tools for deep water archaeology: Surveying an ancient shipwreck with an autonomous underwater vehicle," *Journal of Field Robotics*, vol. 27, no. 6, 2010.

[6] S. Petillo, H. Schmidt, and A. Balasuriya, "Constructing a Distributed AUV Network for Underwater Plume-Tracking Operations," *International Journal of Distributed Sensor Networks*, vol. 8, Jan. 2012.

[7] I. C. Robbins, G. J. Kirkpatrick, S. M. Blackwell, J. Hillier, C. A. Knight, and M. A. Moline, "Improved monitoring of HABs using autonomous underwater vehicles (AUV)," *Harmful Algae*, vol. 5, no. 6, Dec. 2006.

[8] P. Kimball, J. Bailey, S. Das, R. Geyer, T. Harrison, C. Kunz, K. Manganini, K. Mankoff, K. Samuelson, T. Sayre-McCord, F. Straneo, P. Traykovski, and H. Singh, "The WHOI Jetyak: An autonomous surface vehicle for oceanographic research in shallow or dangerous waters," in *Proc. 2014 IEEE/OES Autonomous Underwater Vehicles*, Oct. 2014.

[9] J. Choi, J. Park, J. Jung, Y. Lee, and H.-T. Choi, "Development of an autonomous surface vehicle and performance evaluation of autonomous navigation technologies," *International Journal of Control, Automation and Systems*, vol. 18, no. 3, pp. 535–545, Mar. 2020.

[10] A. Zolich, A. Faltynkova, G. Johnsen, and T. A. Johansen, "Portable Catamaran Drone – an uncrewed sampling vehicle for micro-plastics and aquaculture research," in *Proc. OCEANS*, Oct. 2022.

[11] *Blue Robotics BlueROV2 Datasheet*, `https://bluerobotics.com/store/rov/bluerov2/`, Accessed 2023-01-10.

[12] A. Sousa, L. Madureira, J. Coelho, J. Pinto, J. Pereira, J. Borges Sousa, and P. Dias, "LAUV: The man-portable autonomous underwater vehicle," in *3rd IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles*, 2012.

[13] M. Purcell, C. von Alt, B. Allen, T. Austin, N. Forrester, R. Goldsborough, and R. Stokey, "New capabilities of the REMUS autonomous underwater vehicle," in *Proc. OCEANS*, Sep. 2000.

[14] C. Paliotta, E. Lefeber, K. Y. Pettersen, J. Pinto, M. Costa, and J. T. de Figueiredo Borges de Sousa, "Trajectory tracking and path following for underactuated marine vehicles," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 4, pp. 1423–1437, 2019.

[15] F. Rezazadegan, K. Shojaei, F. Sheikholeslam, and A. Chatraei, "A novel approach to 6-DOF adaptive trajectory tracking control of an AUV in the presence of parameter uncertainties," *Ocean Engineering*, vol. 107, Oct. 2015.

[16] F. Alonge, F. D'Ippolito, and F. Raimondi, "Trajectory tracking of underactuated underwater vehicles," in *Proc. 40th IEEE Conference on Decision and Control*, vol. 5, Dec. 2001.

[17] T. Elmokadem, M. Zribi, and K. Youcef-Toumi, "Trajectory tracking sliding mode control of underactuated AUVs," *Nonlinear Dynamics*, Apr. 2016.

[18] A. P. Aguiar and J. P. Hespanha, "Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty," *IEEE Transactions on Automatic Control*, vol. 52, no. 8, Aug. 2007.

[19] M. Abdelaal, M. Franzle, and A. Hahn, "Nonlinear model predictive control for tracking of underactuated vessels under input constraints," in *Proc. 2015 IEEE European Modelling Symposium*, Oct. 2015.

[20] W. Caharija, K. Y. Pettersen, M. Bibuli, P. Calado, E. Zereik, J. Braga, J. T. Gravdahl, A. J. Sørensen, M. Milovanović, and G. Bruzzone, "Integral line-of-sight guidance and control of underactuated marine vehicles: Theory, simulations, and experiments," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 5, pp. 1623–1642, Sep. 2016.

[21] X. Xiang, C. Yu, and Q. Zhang, "Robust fuzzy 3D path following for autonomous underwater vehicle subject to uncertainties," *Computers & Operations Research*, vol. 84, pp. 165–177, Aug. 2017.

[22] J. Miao, S. Wang, Z. Zhao, Y. Li, and M. M. Tomovic, "Spatial curvilinear path following control of underactuated AUV with multiple uncertainties," *ISA Transactions*, vol. 67, pp. 107–130, Mar. 2017.

[23] B. Das, B. Subudhi, and B. B. Pati, "Cooperative formation control of autonomous underwater vehicles: An overview," en, *International Journal of Automation and Computing*, vol. 13, no. 3, pp. 199–225, Jun. 2016.

[24] E. Borhaug and K. Y. Pettersen, "Formation control of 6-DOF Euler Lagrange systems with restricted inter-vehicle communication," in *Proc. 45th IEEE Conference on Decision and Control*, 2006.

[25] R. P. Jain, A. P. Aguiar, and J. B. de Sousa, "Cooperative path following of robotic vehicles using an event-based control and communication strategy," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, 2018.

[26] R. Cui, S. Sam Ge, B. Voon Ee How, and Y. Sang Choo, "Leader-follower formation control of underactuated autonomous underwater vehicles," *Ocean Engineering*, vol. 37, no. 17, pp. 1491–1502, 2010.

[27] M. Soorki, H. Talebi, and S. Nikravesh, "A robust dynamic leader-follower formation control with active obstacle avoidance," in *Proc. 2011 IEEE International Conference on Systems, Man, and Cybernetics*, 2011.

[28] Y. Wang, Y. Yang, Y. Pu, and C. Manzie, "Path following by formations of agents with collision avoidance guarantees using distributed model predictive control," in *Proc. 2021 American Control Conference*, 2021.

[29] K. Kanjanawanishkul and A. Zell, "Distributed model predictive control for coordinated path following control of omnidirectional mobile robots," in *Proc. 2008 IEEE International Conference on Systems, Man and Cybernetics*, 2008.

[30] M. Saska, V. Spurný, and V. Vonásek, "Predictive control and stabilization of nonholonomic formations with integrated spline-path planning," *Robotics and Autonomous Systems*, vol. 75, pp. 379–397, 2016.

[31] R. Van Parys and G. Pipeleers, "Distributed MPC for multi-vehicle systems moving in formation," *Robotics and Autonomous Systems*, 2017.

[32] G. Antonelli, F. Arrichiello, and S. Chiaverini, "Experiments of formation control with multirobot systems using the null-space-based behavioral control," *IEEE Transactions on Control Systems Technology*, 2009.

[33] F. Arrichiello, S. Chiaverini, and T. I. Fossen, "Formation control of underactuated surface vessels using the null-space-based behavioral control," in *Proc. 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006.

[34] S.-K. Pang, Y.-H. Li, and H. Yi, "Joint formation control with obstacle avoidance of towfish and multiple autonomous underwater vehicles based on graph theory and the null-space-based method," *Sensors*, 2019.

[35] Å. Eek, K. Y. Pettersen, E.-L. M. Ruud, and T. R. Krogstad, "Formation path following control of underactuated USVs," *European Journal of Control*, vol. 62, 2021.

[36] S. Ahmad, Z. Feng, and G. Hu, "Multi-robot formation control using distributed null space behavioral approach," in *Proc. 2014 IEEE International Conference on Robotics and Automation*, 2014.

[37] G. Tan, H. Sun, L. Du, J. Zhuang, J. Zou, and L. Wan, "Coordinated control of the heterogeneous unmanned surface vehicle swarm based on the distributed null-space-based behavioral approach," *Ocean Engineering*, vol. 266, 2022.

[38] T. Statheros, G. Howells, and K. M. Maier, "Autonomous ship collision avoidance navigation concepts, technologies and techniques," *The Journal of Navigation*, vol. 61, no. 1, Jan. 2008.

[39] C. Tam, R. Bucknall, and A. Greig, "Review of collision avoidance and path planning methods for ships in close range encounters," *The Journal of Navigation*, vol. 62, no. 3, Jul. 2009.

[40] M. Hoy, A. S. Matveev, and A. V. Savkin, "Algorithms for Collision-free navigation of mobile robots in complex cluttered environments: A survey," *Robotica*, vol. 33, no. 3, Mar. 2015.

[41] T. Wang, X. P. Yan, Y. Wang, and Q. Wu, "Ship domain model for multi-ship collision avoidance decision-making with COLREGs based on artificial potential field," *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation*, 2017.

[42] Y. Kuwata, M. T. Wolf, D. Zarzhitsky, and T. L. Huntsberger, "Safe maritime autonomous navigation with COLREGS, using velocity obstacles," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, 2014.

[43] H.-T. L. Chiang and L. Tapia, "COLREG-RRT: An RRT-based COLREGs-compliant motion planner for surface vehicle navigation," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, 2018.

[44] A. Lazarowska, "Ship's trajectory planning for collision avoidance at sea based on ant colony optimisation," *The Journal of Navigation*, vol. 68, no. 2, Mar. 2015.

[45] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics Automation Magazine*, vol. 4, no. 1, Mar. 1997.

[46] I. B. Hagen, D. K. M. Kufoalor, E. F. Brekke, and T. A. Johansen, "MPC-based collision avoidance strategy for existing marine vessel guidance systems," in *Proc. 2018 IEEE International Conference on Robotics and Automation*, May 2018.

[47] X. Sun, G. Wang, Y. Fan, D. Mu, and B. Qiu, "Collision avoidance using finite control set model predictive control for unmanned surface vehicle," *Applied Sciences*, vol. 8, no. 6,

[48] Y. Kuriki and T. Namerikawa, "Formation control with collision avoidance for a multi-UAV system using decentralized MPC and consensus-based control," *SICE Journal of Control, Measurement, and System Integration*, vol. 8, no. 4, 2015.

[49] L. Dai, Q. Cao, Y. Xia, and Y. Gao, "Distributed MPC for formation of multi-agent systems with collision avoidance and obstacle avoidance," *Journal of the Franklin Institute*, vol. 354, no. 4, Mar. 2017.

[50] G. P. Roussos, D. V. Dimarogonas, and K. J. Kyriakopoulos, "3D navigation and collision avoidance for a non-holonomic vehicle," in *Proc. 2008 American Control Conference*, Jun. 2008.

[51] A. Mujumdar and R. Padhi, "Reactive collision avoidance using nonlinear geometric and differential geometric guidance," *Journal of Guidance, Control, and Dynamics*, vol. 34, no. 1, Jan. 2011.

[52] E. Squires, P. Pierpaoli, and M. Egerstedt, "Constructive barrier certificates with applications to fixed-wing aircraft collision avoidance," in *Proc. 2nd IEEE Conference on Control Technology and Applications*, Aug. 2018.

[53] M. Igarashi and H. Nakamura, "Collision avoidance assist control for two-wheel vehicle robots by control barrier function," in *Proc. 2018 International Automatic Control Conference*, Nov. 2018.

[54] M. Z. Romdlony and B. Jayawardhana, "Stabilization with guaranteed safety using control lyapunov–barrier function," *Automatica*, vol. 66, Apr. 2016.

[55] E. A. Basso, E. H. Thyri, K. Y. Pettersen, M. Breivik, and R. Skjetne, "Safety-critical control of autonomous surface vehicles in the presence of ocean currents," in *Proc. 4th IEEE Conference on Control Technology and Applications*, Aug. 2020.

[56] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *Proc. 53rd IEEE Conference on Decision and Control*, Dec. 2014.

[57] R. Hedjar and M. Bounkhel, "An automatic collision avoidance algorithm for multiple marine surface vehicles," *International Journal of Applied Mathematics and Computer Science*, vol. 29, no. 4, Dec. 2019.

[58] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *Proc. 18th European Control Conference*, Jun. 2019.

[59] S. K. Gan, R. Fitch, and S. Sukkarieh, "Online decentralized information gathering with spatial-temporal constraints," *Autonomous Robots*, vol. 37, pp. 1–25, 2014.

[60] R. Gomes and F. L. Pereira, "A model predictive control scheme for autonomous underwater vehicle formation control," in *Proc. 13th APCA International Conference on Automatic Control and Soft Computing*, 2018, pp. 195–200.

[61] Q. Jia and G. Li, "Formation control and obstacle avoidance algorithm of multiple autonomous underwater vehicles (AUVs) based on potential function and behavior rules," in *Proc. 2007 IEEE International Conference on Automation and Logistics*, 2007, pp. 569–573.

[62] L. Briñón Arranz, A. Seuret, and C. Canudas de Wit, "Contraction control of a fleet circular formation of AUVs under limited communication range," in *Proc. 2010 American Control Conference*, 2010, pp. 5991–5996.

[63] Z. Gao and G. Guo, "Velocity free leader-follower formation control for autonomous underwater vehicles with line-of-sight range and angle constraints," *Information Sciences*, pp. 359–378, 2019.

[64] A. Naderolasli, K. Shojaei, and A. Chatraei, "Platoon formation control of autonomous underwater vehicles under LOS range and orientation angles constraints," *Ocean Engineering*, 2023.

[65] T. A. Johansen and T. I. Fossen, "Control allocation—A survey," *Automatica*, vol. 49, no. 5, May 2013.

[66] M. W. Oppenheimer, D. B. Doman, and M. A. Bolender, "Control allocation for over-actuated systems," in *Proc. 14th Mediterranean Conference on Control and Automation*, Jun. 2006.

[67] O. Härkegård, "Dynamic control allocation using constrained quadratic programming," *Journal of Guidance, Control, and Dynamics*, vol. 27, 2004.

[68] T. A. Johansen, T. I. Fossen, and S. P. Berge, "Constrained nonlinear control allocation with singularity avoidance using sequential quadratic programming," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 1, Jan. 2004.

[69] E. H. Thyri, E. A. Basso, M. Breivik, K. Y. Pettersen, R. Skjetne, and A. M. Lekkas, "Reactive collision avoidance for ASVs based on control barrier functions," in *Proc. 4th IEEE Conference on Control Technology and Applications*, 2020.

[70] K. Y. Pettersen, "Lyapunov sufficient conditions for uniform semiglobal exponential stability," *Automatica*, vol. 78, Apr. 2017.

[71] S. Moe, K. Y. Pettersen, T. I. Fossen, and J. T. Gravdahl, "Line-of-sight curved path following for underactuated USVs and AUVs in the horizontal plane under the influence of ocean currents," in *Proc. 24th Mediterranean Conference on Control and Automation*, 2016.

[72] E. Borhaug, A. Pavlov, and K. Y. Pettersen, "Straight line path following for formations of underactuated underwater vehicles," in *Proc. 46th IEEE Conference on Decision and Control*, 2007.

[73] J.-B. Pomet, B. Thuilot, G. Bastin, and G. Campion, "A hybrid strategy for the feedback stabilization of nonholonomic mobile robots," in *Proc. 1992 IEEE International Conference on Robotics and Automation*, Jan. 1992.

[74] J. Lawton, R. Beard, and B. Young, "A decentralized approach to formation maneuvers," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 6, pp. 933–941, 2003.

[75] X. Cai and M. d. Queiroz, "Adaptive rigidity-based formation control for multirobotic vehicles with dynamics," *IEEE Transactions on Control Systems Technology*, vol. 23, pp. 389–396, Jan. 2015.

[76] X. Li, C. Wen, and C. Chen, "Adaptive formation control of networked robotic systems with bearing-only measurements," *IEEE Transactions on Cybernetics*, vol. 51, pp. 199–209, Jan. 2021.

[77] E. Restrepo, J. Matouš, and K. Y. Pettersen, "Tracking-in-formation of multiple autonomous marine vehicles under proximity and collision-avoidance constraints," in *Proc. 2022 European Control Conference*, Jul. 2022.

[78] M. Mesbahi and M. Egerstedt, *Graph theoretic methods in multiagent networks* (Princeton series in applied mathematics). 2010.

[79] J. Matouš, K. Y. Pettersen, D. Varagnolo, and C. Paliotta, "Singularity-free formation path following of underactuated AUVs," in *Proc. 2023 IFAC World Congress*, 2023.

[80] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, "Differential kinematics and statics," in *Robotics: Modelling, Planning and Control*, 2009.

[81] S. Chiaverini, G. Oriolo, and I. D. Walker, "Kinematically redundant manipulators," in *Springer Handbook of Robotics*, 2008, pp. 245–268.

[82] J. Matouš, E. A. Basso, E. H. Thyri, and K. Y. Pettersen, "Unifying reactive collision avoidance and control allocation for multi-vehicle systems," in *Proc. 2021 IEEE Conference on Control Technology and Applications*, 2021.

[83] J. Matouš, K. Y. Pettersen, and C. Paliotta, "Formation path following control of underactuated AUVs," in *Proc. 2022 European Control Conference*, 2022.

[84] J. Matouš, D. Varagnolo, K. Y. Pettersen, and C. Paliotta, "Distributed mpc for formation path-following of multi-vehicle systems," in *Proc. 9th IFAC Conference on Networked Systems*, Jul. 2022.

[85] E. S. Lie, J. Matouš, and K. Y. Pettersen, "Formation control of underactuated AUVs using the hand position concept," in *To appear in Proc. 2023 IEEE Conference on Decision and Control*, 2023.

[86] J. Matouš, K. Y. Pettersen, D. Varagnolo, and C. Paliotta, "A distributed NSB algorithm for formation path following," *Submitted to IEEE Transactions on Control Systems Technology*, 2023.

[87] J. Matouš, C. Paliotta, K. Y. Pettersen, and D. Varagnolo, "Trajectory tracking and path following of underactuated AUVs using the hand position concept," *Submitted to IEEE Transactions on Control Systems Technology*, 2023.

[88] E. Restrepo, J. Matouš, and K. Y. Pettersen, "Tracking control of cooperative marine vehicles under hard and soft constraints," *Submitted to IEEE Transactions on Control of Network Systems*, 2023.

[89] *DUNE: Unified navigation environment*, `https://github.com/LSTS/dune/`, Accessed 2023-04-25.

[90] E. Fredriksen and K. Pettersen, "Global κ-exponential waypoint maneuvering of ships: Theory and experiments," *Automatica*, vol. 42, no. 4, 2006.

[91] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, May 2011.

[92] N. A. Chaturvedi, A. K. Sanyal, and N. H. McClamroch, "Rigid-body attitude control," *IEEE Control Systems Magazine*, vol. 31, no. 3, 2011.

[93] S. P. Bhat and D. S. Bernstein, "A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon," *Systems & Control Letters*, vol. 39, no. 1, 2000.

[94] B. A. Barsky and T. D. DeRose, "Geometric continuity of parametric curves," *Computer Graphics and Applications*, 1984.

[95] N. Gu, D. Wang, Z. Peng, J. Wang, and Q.-L. Han, "Advances in line-of-sight guidance for path following of autonomous marine vehicles: An overview," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 1, pp. 12–28, 2023.

[96] K. Pettersen and E. Lefeber, "Way-point tracking control of ships," in *Proc. 40th IEEE Conference on Decision and Control*, 2001.

[97] W. Caharija, K. Y. Pettersen, J. T. Gravdahl, and E. Børhaug, "Path following of underactuated autonomous underwater vehicles in the presence of ocean currents," in *Proc. 51st IEEE Conference on Decision and Control*, 2012, pp. 528–535.

[98] B. Abdurahman, A. Savvaris, and A. Tsourdos, "Switching LOS guidance with speed allocation and vertical course control for path following of unmanned underwater vehicles under ocean current disturbances," *Ocean Engineering*, vol. 182, pp. 412–426, 2019.

[99] M. Breivik and T. Fossen, "Principles of guidance-based path following in 2D and 3D," in *Proc. 44th IEEE Conference on Decision and Control*, 2005.

[100] C. Yu, X. Xiang, L. Lapierre, and Q. Zhang, "Nonlinear guidance and fuzzy control for three-dimensional path following of an underactuated autonomous underwater vehicle," *Ocean Engineering*, vol. 146, 2017.

[101] C. Yu, C. Liu, X. Xiang, Z. Zeng, Z. Wei, and L. Lian, "Line-of-sight guided time delay control for three-dimensional coupled path following of underactuated underwater vehicles with roll dynamics," *Ocean Engineering*, 2020.

[102] G. Antonelli and S. Chiaverini, "Kinematic control of platoons of autonomous vehicles," *IEEE Transactions on Robotics*, vol. 22, no. 6, 2006.

[103] G. Antonelli, F. Arrichiello, and S. Chiaverini, "Stability analysis for the null-space-based behavioral control for multi-robot systems," in *Proc. 47th IEEE Conference on Decision and Control*, 2008.

[104] T. I. Fossen and K. Y. Pettersen, "On uniform semiglobal exponential stability (USGES) of proportional line-of-sight guidance laws," *Automatica*, vol. 50, no. 11, pp. 2912–2917, 2014.

[105] A. Loría and E. Panteley, "Cascaded nonlinear time-varying systems: Analysis and design," in *Advanced Topics in Control Systems Theory: Lecture Notes from FAP 2004*, Springer, 2005, pp. 23–64.

[106] A. A. Pedersen, "Optimization based system identification for the milli-Ampere ferry," 2019, Master's thesis, Norwegian University of Science and Technology (NTNU).

[107] J. Bjørnø, "Thruster-assisted position mooring of C/S Inocean Cat I Drillship," 2016, Master's thesis, Norwegian University of Science and Technology (NTNU).

[108] S. Moe and K. Y. Pettersen, "Set-based line-of-sight (LOS) path following with collision avoidance for underactuated unmanned surface vessels under the influence of ocean currents," in *Proc. 2017 IEEE Conference on Control Technology and Applications*, 2017.

[109] D. Belleter, M. A. Maghenem, C. Paliotta, and K. Y. Pettersen, "Observer based path following for underactuated marine vessels in the presence of ocean currents: A global approach," *Automatica*, vol. 100, pp. 123–134, 2019.

[110] D. Angeli and E. D. Sontag, "Forward completeness, unboundedness observability, and their Lyapunov characterizations," *Systems & Control Letters*, vol. 38, no. 4-5, pp. 209–217, 1999.

[111] A. Iserles, H. Z. Munthe-Kaas, S. P. Nørsett, and A. Zanna, "Lie-group methods," *Acta Numerica*, vol. 9, 2000.

[112] T. Lee, "Global exponential attitude tracking controls on SO(3)," *IEEE Transactions on Automatic Control*, vol. 60, no. 10, 2015.

[113] A. Chakravarthy and D. Ghose, "Obstacle avoidance in a dynamic environment: A collision cone approach," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 28, no. 5, 1998.

[114] M. S. Wiig, K. Y. Pettersen, and T. R. Krogstad, "Collision avoidance for underactuated marine vehicles using the constant avoidance angle algorithm," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 3, 2019.

[115] A. Haraldsen, M. S. Wiig, and K. Y. Pettersen, "Reactive collision avoidance for underactuated surface vehicles using the collision cone concept," in *Proc. 2021 IEEE Conference on Control Technology and Applications*, 2021.

[116] D. Nakath, J. Clemens, and C. Rachuy, "Rigid body attitude control based on a manifold representation of direction cosine matrices," in *Proc. 13th European Workshop on Advanced Control and Diagnosis*, 2017.

[117] S. A. Al-Zhrani, N. M. Bedaiwi, I. F. E. El-Ramli, A. Z. Barasheed, A. Abduldaiem, Y. A. Al-Hadeethi, and A. Umar, "Underwater optical communications: A brief overview and recent developments," *Engineered Science*, vol. 16, pp. 146–186, 2021.

[118] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 84–89, 2009.

[119] L. Ding, Q.-L. Han, X. Ge, and X.-M. Zhang, "An overview of recent advances in event-triggered consensus of multiagent systems," *IEEE Transactions on Cybernetics*, vol. 48, no. 4, pp. 1110–1123, 2017.

[120] H. K. Khalil, *Nonlinear Systems*, 3rd edition. Prentice Hall, 2002.

[121] M. Breivik and T. I. Fossen, "Path following of straight lines and circles for marine surface vessels," in *Proc. 2004 IFAC Conference on Computer Applications in Marine Systems*, 2004, pp. 65–70.

[122] A. P. Aguiar and J. P. Hespanha, "Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty," *IEEE Transactions on Automatic Control*, vol. 52, no. 8, pp. 1362–1379, 2007.

[123] M. Fliess, J. Lévine, P. Martin, and P. Rouchon, "Flatness and defect of nonlinear systems: Introductory theory and examples," *International Journal of Control*, 1995.

[124] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Foundations and Trends in Machine Learning, 2011, pp. 1–122.

[125] N. Bastianello, R. Carli, L. Schenato, and M. Todescato, "Asynchronous distributed optimization over lossy networks via relaxed ADMM: Stability and linear convergence," *IEEE Transactions on Automatic Control*, 2021.

[126] D. Davis and W. Yin, "Convergence rate analysis of several splitting schemes," in *Splitting Methods in Communication, Imaging, Science, and Engineering*, R. Glowinski, S. J. Osher, and W. Yin, Eds. 2016.

[127] Z. Zhu, S.-L. J. Hu, and H. Li, "Effect on Kalman based underwater tracking due to ocean current uncertainty," in *Proc. 2016 IEEE/OES Autonomous Underwater Vehicles*, 2016, pp. 131–137.

[128] E. Restrepo, A. Loría, I. Sarras, and J. Marzat, "Edge-based strict Lyapunov functions for consensus with connectivity preservation over directed graphs," *Automatica*, vol. 132, 2021.

[129] K. P. Tee, S. S. Ge, and E. H. Tay, "Barrier Lyapunov functions for the control of output-constrained nonlinear systems," *Automatica*, vol. 45, no. 4, pp. 918–927, 2009.

[130] E. Rimon and D. E. Koditschek, "Exact robot navigation using artificial potential functions," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 5, pp. 501–518, Oct. 1992.

[131] P. Forni and D. Angeli, "Input-to-state stability for cascade systems with multiple invariant sets," *Systems & Control Letters*, vol. 98, pp. 97–110, 2016.

[132] P. Monzón and R. Potrie, "Local and global aspects of almost global stability," in *Proc. 45th IEEE Conference on Decision and Control*, San Diego, CA, USA, 2006, pp. 5120–5125.

[133] D. Mukherjee and D. Zelazo, "Robustness of consensus over weighted digraphs," *IEEE Transactions on Network Science and Engineering*, vol. 6, no. 4, pp. 657–670, 2019.

[134] F. Mehdifar, C. P. Bechlioulis, and D. V. Dimarogonas, "Funnel control under hard and soft output constraints," in *Proc. 61st Conference on Decision and Control*, 2022, pp. 4473–4478.

[135] B. Siciliano, "A closed-loop inverse kinematic scheme for on-line joint-based robot control," *Robotica*, vol. 8, no. 3, pp. 231–243, 1990.

ISBN 978-82-326-7394-0 (printed ver.)
ISBN 978-82-326-7393-3 (electronic ver.)
ISSN 1503-8181 (printed ver.)
ISSN 2703-8084 (online ver.)

NTNU

Norwegian University of
Science and Technology