COLLECTIVE DYNAMICS AND CONTROL OF A FLEET OF HETEROGENEOUS MARINE VEHICLES

A Thesis
Presented to
The Academic Faculty

by

Chuanfeng Wang

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
George W. Woodruff School of Mechanical Engineering

Georgia Institute of Technology
December 2013

COLLECTIVE DYNAMICS AND CONTROL OF A FLEET OF HETEROGENEOUS MARINE VEHICLES

Approved by:

Dr. Dirk Schaefer, Co-advisor,
Committee Chair
George W. Woodruff School of Mechanical
Engineering
Georgia Institute of Technology

Dr. Jun Ueda
George W. Woodruff School of Mechanical
Engineering
Georgia Institute of Technology

Dr. Jianxin Jiao
George W. Woodruff School of Mechanical
Engineering
Georgia Institute of Technology

Dr. Fumin Zhang, Co-advisor
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Dr. Ayanna Howard
School of Electrical and Computer
Engineering
Georgia Institute of Technology

Date Approved: October 28, 2013

This thesis is dedicated to my parents for their unconditional love and endless support.

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the help of numerous people. I owe my gratitude to all these people.

First, I would like to express my deepest appreciation to my advisors Dr. Dirk Schaefer and Dr. Fumin Zhang, who, in the past few years, have provided me with a great amount of guidance and support. They spent lots of time providing me advice, correction, and scientific research training and contributed significantly to all results in this dissertation. I would also like to extend my deepest gratitude to my committee professors, Dr. Ayanna Howard, Dr. Jun Ueda, and Dr. Jianxin (Roger) Jiao, for their valuable suggestions for the work presented in this dissertation.

I would like to extend my sincere thanks to Dr. Huizhen Yang, a professor at Northwestern Polytechnical University in Xi'an, China. Results in Chapter 3 of this dissertation would not have been possible without the collaboration with her. I am also grateful to my labmate, Carol Young, for providing inspiring discussion and suggestions for describing human attention in Chapter 4. I very much appreciate Dr. Mark Patterson, a professor at the College of William and Mary, and Dr. Michael Malisoff, a professor at Louisiana State University, who supported the field experiment project in Chapter 5 together with Dr. Fumin Zhang. I am deeply indebted to my labmate, Shayok Mukhopadhyay, who contributed significantly to the project by leading the student team. He carried out the parameter tuning and performance evaluation for controllers during the field experiment. To complete this dissertation, I incorporated his work into Chapter 5. Thanks should also go to undergraduate students (at that time), Steven Bradshaw, Sen Maxon, Valerie Bazie, Lisa Hicks, Phillip Cheng, and Brian Redden, who also contributed significantly by assisting in operating the vehicles during the project. We would like to acknowledge the support of the Grand Isle State Park during the field experiment.

Many thanks to all my fellow lab members and my friends who provided collaboration and assistance during my study.

Finally, my special appreciation goes to my parents for their endless love and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

3D: three-dimensional

AUV: autonomous underwater vehicle

CFD: computational fluid dynamics

COB: center of buoyancy

COM: center of mass

DOF: degree of freedom

human-AUV system: system comprised of an HUV and a group of AUVs

HUV: human-driven underwater vehicle

multi-AUV system: system comprised of multiple AUVs

SUMMARY

Cooperative control enables combinations of sensor data from multiple autonomous underwater vehicles (AUVs) so that multiple AUVs can perform smarter behaviors than a single AUV. In addition, in some situations, a human-driven underwater vehicle (HUV) and a group of AUVs need to collaborate and preform formation behaviors. However, the collective dynamics of a fleet of heterogeneous underwater vehicles are more complex than the non-trivial single vehicle dynamics, resulting in challenges in analyzing the formation behaviors of a fleet of heterogeneous underwater vehicles. The research addressed in this dissertation investigates the collective dynamics and control of a fleet of heterogeneous underwater vehicles, including multi-AUV systems and systems comprised of an HUV and a group of AUVs (human-AUV systems). This investigation requires a mathematical motion model of an underwater vehicle. This dissertation presents a review of a six-degree-of-freedom (6DOF) motion model of a single AUV and proposes a method of identifying all parameters in the model based on computational fluid dynamics (CFD) calculations. Using the method, we build a 6DOF model of the EcoMapper and validate the model by field experiments. Based upon a generic 6DOF AUV model, we study the collective dynamics of a multi-AUV system and develop a method of decomposing the collective dynamics. After the collective dynamics decomposition, we propose a method of achieving orientation control for each AUV and formation control for the multi-AUV system. We extend the results and propose a cooperative control for a human-AUV system so that an HUV and a group of AUVs will form a desired formation while moving along a desired trajectory as a team. For the post-mission stage, we present a method of analyzing AUV survey data and apply this method to AUV measurement data collected from our field experiments carried out in Grand Isle, Louisiana in 2011, where AUVs were used to survey a lagoon, acquire bathymetric data, and measure the concentration of reminiscent crude oil in the water of the lagoon after the BP Deepwater Horizon oil spill in the Gulf of Mexico in 2010.

CHAPTER I

INTRODUCTION

Formation behaviors, such as fish schooling, bird flocking, and insect swarming, are common in nature. Formation behavior benefits each animal in formation, for it maximizes the detection of predators or the more efficient foraging of food by combining their sensors [1]. Similar to the formation behaviors of creatures in nature, the formation behaviors of artificial agents also enable combinations of sensor data; therefore, a group of artificial agents in a formation can create smarter behaviors than a single agent does. Numerous studies have been devoted to the investigation of the formation control of multiple autonomous underwater vehicles (multi-AUV system) [2–10] because of its broad applications in oceanographic research, seafloor surveys, underwater archeology and meteorology. These studies have proven that enabling a number of autonomous underwater vehicles (AUVs) to work cooperatively is advantageous [11]. In addition, in some situations, to execute a task, a number of AUVs and a human-driven vehicle (HUV) need to collaborate and perform formation behaviors. For example, a group of AUVs equipped with different tools and devices need to be in specified positions relative to an HUV to assist the human operator, who is working on a task that only humans can perform. In this case, the HUV and AUVs need to get into a specified formation while they move along a trajectory together. Another example is when the human operator of HUV needs to ensure safety by being in a particular position in the formation formed by the HUV and AUVs, while they follow a specified course for executing a task. In both cases, the system comprised of an HUV and a group of AUVs (human-AUV system) needs to take on a desired formation and track a desired trajectory. This dissertation focuses on the control of the formation behavior of a fleet of heterogeneous marine vehicles, including multi-AUV systems and human-AUV systems. The objective of this dissertation is to develop a control methodology that enables a fleet of heterogeneous marine vehicles to collaborate by taking on and maintaining desired formations and going along desired trajectories as a team. To achieve this purpose, since the dynamics of a single underwater vehicle is the basis of investigating the collective dynamics of

all underwater vehicles, we study the dynamics of a single underwater vehicle and present a method of modeling its motion in 3D space. Based on the dynamic model of a single AUV, we study the collective dynamics of a multi-AUV system and present a decoupled controller design approach for its formation dynamics. The results on multi-AUV system is then extended to a human-AUV system so that we present the cooperative control for the human-AUV system to enable the HUV and AUVs to cooperate to complete a task together. If the multi-AUV system or human-AUV system are used for environmental survey, they will return survey data after they complete the task. For the post-mission stage, we present a method of analogizing the survey data collected by marine vehicles.

The study of the formation behaviors of a fleet of heterogeneous underwater vehicles requires a dynamic model of a single underwater vehicle. Using the EcoMapper as an example, we propose a method of building a dynamic model for underwater vehicles. The EcoMapper is an AUV with broad applications such as water quality monitoring and bathymetric surveying [12–15]. To simulate the dynamics of and precisely control the EcoMapper, we develop a mathematical motion model based on computational-fluid-dynamics (CFD) calculations, strip theory, and open-water tests. In modeling of marine vehicles, the hydrodynamic damping forces and moments are usually studied through conventional towing-tank experiments [16–19] that are expensive. As the computation technology advances, the computational fluid dynamics (CFD) method become important as a less expensive alternative [17, 20–24]. In this dissertation, we combine both the CFD method and field tests to study the hydrodynamic damping terms to build a practical motion model for the EcoMapper with low cost. The complete model is validated by field experiments carried out in the west pond in the Georgia Tech Savannah Campus.

Based on a generic six-degree-of-freedom (6DOF) dynamic model for the motions of a single underwater vehicle, we investigate the collective dynamics of a group of AUVs and introduce a decoupled design procedure so that formation controllers designed for particle dynamics can be generalized to formation controllers for fully-actuated AUVs with 6DOF motions in three-dimensional (3D) space. Formation control of multiple AUVs has received much attention [2, 4–6, 9, 25–28] due to potentially broad applications in ocean engineering and science. Despite of the complex AUV dynamics, many previous results on formation control are developed assuming each agent is modeled

by either a single or a double integrator particle model [29–31]. Each agent cooperates with others based on information shared through communication links that introduce time delays in the system dynamics. [32–35] analyzed the stability of a multi-agent system with time delays. [36] proved the input-to-state stability of a single-master-N-slave structured agent group under time delays, assuming double-integrator dynamics for each agent. Formation control becomes more challenging if more practical and complex dynamics and communication constraints are concerned. These challenges are crucial for AUVs because of their complex and coupled dynamics. In this dissertation, we study the collective dynamics of multi-AUV system based on a complete 6DOF dynamic motion model of a single underwater vehicle. Following a standard inner-outer loop approach, we first decouple the orientation and translation controls, and then use a geometric approach to separate the translation dynamics into formation shape dynamics and formation center dynamics. The coupling terms of the two portions of the dynamics are treated as perturbations and tolerated by a robust formation-keeping controller. The controller is also robust to constant bounded time delays. Compared to other existing approaches with similar goals, this decoupling procedure simplifies the entire design process. To justify the effectiveness of this method, we present both rigorous theoretical analysis and simulation results.

As cooperative control is often needed for interaction between an human and autonomous agents [37–48], we extend the results of formation control of a multi-AUV system to obtain cooperative control of a human-AUV system. In the human-AUV system we investigate, vehicles do not share information through communication links, which is different from most results on formation control in the literature [49–55]. Every vehicle measures the positions, velocities, and accelerations of all other vehicles within its own coordinate system. We analyze situations when vehicles in the system need to position themselves into a desired formation shape and track a desired trajectory as a team and design controllers for AUVs that enable the system to achieve it. To predict the motions of an HUV, inspired by the results in human's emotional arousal [38–40] during human-robot interaction, we build a dynamic model, taking the attention of the human operator of an HUV into account. We decouple the collective dynamics of all vehicles into formation-center, formation-rotation, and formation-shape dynamics, and then design controllers to drive the formation center trajectory and the formation shape to desired values. For the formation center, which can be viewed as a free

particle after collective dynamics decomposition, we design a curve-tracking controller that enables a free particle to track any smooth curve in 3D space. This enriches the results on curve tracking problem, as results reported in the literature considers only 2D smooth curves, including smooth planar curves ([56,57]) and smooth curves constrained to a sphere surface ([58]). For the formation shape, we also design a controller that drives it to a desired formation shape. The formation shape controller design takes into account the constraints of both human operator's attention and shape variables to prevent a human operator from being scared and vehicles from colliding, respectively. Our method calculates control forces based on sensor measurements so that vehicles do not need to communicate and they have freedom to choose their own coordinate system. We illustrate a system comprised of one HUV and two AUVs and present simulation results that demonstrate the effectiveness of our method.

Because autonomous surveys are especially attractive in situations where the marine environment is less than ideal for human-based methods, in this dissertation, we develop novel methods for performing marine environmental surveys using a fleet of heterogeneous autonomous robotic vehicles and present mapping algorithms that reconcile data from heterogeneous marine vehicles on multiple different paths and create a high-fidelity visual representation of the survey data. Although path following for marine vehicles is very important, and, therefore, has been widely studied [59–65], yet the performance of only few theoretical results has been evaluated and reported in field tests [66–68], so we are motivated to evaluate the controllers on marine robots for environmental surveys. In this dissertation, to demonstrate the effectiveness of our survey methods, we performed a twenty-one-day survey in July, 2011, for a coastal lagoon in Grand Isle, Louisiana, where heavy pollution had been reported during the DeepWater Horizon oil spill, and large scale cleaning efforts have been performed after the spill was contained. Using a fleet of heterogeneous marine vehicles, we evaluate the performance of our controllers, analyze the bathymetric and crud-oil data collected by our autonomous vehicles in survey missions, and create bathymetric and oil-concentration maps. The experimental results show the effectiveness of our method and also provide guidance for mission design of further autonomous environmental surveys.

The reminder of the dissertation is organized as follows. In Chapter 2, using the EcoMapper as an example, we present the dynamic modeling method for underwater vehicles. In Chapter 3, we

develop a decoupled controller design approach for formation control of multi-AUV system with time delays. We extend this approach and present a cooperative control design for human-AUV system in Chapter 4. In Chapter 5, we present the method of analyzing the survey data collected by marine vehicles during environmental surveys. Chapter 6 gives the conclusion and future work.

DYNAMIC MODELING OF AN AUTONOMOUS UNDERWATER VEHICLE: THE ECOMAPPER

The YSI EcoMapper autonomous underwater vehicle (Figure 1) has a large sensor payload, a small size for rapid deployment by one person, and intuitive mission planning software. It is widely used in environmental mapping [12–15]. Using the remote helm functionality of the EcoMapper, users can take full control of the vehicle [69]. To precisely control the EcoMapper, a dynamic model is needed. However, to the best of our knowledge, no dynamic model of the EcoMapper has been reported in the literature. Our goal is to develop a mathematical dynamic model of the EcoMapper to serve the purpose of simulation and real-time control.

Figure 1: EcoMapper on dock

As shown in Figure 1, the main body of the EcoMapper is a slender cylinder with two horizontal fins for pitch angle control, and two vertical fins for yaw angle control. The thrust of the EcoMapper is generated by a two-blade propeller. To build the dynamic model, we need to identify the rigid-body inertia matrix, the rigid-body Coriolis and centripetal matrix, the hydrodynamic added inertia matrix, the hydrodynamic added Coriolis and centripetal matrix, the hydrodynamic damping terms, and the propeller coefficient. We determine the rigid-body inertia matrix and the

rigid-body Coriolis and centripetal matrix using the software "Solidworks" and calculate the hydrodynamic added inertia matrix and the hydrodynamic added Coriolis and centripetal matrix using strip theory [70, 71].

The hydrodynamic damping forces and moments are usually studied through conventional towing-tank experiments [16–18]. For example in [19], tow-tank experiments were used to build and verify a motion model of REMUS (an autonomous underwater vehicles developed by von Alt and associates at the Oceanographic Systems Laboratory at the Woods Hole Oceanographic Institution [72]). However, tow-tank experiments are expensive. As the computation technology advances, the computational fluid dynamics (CFD) method become important as a less expensive alternative [17, 20–24]. For example in [73], CFD simulations were used to estimate hydrodynamic coefficients of TUNA-SAND (an remotely operated underwater vehicle developed by URA Laboratory, The University of Tokyo [74]). In this chapter, we combine both the CFD method and field tests to study the hydrodynamic damping terms to build a practical motion model for the EcoMapper with low cost. We split the hydrodynamic damping forces and moments into two parts. The first part is controllable by vertical and horizontal fins. We explicitly derive the form of this part, i.e., fin-related hydrodynamic damping terms, then CFD simulations are used to identify parameters in this part. Different from CFD experiments for TUNA-SAND in [73] and RRC ROV in [24], which have no control surfaces, CFD experiments for the EcoMapper are performed at a range of fin angles, so that the CFD experiments results can provide enough data to identify parameters in the relationships between hydrodynamic damping terms and fin angles. One advantage of this approach is that the analytical analysis of the geometry of control surfaces (e.g., Chapter 5 in [75]) can be avoided. This approach also applies to other underwater vehicles with control surfaces, for example, the Yellowfin (an autonomous underwater vehicle developed at the Georgia Tech Research Institute [76, 77]). After the fin-related hydrodynamic damping terms are identified, the remaining hydrodynamic damping dynamics are identified through a field test. Finally, we carry out open-water experiments to obtain the thruster coefficient to complete the dynamic model. The complete model is validated by field experiments carried out in the west pond in the Georgia Tech Savannah Campus.

The remainder of this chapter is organized as follows. Section 2.1 gives the six-degree-of-freedom model for the EcoMapper. Section 2.2 explains the detailed procedure to calculate the parameters in this model and the parameter identification results. Section 2.3 provides field experiment results that validate the proposed model. Conclusions are presented in Section 2.4.

2.1 Derivation of the Dynamic Model for Underwater Vehicles

We apply a six-degree-of-freedom motion model [70] to the EcoMapper to describe its surge, sway, heave, roll, pitch, and yaw motions. We set the origin of the body-fixed frame of the EcoMapper at its center of buoyancy. Figure 2 shows the body-fixed frame, earth-fixed frame, and the elementary motions of an AUV. Because the EcoMapper is usually ballasted to neutral buoyancy in water without control, which means that the sum of the buoyancy force and the gravitational force is zero, there is no restoring force acting on the EcoMapper along "upright" direction. The EcoMapper has a bottom-heavy design so the restoring moment on roll will keep the roll angle stabilized around 0, therefore, we do not consider the control for roll moments. To simplify the problem, we assume the EcoMapper homogeneous and completely immersed in water, so the center of buoyancy and the center of gravity coincide. We can also infer that the density of the EcoMapper is the same as the density of the surrounding fluid (fresh water in this paper). We define $\eta_1 = [x,y,z]^T$, $\eta_2 = [\phi,\theta,\psi]^T$, $\eta = [\eta_1^T, \eta_2^T]^T$, $v_1 = [u,v,w]^T$, $v_2 = [p,q,r]^T$, $v = [v_1^T, v_2^T]^T$, $\tau_1 = [\tau_x, \tau_y, \tau_z]^T$, $\tau_2 = [\tau_\phi, \tau_\theta, \tau_\psi]^T$, $\tau_{hydr} = [\tau_1^T, \tau_2^T]^T$, and $\tau_{thrust} = [f_{thrust}, 0, 0, 0, 0, 0]^T$, where $[x,y,z]^T$ represents the EcoMapper position in the earth-fixed frame; $[\phi, \theta, \psi]^T$ represents the Euler angle vector for roll, pitch, and yaw of the EcoMapper in the earth-fixed frame; $[u,v,w]^T$ represents the body-fixed linear velocity vector for surge, sway, and heave; $[p,q,r]^T$ represents the body-fixed angular speed vector for roll, pitch, and yaw; τ_x, τ_y, τ_z are fin-related hydrodynamic damping forces along x, y, and z directions, respectively; $\tau_\phi, \tau_\theta, \tau_\psi$ are fin-related hydrodynamic damping moments along x, y, and z directions, respectively; f_{thrust} is the thrust force of the propeller, which is along x direction. The dynamics of the EcoMapper can be expressed by the following two equations ([70], [71]):

$$\dot{\eta} = J(\eta_2)v, \tag{1}$$

$$M\dot{v} + C(v)v + Dv = \tau_{hydr} + \tau_{thrust}. \tag{2}$$

Detailed derivations of Equations (1) and (2) are provided in Appendix A.
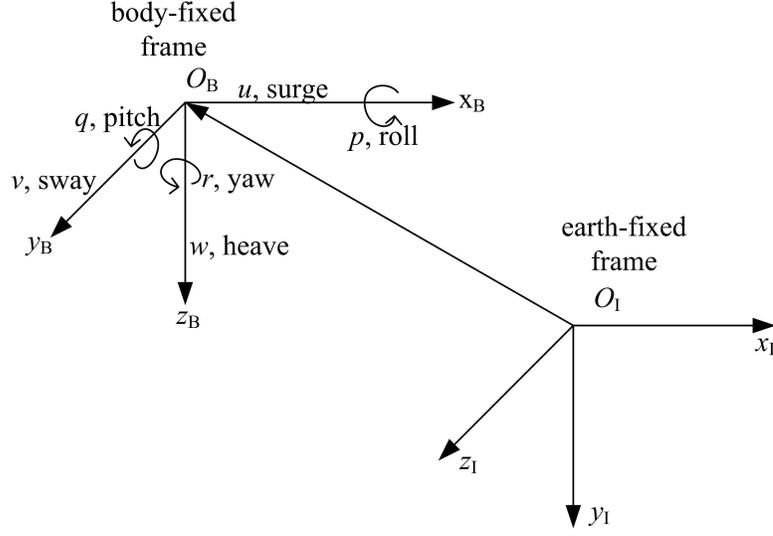
Figure 2: Frames and elementary vehicle motions

In Equation (1), $J(\eta_2)$ is the invertible rotation matrix from the body-fixed frame to the earth-fixed frame:

$$J(\eta_2) = \begin{bmatrix} J_1(\eta_2) & 0_{3\times3} \\ 0_{3\times3} & J_2(\eta_2) \end{bmatrix},$$ (3)

where

$$J_1(\eta_2) = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\ s\psi c\theta & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix}$$

and

$$J_2(\eta_2) = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix}.$$

Here, $s\cdot = \sin(\cdot)$ and $c\cdot = \cos(\cdot)$.

In Equation (2),

$$M = M_{RB} + M_A,$$ (4)

where M_{RB} denotes the rigid-body inertia matrix, and M_A denotes the hydrodynamic added inertia.

9

They are represented by the following two equations:

$$
M_{RB} =
\begin{bmatrix}
m & 0 & 0 & 0 & 0 & 0 \\
0 & m & 0 & 0 & 0 & 0 \\
0 & 0 & m & 0 & 0 & 0 \\
0 & 0 & 0 & I_x & -I_{xy} & -I_{xz} \\
0 & 0 & 0 & -I_{yx} & I_y & -I_{yz} \\
0 & 0 & 0 & -I_{zx} & -I_{zy} & I_z
\end{bmatrix},
\tag{5}
$$

$$
M_A = -
\begin{bmatrix}
X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\
Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\
Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\
K_{\dot{u}} & K_{\dot{v}} & K_{\dot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\
M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\
N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}}
\end{bmatrix},
\tag{6}
$$

where m is the mass of the AUV, $I = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{yx} & I_y & -I_{yz} \\ -I_{zx} & -I_{zy} & I_z \end{bmatrix}$ is the inertia tensor of the EcoMapper in the body-fixed frame, and all terms in M_A are hydrodynamic added mass force coefficients.

In Equation (2),

$$
C(v) = C_{RB}(v) + C_A(v),
\tag{7}
$$

where $C_{RB}(v)$ is the coefficient matrix for rigid-body Coriolis and centripetal terms and $C_A(v)$ is the coefficient matrix for hydrodynamic added Coriolis and centripetal terms. They are expressed in the following two matrices:

$$
C_{RB}(v) =
\begin{bmatrix}
0 & 0 & 0 & 0 & mw & -mv \\
0 & 0 & 0 & -mw & 0 & mu \\
0 & 0 & 0 & mv & -mu & 0 \\
0 & mw & -mv & 0 & -I_{yz}q - I_{xz}p + I_zr & I_{yz}r + I_{xy}p - I_yq \\
-mw & 0 & mu & I_{yz}q + I_{xz}p - I_zr & 0 & -I_{xz}r - I_{xy}q + I_xp \\
mv & -mu & 0 & -I_{yz}r - I_{xy}p + I_yq & I_{xz}r + I_{xy}q - I_xp & 0
\end{bmatrix},
\tag{8}
$$

$$C_A(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & -a_3 & a_2 \\ 0 & 0 & 0 & a_3 & 0 & -a_1 \\ 0 & 0 & 0 & -a_2 & a_1 & 0 \\ 0 & -a_3 & a_2 & 0 & -b_3 & b_2 \\ a_3 & 0 & -a_1 & b_3 & 0 & -b_1 \\ -a_2 & a_1 & 0 & -b_2 & b_1 & 0 \end{bmatrix}, \tag{9}$$

where

$$a_1 = X_{\dot{u}}u + X_{\dot{v}}v + X_{\dot{w}}w + X_{\dot{p}}p + X_{\dot{q}}q + X_{\dot{r}}r,$$

$$a_2 = X_{\dot{v}}u + Y_{\dot{v}}v + Y_{\dot{w}}w + Y_{\dot{p}}p + Y_{\dot{q}}q + Y_{\dot{r}}r,$$

$$a_3 = X_{\dot{w}}u + Y_{\dot{w}}v + Z_{\dot{w}}w + Z_{\dot{p}}p + Z_{\dot{q}}q + Z_{\dot{r}}r,$$

$$b_1 = X_{\dot{p}}u + Y_{\dot{p}}v + Z_{\dot{p}}w + K_{\dot{p}}p + K_{\dot{q}}q + K_{\dot{r}}r,$$

$$b_2 = X_{\dot{q}}u + Y_{\dot{q}}v + Z_{\dot{q}}w + K_{\dot{q}}p + M_{\dot{q}}q + M_{\dot{r}}r,$$

$$b_3 = X_{\dot{r}}u + Y_{\dot{r}}v + Z_{\dot{r}}w + K_{\dot{r}}p + M_{\dot{r}}q + N_{\dot{r}}r.$$

In the proposed model, we split the hydrodynamic damping forces and moments into two parts. The first part is τ_{hydr} in Equation (2), which is controllable by vertical and horizontal fins. The remaining part is Dv in Equation (2), where D is a damping matrix. As the speed of EcoMapper is usually below 2 m/s, which is relatively low, so we consider only linear damping terms in Dv, and neglect the coupling dissipative terms. Therefore, we define

$$D = -\text{diag}\{X_u, Y_v, Z_w, K_p, M_q, N_r\}, \tag{10}$$

where $X_u, Y_v, Z_w, K_p, M_q, N_r$ are all negative scalar coefficients.

In Equation (2), τ_{hydr} includes the fin-related hydrodynamic damping terms, which are functions of the fin angles and the vehicle speed. During operations of the EcoMapper, the two horizontal fins keep the same angle, denoted by α, and the two vertical fins have the same angle, denoted by β. Because the horizontal fins of the EcoMapper are always symmetric about $x - z$ plane, their corresponding hydrodynamic forces on right and left side of the EcoMapper produced by the flow along x direction have the same magnitude and opposite directions, therefore, they do not affect the total hydrodynamic forces along y direction, i.e., τ_y does not depend on α. Similarly, τ_ϕ, and τ_ψ do

not depend on α. As the vertical fins of the EcoMapper are symmetric about $x - y$ plane, so they do not generate moments around x and y axes and forces along z axis, therefore τ_z, τ_ϕ, and τ_θ do not depend on β. As the EcoMapper is designed to be self-stabilized in roll angle, we assume $\phi = 0$ and

$$\tau_\phi = 0 \tag{11}$$

hold all the time. We also assume the velocity of the EcoMapper is parallel with the vehicle body. Now we summarize the relationship between the fin-related hydrodynamic damping terms and the fin angles into the following equations:

$$\tau_x = \tau_x(u, \alpha, \beta), \ \tau_y = \tau_y(u, \beta), \ \tau_z = \tau_z(u, \alpha),$$

$$\tau_\phi = 0, \ \tau_\theta = \tau_\theta(u, \alpha), \ \tau_\psi = \tau_\psi(u, \beta).$$

We know that when the EcoMapper is still in water, i.e., when $u = 0$, all the hydrodynamic terms are zero, i.e.,

$$\tau_x(0, \alpha, \beta) = \tau_y(0, \beta) = \tau_z(0, \alpha) = \tau_\theta(0, \alpha) = \tau_\psi(0, \beta) = 0.$$

As the EcoMapper is symmetric about $x - z$ plane without considering vertical fins, and the vertical fins are symmetric about $x - y$ plane, it is easy to derive the following properties from physics.

$$\tau_x(u, \alpha, -\beta) = \tau_x(u, \alpha, \beta), \tag{12}$$

$$\tau_y(u, -\beta) = -\tau_y(u, \beta), \tag{13}$$

$$\tau_\psi(u, -\beta) = -\tau_\psi(u, \beta). \tag{14}$$

As α-degree horizontal fins and $-\alpha$-degree horizontal fins are symmetric about $x - y$ plane, we get

$$\tau_z(u, -\alpha) - \tau_z(u, 0) = -[\tau_z(u, \alpha) - \tau_z(u, 0)], \tag{15}$$

$$\tau_\theta(u, -\alpha) - \tau_\theta(u, 0) = -[\tau_\theta(u, \alpha) - \tau_\theta(u, 0)], \tag{16}$$

$$\tau_x(u, -\alpha, \beta) \approx \tau_x(u, \alpha, \beta). \tag{17}$$

Apply the forth-order Maclaurin's expansion to τ_y, we obtain

$$\tau_y = \Phi_\beta^T b_{\tau_y} + \Phi_{\beta 1}^T b'_{\tau_y} + \Phi_{\beta 2}^T b''_{\tau_y}, \tag{18}$$

12

where $\Phi_{\beta 1} = [1, \beta, \beta^2, \beta^3, \beta^4]^T$, $\Phi_{\beta 2} = [u, \beta^2 u, u^2, \beta^2 u^2, u^3, u^4]^T$, and

$$\Phi_\beta = [\beta u, \beta^3 u, \beta u^2, \beta u^3]^T. \tag{19}$$

b_{τ_y}, b'_{τ_y} and b''_{τ_y} are coefficient vectors. From $\tau_y(0, \beta) = 0$, we obtain $\Phi_{\beta 1}^T b'_{\tau_y} = 0$, and then from Equation (13), we get $\Phi_{\beta 2}^T b''_{\tau_y} = 0$. Therefore, Equation (18) reduces to

$$\tau_y = \Phi_\beta^T b_{\tau_y}. \tag{20}$$

Similarly, we can get

$$\tau_\psi = \Phi_\beta^T b_{\tau_\psi}, \tag{21}$$

where b_{τ_ψ} is a coefficient vector.

Apply the forth-order Maclaurin's expansion to τ_z, we get

$$\tau_z = \Phi_\alpha^T b_{\tau_z} + \Phi_{\alpha 1}^T b'_{\tau_z} + \Phi_{\alpha 2}^T b''_{\tau_z}, \tag{22}$$

where $\Phi_{\alpha 1} = [1, \alpha, \alpha^2, \alpha^3, \alpha^4]^T$, $\Phi_{\beta 2} = [\alpha^2 u, \alpha^2 u^2]^T$, and

$$\Phi_\alpha = [u, \alpha u, \alpha^3 u, u^2, \alpha u^2, u^3, \alpha u^3, u^4]^T. \tag{23}$$

b_{τ_z}, b'_{τ_z} and b''_{τ_z} are coefficient vectors. From $\tau_z(0, \alpha) = 0$, we obtain $\Phi_{\alpha 1}^T b'_{b_{\tau_z}} = 0$, and then from Equation (15), we get $\Phi_{\alpha 2}^T b''_{\tau_z} = 0$. Therefore, Equation (22) reduces to

$$\tau_z = \Phi_\alpha^T b_{\tau_z}. \tag{24}$$

Similarly, we can get

$$\tau_\theta = \Phi_\alpha^T b_{\tau_\theta}, \tag{25}$$

where τ_θ is a coefficient vector.

Now we apply the forth-order Maclaurin's expansion to τ_x and get

$$\tau_x = \Phi_1^T b_1 + \Phi_2^T b_2 + \Phi_3^T b_3 + \Phi_{\alpha\beta}^T b_{\tau_x}, \tag{26}$$

where $\Phi_1 = [1, \alpha, \alpha^2, \alpha^3, \alpha^4, \beta, \beta^2, \beta^3, \beta^4]^T$, $\Phi_2 = [\alpha u, \alpha u^2, \alpha u^3, \alpha^3 u]^T$, $\Phi_3 = [\beta u, \beta u^2, \beta u^3, \beta^3 u]^T$, and $\Phi_{\alpha\beta} = [u, u^2, u^3, \alpha^2 u, \beta^2 u, u^4, \alpha^2 u^2, \beta^2 u^2]^T$. b_1, b_2, b_3, and b_{τ_x} are coefficient vectors. From

$f_1(0, \alpha, \beta) = 0$, we get $\Phi_1^T b_1 = 0$, then by plugging Equation (12) into Equation (26), we obtain $\Phi_2^T b_2 = 0$. In addition, Equation (17) leads to $\Phi_3^T b_3 = 0$. Therefore Equation (26) reduces to

$$\tau_x = \Phi_{\alpha\beta}^T b_{\tau_x}, \tag{27}$$

Now the fin-related hydrodynamic damping terms are fully expressed by Equations (11), (20), (21), (24), (25) and (27), where b_{τ_y}, b_{τ_ψ}, b_{τ_z}, b_{τ_θ}, and b_{τ_x} are hydrodynamic damping coefficients.

In Equation (2), $\tau_{thrust} = [f_{thrust}, 0, 0, 0, 0, 0]^T$ and f_{thrust} is the thrust provided by the propeller. It is a function of the propeller diameter, which is fixed, the density and viscosity of water, which are assumed to be constant, and propeller rotation speed, i.e., revolutions per unit time, denoted by n. Now f_{thrust} can be described by the following equation:

$$f_{thrust} \approx cn^2, \tag{28}$$

where c is the propeller coefficient.

2.2 Parameter Identification

This section provides the procedure to calculate M_{RB}, M_A, C_{RB}, C_A, hydrodynamic damping coefficients b_{τ_x}, b_{τ_y}, b_{τ_z}, b_{τ_θ}, and b_{τ_ψ}, and propeller coefficient c for the dynamic model of the EcoMapper.

2.2.1 M_{RB} and C_{RB}

We use "Solidworks", a three-dimensional mechanical computer-aided-design software, to calculate inertia matrix M_{RB} for the EcoMapper. In Solidworks, we draw the geometry of the EcoMapper as shown in Figure 3 and use the "mass properties" functionality of Solidworks to calculate inertia matrix M_{RB}. The Solidworks geometry file of the Ecoampper will be further used as the database file for grid generating for CFD calculations.

The mass m and density ρ of the EcoMapper are as follows:

$$m = 27.2 \quad \text{kg}, \tag{29}$$

$$\rho = 1000 \quad \text{kg/m}^3. \tag{30}$$

According to the geometry of the EcoMapper, we calculate the inertia tensor I and get $I_x = 0.0743$,
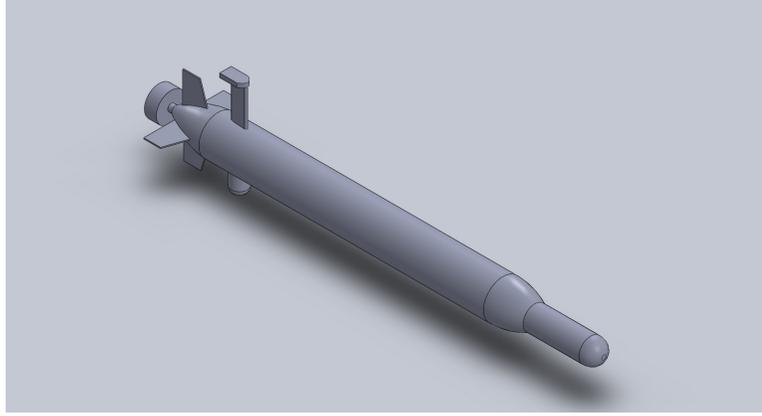
Figure 3: Solidworks model of the EcoMapper

$I_y = 4.723$, $I_z = 4.7159$, $I_{xz} = I_{zx} = 0.0011$, $I_{xy} = I_{yx} = I_{yz} = I_{zy} = 0$, with all units N·m²·s². There-

fore $I = \begin{bmatrix} 0.0743 & 0 & -0.0011 \\ 0 & 4.723 & 0 \\ -0.0011 & 0 & 4.7159 \end{bmatrix}$. I is obviously diagonally dominant. The absolute val-

ues of off-diagonal elements are all less than 1.5% of the smallest absolute values of diagonal

elements. Therefore, without causing many errors, we can approximate I to a diagonal matrix

$I \approx \text{diag}\{0.0743, 4.723, 4.7159\}$. As a result,

$$I_x = 0.0743, \; I_y = 4.723, \; I_z = 4.7159,$$
$$I_{xz} = I_{zx} = I_{xy} = I_{yx} = I_{yz} = I_{zy} = 0. \tag{31}$$

This approximation makes sense because the main body of the EcoMapper is a slender cylinder

with three planes of symmetry and a rigid body with three planes of symmetry has a diagonal inertia

matrix. Now Equations (29) and (31) specifies all the parameter for M_{RB} and C_{RB} in Equations (5)

and (8), so we can get

$$M_{RB} = \text{diag}\{27.2, 27.2, 27.2, 0.0743, 4.723, 4.7159\}, \tag{32}$$

$$C_{RB}(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & 27.2w & -27.2v \\ 0 & 0 & 0 & -27.2w & 0 & -27.2u \\ 0 & 0 & 0 & 27.2v & -27.2u & 0 \\ 0 & 27.2w & -27.2v & 0 & 4.7159r & -4.723q \\ -27.2w & 0 & 27.2u & -4.7159r & 0 & 0.0743p \\ 27.2v & -27.2u & 0 & 4.723q & -0.0743p & 0 \end{bmatrix}. \tag{33}$$

15

2.2.2 M_A and C_A

The nose of the EcoMapper is a light plastic cylindrical shell, and most spaces in the nose are empty, so we neglect the mass of the nose when calculate M_A and C_A. We also neglect the four fins, and treat the EcoMapper as a cylindrical rigid body to simplify the calculation for M_A and C_A. We can see this approximation is rational from the fact that M_{RB} is approximately a diagonal matrix.

According to the strip theory [70], [71] provides the following formulas for all non-zero hydrodynamic added mass force coefficients for a cylindrical rigid body with a mass m, a length L, and a radius of the circular section r, assuming it is immersed in a fluid with density ρ.

$$X_{\dot{u}} = -0.1m, \ Y_{\dot{v}} = -\pi \rho r^2 L, \ Z_{\dot{w}} = -\pi \rho r^2 L, \ M_{\dot{q}} = -\frac{1}{12}\pi \rho r^2 L^3, \ N_{\dot{r}} = -\frac{1}{12}\pi \rho r^2 L^3. \tag{34}$$

For the EcoMapper, $L = 1.4$ m, $r = 0.0736$ m, the fluid is water with density $\rho_{water} = 1000$ kg/m^3, as a result, we obtain all non-zero hydrodynamic added mass force coefficients as follows:

$$X_{\dot{u}} = -2.72, \ Y_{\dot{v}} = -23.8250, \ Z_{\dot{w}} = -23.8250, \ M_{\dot{q}} = -3.8914, \ N_{\dot{r}} = -3.8914. \tag{35}$$

Now Equations (35) specifies all non-zero parameters to calculate M_A and C_A from Equations (6) and (9), so we get

$$M_A = \text{diag}\{2.72, 23.8250, 23.8250, 0, 3.8914, 3.8914\}, \tag{36}$$

$$C_A(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & 23.8250w & -23.8250v \\ 0 & 0 & 0 & -23.8250w & 0 & 2.72u \\ 0 & 0 & 0 & 23.8250v & -2.72u & 0 \\ 0 & 23.8250w & -23.8250v & 0 & 3.8914r & -3.8914q \\ -23.8250w & 0 & 2.72u & -3.8914r & 0 & 0 \\ 23.8250v & -2.72u & 0 & 3.8914q & 0 & 0 \end{bmatrix}. \tag{37}$$

2.2.3 τ_{hydr}

To get enough data to study the hydrodynamic damping coefficients, we run CFD simulations for the EcoMapper at four speeds, that is, 0.25 m/s, 0.5 m/s, 0.75 m/s, and 1 m/s, as the maximum speed of the EcoMapper is designed around 1 m/s. For each speed, we change the vertical-fin

angle from $0°$ to $35°$ and horizontal-fin angle from $-35°$ to $35°$, with a step size $5°$, and calculate

the corresponding hydrodynamic damping forces and moments in the CFD software "Ansys-CFX".

Therefore, CFD calculations are carried out at 88 data points. For each specified fin angle, we first

draw the corresponding EcoMapper geometry in Solidworks, then using the Solidworks geometry

file as the database file, we generate a mesh file for the EcoMapper in "Gridgen", a mesh generator,

and import it into Ansys-CFX. In CFX, we specify the EcoMapper speed in the boundary condition

and calculate the hydrodynamic damping forces and moments.

To calculate the hydrodynamic damping terms corresponding to a specified EcoMapper speed,

which is along the x direction in its body-fixed frame, we set the EcoMapper static and set the

fluid flowing along the "$-x$" direction in CFD simulation, as the hydrodynamic damping forces

and moments depend only on the relative motion between the EcoMapper and the fluid. Therefore,

we can adopt fixed mesh generating for CFD calculation, instead of using the relatively complex

dynamic meshes.

To study the hydrodynamic damping terms on the EcoMapper in an unbounded fluid domain,

we need to use a large constant-speed flow field. In this chapter, we define a fluid domain, the

length, width, and height of which are five times of the length, width, and height of the EcoMapper,

respectively, as shown in Figure 4. In this fluid domain, we draw a small box to enclose the area

that is close to the EcoMapper, and generate fine block grids within this area. In the area far from

the EcoMapper, we use relatively coarser block grids to reduce the CFD computation time (see

Figure 5). To build block grids, we first generate surface grids on all boundaries, including the

surface of the EcoMapper and the boundary of the fluid domain. On the outer boundary of the fluid

domain, which consists of six rectangles, we generate a "structured domain", and on the surface of

the EcoMapper, some parts of which are complex, we generate an "unstructured domain", shown in

Figure 6, then we generate "unstructured block" based on those "domains". In Gridgen, "domains"

mean surface grids and "blocks" mean block grids.

The block grids generated in Gridgen are imported into CFX for CFD calculation. In CFX, we

set water as the fluid, set the inlet and outlet of the flow as Figure 7, and carry out single-phase

steady-state simulations using shear-stress-transport model. Figure 8 shows one of the CFX simula-

tion results, in which the fluid flow speed is set to 1m/s, the horizontal-fin angle of the EcoMapper
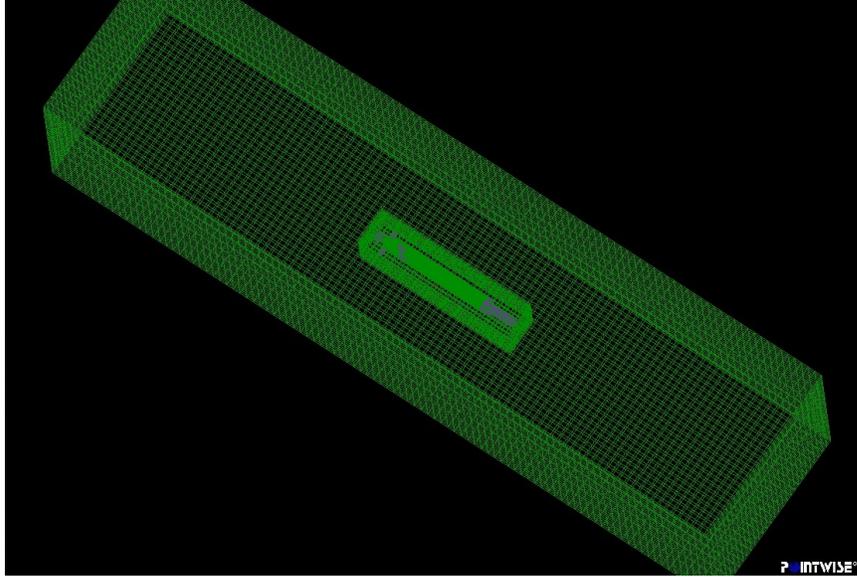
Figure 4: Fluid domain
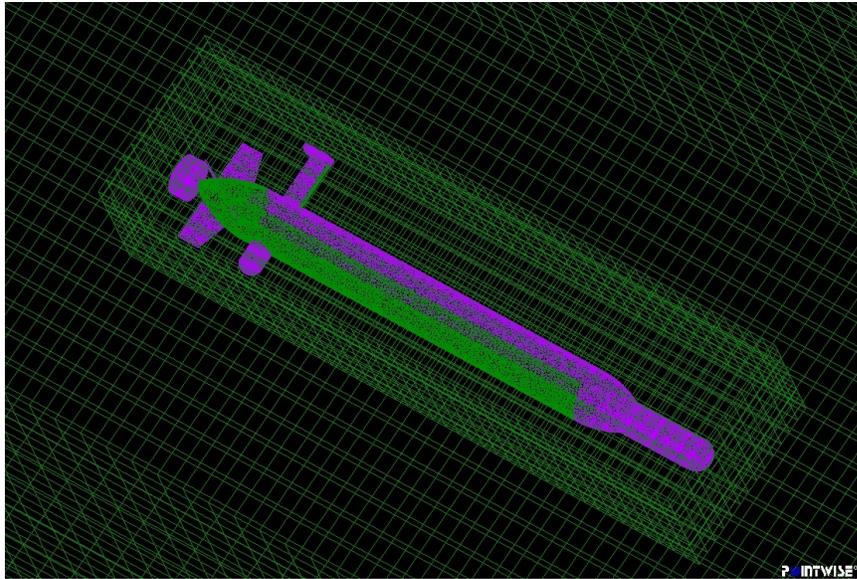
Figure 5: Fine and coarse block grids

to $0°$, and the vertical-fin angle to $30°$. We list simulation results for all EcoMapper speeds and fin angles in Tables 1, 2, 3, and 4. From data in Tables 1 and 2 in which $\beta = 0$, we can see that τ_y, τ_ϕ, and τ_ψ do not change with α, and the average values of τ_y, τ_ϕ, and τ_ψ are all close to zero. From data in Tables 3 and 4 in which $\alpha = 0$, we can see that τ_z, τ_ϕ, and τ_θ are all small and do not change with β. From data in Table 1, we observe that $f_1(u, -\alpha, \beta) \approx f_1(u, \alpha, \beta)$. These facts all agree with our assumption in Section 2.1.
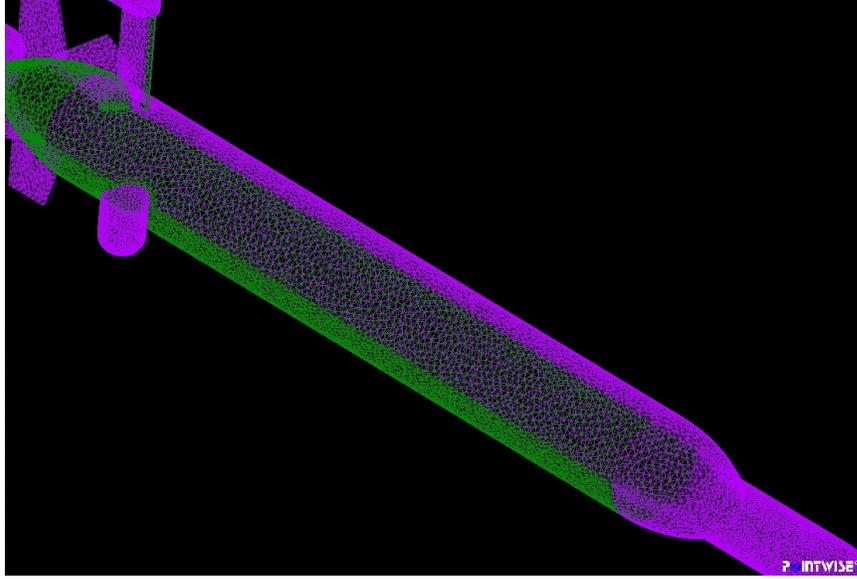
Figure 6: Unstructured domain on the EcoMapper surface
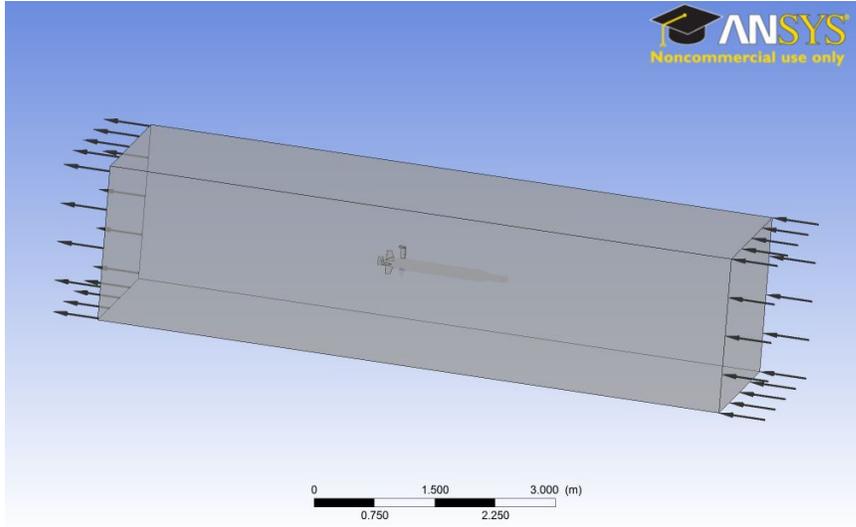
Figure 7: Inlet and outlet boundary condition in CFX

Now we use the least-mean-square method to estimate parameters b_{τ_x}, b_{τ_y}, b_{τ_z}, b_{τ_θ}, and b_{τ_ψ} in Equations (27), (20), (24), (21), and (25). We use τ_y as an example to explain the procedure, using the $8 \times 4 = 32$ data points provided in Table 3. For each point $(\tau_{y,i}, \beta_i, u_i)$, where $i = 1, 2 \cdots 32$, we calculate the corresponding $\Phi_{\beta,i}$ according to Equation (19). Define $Y = [\tau_{y,1}, \tau_{y,2} \cdots \tau_{y,N}]^T$ and $A = [\Phi_{\beta,1}, \Phi_{\beta,2} \cdots \Phi_{\beta,N}]^T$, the least-mean-square estimation of b_{τ_y} is

$$b_{\tau_y} = (A^T A)^{-1} A^T Y. \tag{38}$$
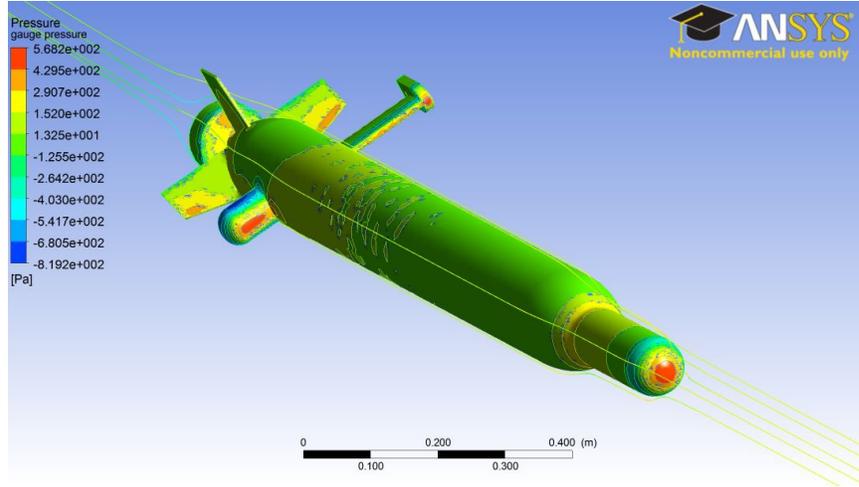
Figure 8: Stream line and gauge pressure contour in CFX

Table 1: $[\tau_x, \tau_y, \tau_z]$ VS horizontal-fin angle α and the EcoMapper speed

α	u=1m/s	u=0.75m/s	u=0.5m/s	u=0.25m/s
35	[-10.9907 0.3151 -7.8490]	[-6.2307 0.2211 -4.3690]	[-2.7947 0.0721 -1.9100]	[-0.7132 0.0161 -0.4839]
30	[-10.1951 0.3323 -7.5768]	[-5.7736 0.1829 -4.1911]	[-2.6082 0.1547 -1.8784]	[-0.6640 0.0355 -0.4742]
25	[-9.3262 -0.2538 -7.4454]	[-5.3017 -0.1216 -4.1838]	[-2.3936 -0.0403 -1.8569]	[-0.6123 -0.0067 -0.4659]
20	[-8.3563 0.0193 -7.0843]	[-4.7641 0.0090 -3.9594]	[-2.1491 0.0310 -1.7387]	[-0.5484 0.0046 -0.4306]
15	[-7.5640 -0.1620 -6.0628]	[-4.3034 -0.0333 -3.3790]	[-1.9457 0.0053 -1.4877]	[-0.4976 0.0034 -0.3670]
10	[-7.0246 0.5152 -4.7866]	[-3.9966 0.2610 -2.6683]	[-1.8010 0.0781 -1.1789]	[-0.4628 0.0296 -0.2932]
5	[-6.7013 -0.6192 -2.5693]	[-3.8337 -0.3044 -1.4390]	[-1.7439 -0.1142 -0.6344]	[-0.4482 -0.0241 -0.1555]
0	[-6.4592 -0.3385 -0.1455]	[-3.6968 -0.1536 -0.0847]	[-1.6850 -0.0605 -0.0376]	[-0.4344 -0.0145 -0.0069]
-5	[-6.5714 -0.5048 2.4216]	[-3.7273 -0.1481 1.3211]	[-1.6842 -0.0210 0.5693]	[-0.4349 -0.0058 0.1420]
-10	[-6.7161 -0.0012 4.4858]	[-3.8156 0.0846 2.4496]	[-1.7431 0.0169 1.0865]	[-0.4506 -0.0035 0.2713]
-15	[-7.1923 -0.1177 6.1951]	[-4.1091 -0.0172 3.4732]	[-1.8685 0.0124 1.5326]	[-0.4806 0.0027 0.3799]
-20	[-8.0637 0.9691 6.9083]	[-4.5735 0.4642 3.8975]	[-2.0620 0.1770 1.7356]	[-0.5276 0.0337 0.4375]
-25	[-8.9998 -1.2318 7.3026]	[-5.1127 -0.6601 4.1338]	[-2.3047 -0.2664 1.8593]	[-0.5877 -0.0528 0.4748]
-30	[-9.7131 -0.8087 7.5272]	[-5.5239 -0.4435 4.2570]	[-2.4731 -0.0952 1.9904]	[-0.6384 -0.0426 0.4842]
-35	[-11.0082 -0.9197 7.8250]	[-6.2213 -0.5140 4.4008]	[-2.7833 -0.2209 1.9572]	[-0.7054 -0.0488 0.4934]

Plugging the data in Table 3 into Equation (38), we get parameter b_{τ_y}, which is listed in Table 5. Using the same procedure, we get b_{τ_x}, b_{τ_y}, b_{τ_z}, b_{τ_θ}, and b_{τ_ψ}, all of which are listed in Table 5. Now we get all parameters for functions in Equations (20), (21), (24), (25), and (27). We plot function τ_z in Equation (24) and the original data points in Figure 9 to illustrate the accordance between original data and the estimation. Other functions and their corresponding original data points are provided in Figures 40, 41, 42, 43, and 44 in Appendix B.

Table 2: $[\tau_\phi, \tau_\theta, \tau_\psi]$ VS horizontal-fin angle α and the EcoMapper speed

α	u=1m/s	u=0.75m/s	u=0.5m/s	u=0.25m/s
35	[-0.0113 -4.1971 0.1587]	[-0.0076 -2.3353 0.1054]	[-0.0020 -1.0192 0.0331]	[-0.0005 -0.2582 0.0074]
30	[-0.0069 -4.0580 0.1271]	[-0.0049 -2.2415 0.0725]	[-0.0033 -1.0044 0.0687]	[-0.0008 -0.2535 0.0163]
25	[0.0149 -3.9498 -0.1184]	[0.0083 -2.2187 -0.0565]	[0.0035 -0.9845 -0.0184]	[0.0007 -0.2473 -0.0029]
20	[0.0134 -3.7315 0.0152]	[0.0076 -2.0857 0.0054]	[0.0030 -0.9170 0.0164]	[0.0006 -0.2275 0.0030]
15	[0.0057 -3.1570 -0.0905]	[0.0022 -1.7609 -0.0210]	[0.0001 -0.7751 0.0008]	[-0.0001 -0.1916 0.0015]
10	[-0.0108 -2.4687 0.2668]	[-0.0048 -1.3750 0.1344]	[-0.0021 -0.6061 0.0404]	[-0.0007 -0.1511 0.0135]
5	[0.0140 -1.2764 -0.2536]	[0.0074 -0.7140 -0.1239]	[0.0028 -0.3145 -0.0460]	[0.0007 -0.0773 -0.0097]
0	[0.0071 0.0510 -0.1526]	[0.0037 0.0269 -0.0692]	[0.0014 0.0116 -0.0283]	[0.0002 0.0040 -0.0079]
-5	[0.0168 1.4386 -0.2422]	[0.0075 0.7875 -0.0691]	[0.0024 0.3399 -0.0092]	[0.0007 0.0845 -0.0022]
-10	[0.0068 2.5589 -0.0312]	[0.0014 1.3982 0.0225]	[0.0010 0.6202 0.0028]	[0.0003 0.1548 -0.0023]
-15	[0.0109 3.4891 -0.0690]	[0.0058 1.9561 -0.0153]	[0.0023 0.8631 0.0026]	[0.0004 0.2137 0.0005]
-20	[-0.0436 3.8862 0.4534]	[-0.0193 2.1943 0.2118]	[-0.0068 0.9771 0.0788]	[-0.0013 0.2458 0.0148]
-25	[0.0397 4.0897 -0.6343]	[0.0201 2.3139 -0.3398]	[0.0073 1.0399 -0.1368]	[0.0012 0.2650 -0.0269]
-30	[0.0284 4.2146 -0.4570]	[0.0152 2.3825 -0.2502]	[0.0009 1.1097 -0.0589]	[0.0012 0.2701 -0.0239]
-35	[0.0141 4.3591 -0.3925]	[0.0078 2.4502 -0.2233]	[0.0035 1.0889 -0.0978]	[0.0006 0.2741 -0.0218]

Table 3: $[\tau_x, \tau_y, \tau_z]$ VS vertical-fin angle β and the EcoMapper speed

β	u=1m/s	u=0.75m/s	u=0.5m/s	u=0.25m/s
35	[-10.6025 6.8378 -0.3275]	[-5.9951 3.7964 -0.1889]	[-2.6742 1.6529 -0.0850]	[-0.6796 0.4133 -0.0197]
30	[9.7767 6.4842 -0.1130]	[-5.5531 3.6186 -0.0760]	[-2.4999 1.5870 -0.0402]	[-0.6356 0.3924 -0.0111]
25	[-8.8230 5.3011 -0.1430]	[-5.0117 2.9466 -0.0785]	[-2.2622 1.2906 -0.0317]	[-0.5777 0.3240 -0.0060]
20	[-7.8793 4.1621 0.1481]	[-4.4812 2.3067 0.0944]	[-2.0258 1.0111 0.0502]	[-0.5190 0.2575 0.0158]
15	[-7.3541 2.8114 0.1495]	[-4.1972 1.5897 0.0969]	[-1.9057 0.7126 0.0521]	[-0.4893 0.1910 0.0152]
10	[-6.9020 2.4963 0.0907]	[-3.9485 1.3838 0.0586]	[-1.7964 0.6074 0.0304]	[-0.4622 0.1524 0.0091]
5	[-6.5234 1.1963 0.2032]	[-3.7352 0.6533 0.1186]	[-1.7036 0.2798 0.0522]	[-0.4394 0.0709 0.0123]
0	[-6.4592 0.3385 0.1455]	[-3.6968 0.1536 0.0847]	[-1.6850 0.0605 0.0376]	[-0.4344 0.0145 0.0069]

Table 4: $[\tau_\phi, \tau_\theta, \tau_\psi]$ VS vertical-fin angle β and the EcoMapper speed

β	u=1m/s	u=0.75m/s	u=0.5m/s	u=0.25m/s
35	[-0.0563 0.2789 -3.6706]	[-0.0331 0.1594 -2.0379]	[-0.0158 0.0709 -0.8868]	[-0.0040 0.0166 -0.2216]
30	[-0.0464 0.1405 -3.5014]	[-0.0271 0.0858 -1.9533]	[-0.0127 0.0415 -0.8563]	[-0.0034 0.0107 -0.2118]
25	[-0.0659 0.1700 -2.8799]	[-0.0375 0.0948 -1.5997]	[-0.0165 0.0405 -0.7001]	[-0.0039 0.0089 -0.1755]
20	[-0.0609 0.0374 -2.2345]	[-0.0333 0.0153 -1.2389]	[-0.0141 0.0025 -0.5436]	[-0.0032 -0.0012 -0.1386]
15	[-0.0533 0.0642 -1.5734]	[-0.0289 0.0296 -0.8901]	[-0.0115 0.0083 -0.3995]	[-0.0026 0.0006 -0.1065]
10	[-0.0312 0.0885 -1.2991]	[-0.0163 0.0459 -0.7228]	[-0.0064 0.0178 -0.3185]	[-0.0015 0.0033 -0.0802]
5	[-0.0284 0.0266 -0.6274]	[-0.0153 0.0128 -0.3443]	[-0.0064 0.0058 -0.1482]	[-0.0015 0.0015 -0.0374]
0	[0.0071 0.0510 -0.1526]	[0.0037 0.0269 -0.0692]	[0.0014 0.0116 -0.0283]	[0.0002 0.0040 -0.0079]

2.2.4 τ_{thrust}

We carry out open-water experiments in a tank (Figure 10) to identify the propeller coefficient c in

Equation (28) for propeller thrust calculation. The propeller thrusts and the corresponding propeller
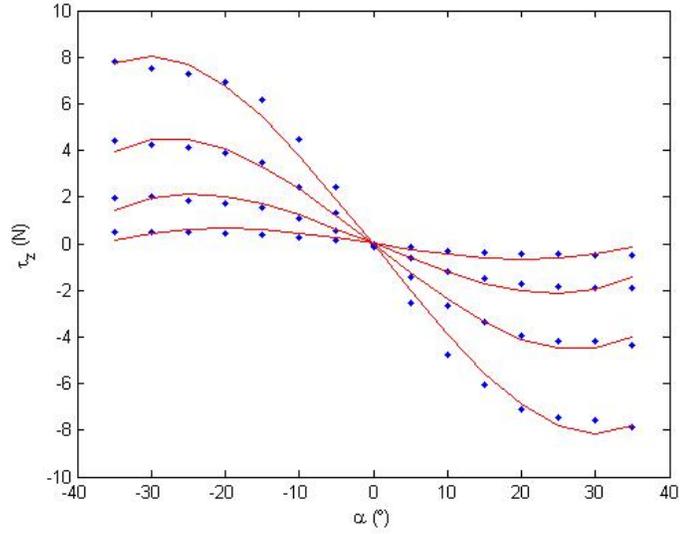
Figure 9: Horizontal-fin angle α VS τ_z

Table 5: Least-mean-square estimation of parameters b_{τ_x}, b_{τ_y}, b_{τ_z}, b_{τ_θ}, and b_{τ_ψ}

$b_{\tau_x}^T$	[-0.0578, -7.0288, 0.7926, 0.0001, 0.0001, -0.2669, -0.0038, -0.0035]
$b_{\tau_y}^T$	[0.0038,0.0000,0.2168,-0.0002]
$b_{\tau_z}^T$	[-0.0647,-0.1238,0.0001,0.4467,-0.2689,-0.8210,-0.0072,0.3821]
$b_{\tau_\theta}^T$	[-0.0377,-0.0668,0.0001,0.3624,-0.1466,-0.4481,-0.0040,0.2066]
$b_{\tau_\psi}^T$	[-0.0045, 0.0000,-0.1131,0.0000]

rotation speeds (in revolutions per minutes) in the experiments are listed in Table 6. From those data, we obtain the least-mean-square estimation of propeller coefficient

$$c = 1.5849 \times 10^{-5}. \tag{39}$$

The original data points and fitted curve are plotted in Figure 11.

2.2.5 Damping Matrix D

The last term to identify is the damping matrix D, given which, we will complete the model and be able to simulate the motion of the EcoMapper, including the linear and angular speed. To identify D, we conduct both field experiments and simulations to obtain a least-mean-square estimation. Now we use X_u to explain the procedure. For X_u, we design and carry out a field experiment so that the Ecoampper goes along a straight line without turning and diving, during a time period from $t = t_0$ to $t = t_N$. Then given an X_u, we can obtain a corresponding simulated forward speed of the EcoMapper,

Figure 10: EcoMapper in tank

Table 6: Propeller thrust VS RPM

n (rpm)	τ_{thrust} in experiment 1 (lb)	τ_{thrust} in experiment 2	τ_{thrust} in experiment 3	τ_{thrust} in experiment 4
0	0	0	0	0
186	0	0	0	0.1
371	1	0.2	0.2	0.2
558	1.2	1	1	1
745	2	1.8	1.9	2
933	3	3	3	3
1118	4	4	4.1	4
1304	6	6	5.9	6
1490	7	8	8	8
1675	9.8	10.8	10.8	10.4
1863	11.8	13.2	11.6	13
2012	14.3	14	14.3	15

denoted by $\widehat{u}$. We choose an optimal X_u satisfying $X_u = \underset{X_u}{\operatorname{argmin}} \sum_{i=0}^{N}(u(t_i) - \widehat{u}(t_i))^2$, where $u(t_i)$ is the sampled forward speed of the Ecoampper at sample time $t = t_i$ in the field experiment. After we get X_u, we plot the field experiment data and simulation results in Figure 12 in blue dots and a red curve, respectively. As the EcoMapper has anti-roll mechanism, we assume $\phi = 0$ all the time and do not care about K_p. All other terms are obtained in similar ways. We list them below and complete the dynamic model of the EcoMapper.

$$X_u = -7, Y_v = -231, Z_w = -229, M_q = -53, N_r = -48. \tag{40}$$
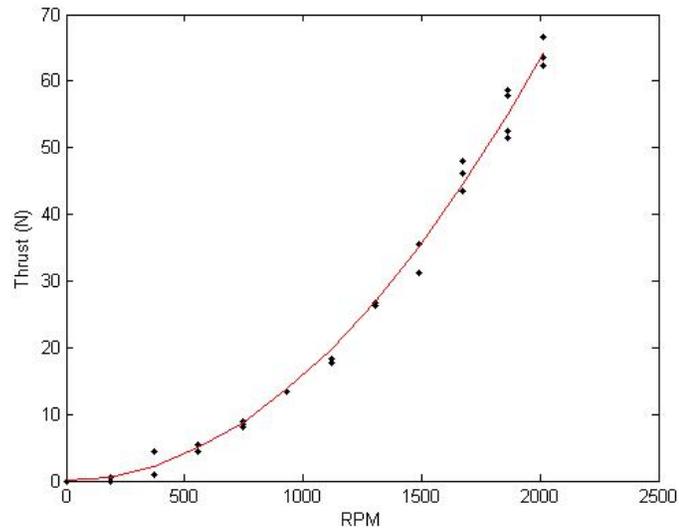
Figure 11: Thrust VS RPM
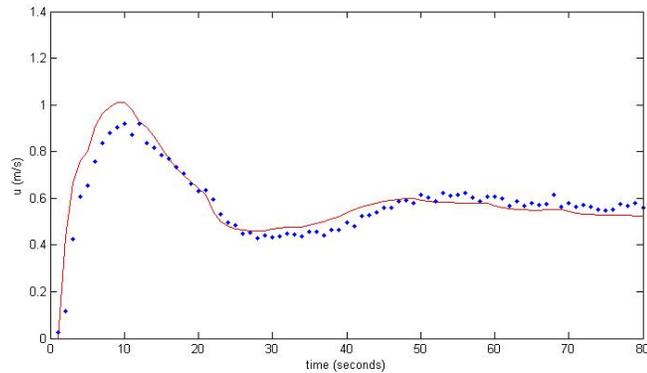
Figure 12: Experimental and simulated forward speed

2.3 Experiment Validation

To demonstrate the validity of the proposed model, we carry out a series of field experiments in the west pond in the Georgia Tech Savannah Campus shown in Figure 13 (latitude:32.167°, longitude:-81.210°). We present the results collected on January 27, 2013 from 14:00EST to 16:00EST. During that time, the maximum wind speed was about 11mph pointing east, which caused intermittent water currents. During the experiments, the EcoMapper worked in mission mode and followed pathes specified by waypoints in mission planing software "VectorMap". By arranging the distribution of the waypoints, we let the EcoMapper perform turning. In addition, we specified different forward

24

speeds and depths for different waypoints so that the EcoMapper achieved acceleration, deceleration, diving and surfacing. During missions, the EcoMapper measured vehicle velocity, attitude angles, and depth from surface through DVL (Doppler Velocity Log), a digital compass, and the YSI Depth Sensor, respectively, and saved these information every one second into log files which we can get access to after the mission completed. The information saved also includes the geographic coordinates of the vehicle position, forward speeds, lateral speeds, headings, pitch angles, depth from surface, propeller command value, and fin-angles command value. Through the propeller rotation speeds and fin angles in the experiments, which can be derived from propeller command values and fin-angle command values in the log file, we can use the proposed model to simulate the motion of the EcoMapper, and compare the simulation results with the experiential results. During simulations, a step size $\Delta t = 0.0001s$ is used. As the propeller rotation speed and fin angles saved are updated only once every second in the log files, in simulation there is only one set of propeller rotation speed and fin angles available and used within one second (i.e., 10000 iterations of simulation computation) before they are updated next second.

Figure 13: West Pond in Georgia Tech Savannah Campus

Figure 14 plots the simulated forward speed in red and experimental one in blue. We can see the simulation results are in accordance with field experiment results in accelerations and decelerations. The average of the absolute error between the simulated and experimental data is 0.1345 m/s. The

average relative error of the simulation is 15.78%. The maximum relative error is 43.93% and it happens at $t = 101$s, which is in the beginning of an acceleration. The experimental and simulated lateral speed are plotted in Figure 15 in blue and red, respectively. We can see they roughly agree with each other. As the lateral speed caused by the vertical fins is small, the lateral speed is more influenced by environment disturbance, like wind and water current, which caused the oscillations in experimental results, as shown by the blue line in Figure 15. Comparing with the field experiment data, the average absolute errors of the simulated lateral speed is 0.000369 m/s. Figure 16 compares the simulated heave speeds (the red line) with experimental ones (the blue line). We can see that they agree with each other. The average of absolute error is 0.0317 m/s. We also compared the simulation and experimental angular velocity (roll speed is excluded as we assume $\phi = 0$ all the time). From the log files, we obtained the heading and pitch angle of the EcoMapper during missions. Forward difference is used to estimate the time derivative of these Euler angles, which are then used to calculate pitch speed q and yaw speed r according to Equation (1). The simulation and experimental results on pitch speed and yaw speed are plotted in Figures 17 and 18, respectively, where red lines show the simulation results and blue lines show the experimental data. We can see that the simulations agree with field experiments, despite some subtle differences. The average of the differences in Figures 17 and 18 are 0.0438 rad/s and 0.0147 rad/s, respectively.
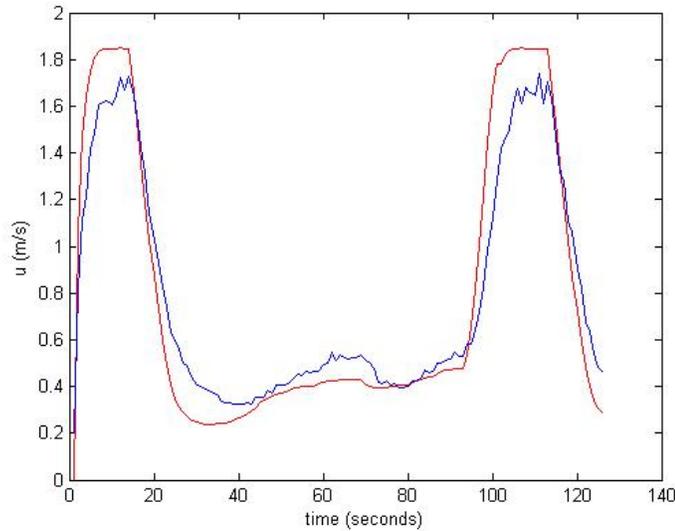
Figure 14: Experimental and simulated surge speed

26
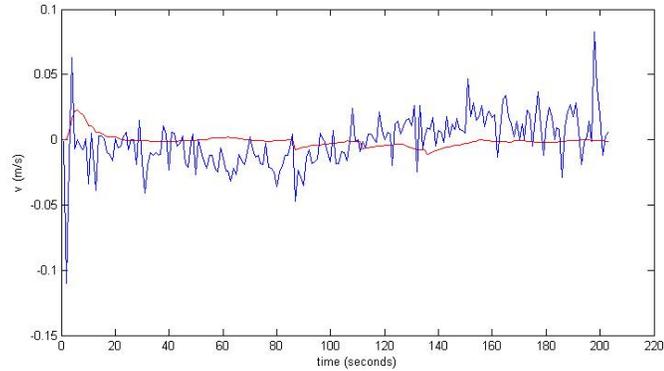
Figure 15: Experimental and simulated sway speed
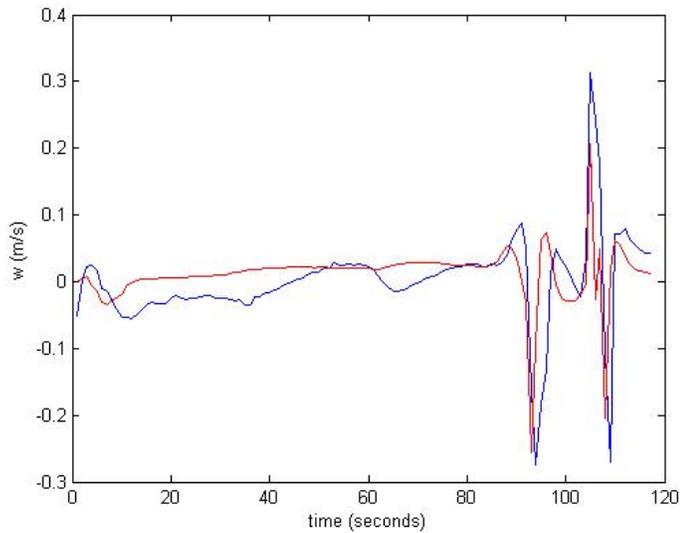
Figure 16: Experimental and simulated heave speed

There are several factors causing difference between experimental data and simulation results. First, the disturbances during experiments, including wind, water currents, cause some of the difference. Second, because the propeller rotation speed and fin angles are recorded only every second during experiments, so in simulation we can use only one available set of these values in every second, i.e., 10000 simulation computation iterations, while the actual propeller rotation speed and fin angles were varying continuously within every second. This also causes some difference between the experimental data and simulation results. But nevertheless, we see a consistent match between the field experiments and simulations, which shows that the EcoMapper model captures the EcoMapper dynamics to a satisfactory accuracy.
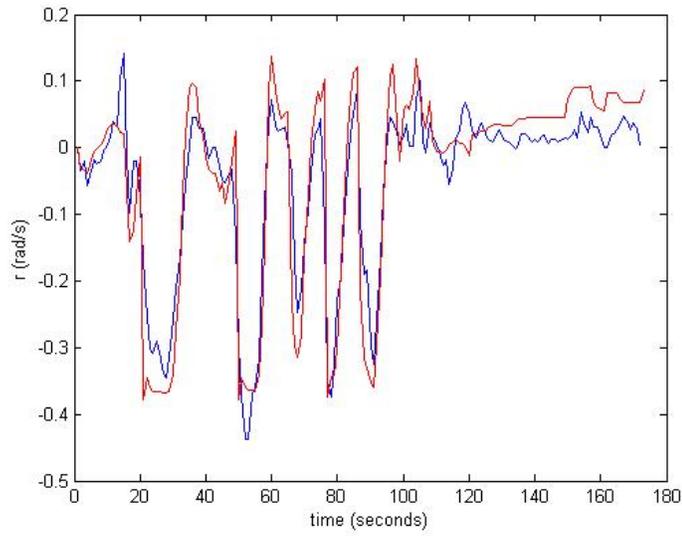
Figure 17: Experimental and simulated pitch speed
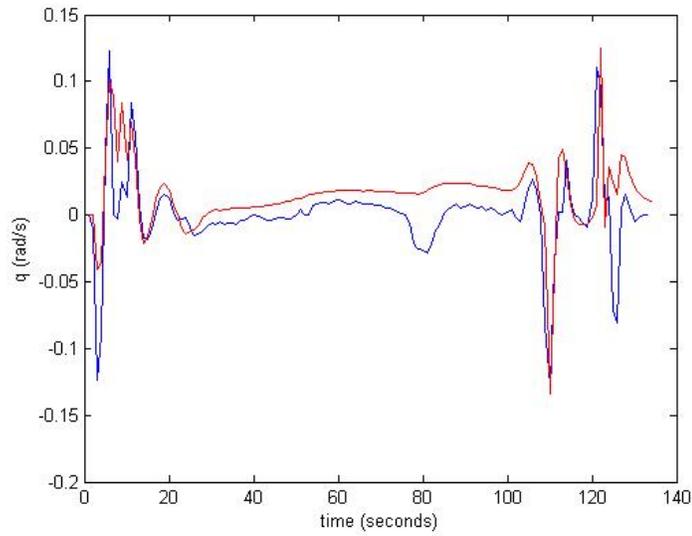
Figure 18: Experimental and simulated yaw speed

2.4   Conclusion

This chapter proposed a dynamic model to describe the motion of the EcoMapper. Using theoretical calculations, computational-fluid-dynamics simulations, and field tests, we identified all the parameters in the model. We also performed field experiments to validate the proposed model, and the experimental data is consistent with simulation results. Therefore, the proposed model may be used to simulate the EcoMapper motions and compute desired control input for the EcoMapper in

real-time control. The modeling methods may also applies to other underwater vehicles with control surfaces.

CHAPTER III

A DECOUPLED CONTROLLER DESIGN APPROACH FOR FORMATION CONTROL OF AUTONOMOUS UNDERWATER VEHICLES WITH TIME DELAYS

Recently, formation control of multiple AUVs has received much attention [2, 4–6, 9, 25–28] due to potentially broad applications in ocean engineering and science. Despite of the complex AUV dynamics, many previous results on formation control are developed assuming each agent is modeled by either a single or a double integrator particle model [29–31]. Each agent cooperates with others based on information shared through communication links that introduce time delays in the system dynamics. [32–35] analyzed the stability of a multi-agent system with time delays. [36] proved the input-to-state stability of a single-master-N-slave structured agent group under time delays, assuming double-integrator dynamics for each agent.

Formation control becomes more challenging if more practical and complex dynamics and communication constraints are concerned. These challenges are crucial for AUVs because of their complex and coupled dynamics. In addition, underwater communication and positioning, which rely heavily on acoustic systems, are plagued with limited communication bandwidth, intermittent failures, latency and multi-path effects, which causes time delays. The complex dynamics of AUVs are considered in [25, 27, 78, 79] with various control strategies for multi-AUV systems proposed without considering time delays. A collection of identical planar unit-speed vehicles described by a two-dimensional Frenet-Serret motion model are investigated to integrate communication and control in [80]. Using Lyapunov theory and a switching communication topology, [81] developed a coordinated path-following controller that can tolerate communication failures between AUVs. [82] proposed a path-following control strategy to coordinate a group of surface vessels moving in a horizontal plane. A cooperative control law which was proved robust to small communication delays is achieved in [83]. [84] studied formation dynamics for a group of AUVs in a horizontal plane. Through Jacobi coordinates, the formation dynamics were expressed as a deformable body and H_∞

full state feedback controllers were designed assuming the time delays negligible. There also exist research works on formation control under time delays for unmanned aerial vehicles(UAVs) [85], bilateral teleoperators [86], and spacecrafts [87] etc.

This chapter extends the Jacobi transform approach in [84] to decompose the formation dynamics for a group of 6DOF AUVs moving in three dimensional space. According to the formation dynamics, we propose a robust controller to achieve both path following and formation keeping while tolerating bounded constant time delays. We propose a decoupled design approach. The orientation subsystem of each AUV is separated from the formation system. The formation shape and the formation center are also decoupled and controlled separately. In addition, we devise a two-step approach for the robust controller design to achieve delay-dependent robust stability.

The remainder of this chapter is organized as follows. Section 3.1 reviews the 6DOF dynamical model of a single AUV. In section 3.2, we derive the formation dynamics of AUVs in three-dimensional space. Section 3.3 designs a robust formation position controller and a velocity controller to stabilize the formation system under perturbations and time delays. We demonstrate simulation results in Section 3.4 and provide conclusions in Section 3.5.

3.1 Dynamics and Orientation Control of A Single AUV

We assume that every AUV in the formation is fully actuated, homogeneous, and neutrally buoyant, with its center of buoyancy coinciding with its center of gravity. We set the origin of the body-fixed frame at the center of buoyancy and assume that the hydrodynamic forces and moments are linear. Let $\eta_1 = [x, y, z]^T$ denote the AUV position in the earth-fixed frame. Let $\eta_2 = [\phi, \theta, \psi]^T$ denote the Euler-angle vector for roll, pith, and yaw in earth-fixed frame. Let $v_1 = [u, v, w]^T$ denote the body-fixed linear velocity for surge, sway, and heave. And let $v_2 = [p, q, r]^T$ denote the body-fixed angular velocity for roll, pith, and yaw. Then the dynamics of each AUV can be expressed as follows [70, 71]:

$$\dot{\eta}_1 = J_1(\eta_2)v_1, \tag{41}$$

$$\dot{\eta}_2 = J_2(\eta_2)v_2, \tag{42}$$

$$M\dot{v} + C(v)v + D(v)v = \tau, \tag{43}$$

where $v = [v_1^T, v_2^T]^T$, and $\tau = \begin{bmatrix} \tau_1^T & \tau_2^T \end{bmatrix}^T$ is the control inputs vector where τ_1^T is the control-

ling force vector and τ_2^T the controlling moment vector. In this model, $\begin{bmatrix} J_1(\eta_2) & 0 \\ 0 & J_2(\eta_2) \end{bmatrix}$ is the

coordinate-transform (Jacobian) matrix from the body-fixed frame to the earth-fixed frame. The

matrices M and D in this model are inertia and damping matrices, defined as

$$M = \text{diag}\{m - X_{\dot{u}}, m - Y_{\dot{v}}, m - Z_{\dot{w}}, I_x, I_y, I_z\}, \tag{44}$$

$$D(v) = -\text{diag}\{X_u, Y_v, Z_w, K_p, M_q, N_r\}. \tag{45}$$

where, m is the AUV mass. $X_{\dot{u}}$, $Y_{\dot{v}}$, $Z_{\dot{w}}$, X_u, Y_v, Z_w, K_p, M_q, and N_r are hydrodynamical parameters.
The terms I_x, I_y, and I_z are the moments of inertia about the three axes of earth-fixed frame.

The matrix $C(v)$ represents the Coriolis and centripetal term, which contains the rigid-body
Coriolis and centripetal term $C_{RB}(v)$ and the hydrodynamic added Coriolis and centripetal term
$C_A(v)$, i.e.,

$$C(v) = C_{RB}(v) + C_A(v), \tag{46}$$

where

$$C_{RB}(v) = \left[\begin{array}{ccc|ccc} 0 & 0 & 0 & 0 & mw & -mv \\ 0 & 0 & 0 & -mw & 0 & mu \\ 0 & 0 & 0 & mv & -mu & 0 \\ \hline 0 & mw & -mv & 0 & I_zr & -I_yq \\ -mw & 0 & mu & -I_zr & 0 & I_xp \\ mv & -mu & 0 & I_yq & I_xp & 0 \end{array}\right], \tag{47}$$

$$C_A(v) = \left[\begin{array}{ccc|ccc} 0 & 0 & 0 & 0 & Z_{\dot{w}}w & -Y_{\dot{v}}v \\ 0 & 0 & 0 & -Z_{\dot{w}}w & 0 & X_{\dot{u}}u \\ 0 & 0 & 0 & Y_{\dot{v}}v & -X_{\dot{u}}u & 0 \\ \hline 0 & Z_{\dot{w}}w & -Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ -Z_{\dot{w}}w & 0 & X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ Y_{\dot{v}}v & -X_{\dot{u}}u & 0 & -M_{\dot{q}}q & K_{\dot{p}}p & 0 \end{array}\right]. \tag{48}$$

Note that the $C_{RB}(v)$ and $C_A(v)$ satisfy the following properties:

$$C'_{RB}(v)v = C_{RB}(v)v,$$

$$C'_A(v)v = C_A(v)v, \tag{49}$$

where,

$$C'_{RB}(v) = \left[\begin{array}{ccc|ccc} 0 & -mr & mq & 0 & 0 & 0 \\ mr & 0 & -mq & 0 & 0 & 0 \\ -mq & mp & 0 & 0 & 0 & 0 \\ \hline 0 & mw & 0-mv & 0 & I_z r & -I_y q \\ -mw & 0 & mu & -I_z r & 0 & I_x p \\ mv & -mu & 0 & I_y q & I_x p & 0 \end{array}\right], \tag{50}$$

$$C'_A(v) = \left[\begin{array}{ccc|ccc} 0 & Y_{\dot{v}}r & -Z_{\dot{w}}q & 0 & 0 & 0 \\ -X_{\dot{u}}r & 0 & Z_{\dot{w}}p & 0 & 0 & 0 \\ X_{\dot{u}}q & -Y_{\dot{v}}p & 0 & 0 & 0 & 0 \\ \hline 0 & Z_{\dot{w}}w & -Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ -Z_{\dot{w}}w & 0 & X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ Y_{\dot{v}}v & -X_{\dot{u}}u & 0 & -M_{\dot{q}}q & K_{\dot{p}}p & 0 \end{array}\right]. \tag{51}$$

Now we can rewrite term $C(v)v+D(v)v$ in Equation (43) as

$$C(v)v+D(v)v = C'_{RB}(v)v+C'_A(v)v+D(v)v = \left[\begin{array}{c|c} C_{11}(v_2) & 0 \\ \hline C_{21}(v_1) & C_{22}(v_2) \end{array}\right]v \doteq C'(v)v. \tag{52}$$

where

$$C_{11}(v_2) = \left[\begin{array}{ccc} -X_u & -mr+Y_{\dot{v}}r & mq-Z_{\dot{w}}q \\ mr-X_{\dot{u}}r & -Y_v & -mp+Z_{\dot{w}}p \\ -mq+X_{\dot{u}}q & mp-Y_{\dot{v}}p & -Z_w \end{array}\right], \tag{53}$$

$$C_{21}(v_1) = \left[\begin{array}{ccc} 0 & mw+Z_{\dot{w}}w & -mv+Y_{\dot{v}}v \\ -mw-Z_{\dot{w}}w & 0 & mu-X_{\dot{u}}u \\ mv-Y_{\dot{v}}v & -mu+X_{\dot{u}}u & 0 \end{array}\right], \tag{54}$$

$$C_{22}(v_2) = \begin{bmatrix} -K_p & I_z r - N_{\dot{r}} r & -I_y q + M_{\dot{q}} q \\ -I_z r + N_{\dot{r}} r & -M_q & I_x p - K_{\dot{p}} p \\ I_y q - M_{\dot{q}} q & -I_x p + K_{\dot{p}} p & -N_r \end{bmatrix}. \tag{55}$$

Therefore, Equation (43) can be decomposed into the following two equations:

$$M_1 \dot{v}_1 + C_{11}(v_2) v_1 = \tau_1, \tag{56}$$

$$M_2 \dot{v}_2 + C_{21}(v_1) v_1 + C_{22}(v_2) v_2 = \tau_2, \tag{57}$$

where

$$M_1 = \text{diag}\{m - X_{\dot{u}}, m - Y_{\dot{v}}, m - Z_{\dot{w}}\},$$

$$M_2 = \text{diag}\{I_x, I_y, I_z\}. \tag{58}$$

Since the mass m is always positive and its absolute value is much larger than the absolute values of hydrodynamical parameters $X_{\dot{u}}$, $Y_{\dot{v}}$, and $Z_{\dot{w}}$, the matrix M_1 is usually nonsingular for an AUV and M_1^{-1} exists.

From Equation (41), we get

$$v_1 = J_1^{-1}(\eta_2) \dot{\eta}_1, \tag{59}$$

and

$$\dot{v}_1 = J_1^{-1}(\eta_2) \ddot{\eta}_1 - J_1^{-1}(\eta_2) \dot{J}_1(\eta_2) J_1^{-1}(\eta_2) \dot{\eta}_1. \tag{60}$$

Substitute Equations (59) and (60) into Equation (56), we obtain

$$\ddot{\eta}_1 = (\dot{J}_1(\eta_2) J_1^{-1}(\eta_2) - J_1(\eta_2) M_1^{-1} C_{11}(v_2) J_1^{-1}(\eta_2)) \dot{\eta}_1 + J_1(\eta_2) M_1^{-1} \tau_1. \tag{61}$$

Define

$$\mathscr{G}(\eta_2, v_2) = \dot{J}_1(\eta_2) J_1^{-1}(\eta_2) - J_1(\eta_2) M_1^{-1} C_{11}(v_2) J_1^{-1}(\eta_2), \tag{62}$$

and

$$H(\eta_2) = J_1(\eta_2) M_1^{-1}, \tag{63}$$

then Equation (61) becomes

$$\ddot{\eta}_1 = \mathscr{G}(\eta_2, v_2) \dot{\eta}_1 + H(\eta_2) \tau_1. \tag{64}$$

Now Equation (64) describes the complete translational dynamics of each AUV.

From equation (61), we observe that if we are able to stablize η_2 and v_2 at desired values η_{2d} and v_{2d}, then we can derive the control τ_1 to be relying only on η_1 and v_1. This idea will be used to derive the formation control of AUVs.

In order to achieve the desired values η_{2d} and v_{2d}, Equations (42) and (57) will be considered. We rewrite Equation (57) as

$$\dot{v}_2 = -M_2^{-1}C_{22}v_2 + M_2^{-1}\tau_2 - M_2^{-1}C_{21}v_1, \tag{65}$$

and make the following invertible transformation for τ_2

$$\varsigma = M_2^{-1}\tau_2 - M_2^{-1}C_{21}v_1, \tag{66}$$

then we get

$$\dot{v}_2 = -M_2^{-1}C_{22}v_2 + \varsigma, \tag{67}$$

where C_{21} and C_{22} are given in Equations (54) and (55). Then the control effort ς can be designed to achieve $\eta \to \eta_{2d}$ and $v \to v_{2d}$. We omit the details here since this is just a standard feedback linearization and pole assignment problem.

3.2 AUV Formation Dynamics

3.2.1 Formation Dynamics of AUVs

We consider a formation of N AUVs moving in a three-dimensional space. Jacobi shape theory was used to study the shape of formation of particles [31] and AUVs in two-dimensional horizontal plane [84]. Let $\eta_1^i = [x_i, y_i, z_i]^T$ denote the position of the i-th AUV in the earth-fixed frame, where $i = 1, 2, ..., N$, then the Jacobi vectors are defined by a linear transform Ξ that results in the following equation:

$$[\rho_1^T, \rho_2^T, ..., \rho_{N-1}^T, q_c^T]^T = \Xi[(\eta_1^1)^T, (\eta_1^2)^T, ..., (\eta_1^N)^T]^T, \tag{68}$$

where ρ_j, $j = 1, 2, \ldots, N-1$, are $N-1$ independent Jacobi vectors describing the geometric formation shape formed by N AUVs and q_c is the formation center defined by $q_c = \frac{1}{N}\sum_{i=1}^{N}\eta_1^i$. The linear transform Ξ is guaranteed to exist [31]. Our goal is to design a formation control to guarantee

$$\rho_j \to \rho_{jd}, \dot{\rho}_j \to \dot{\rho}_{jd}, q_c \to q_{cd}, \dot{q}_c \to \dot{q}_{cd}, \tag{69}$$

where ρ_{jd} is the desired value of the j-th Jacobi vector and q_{cd} is the desired trajectory of the formation center. The control goals (69) assures that the AUVs converge to the desired formation and the formation center tracks the desired trajectory.

Take second-order derivatives on both sides of Equation (68), we get

$$[\ddot{\rho}_1^T, \ddot{\rho}_2^T, ..., \ddot{\rho}_{N-1}^T, \ddot{q}_c^T]^T = \Xi[(\ddot{\eta}_1^1)^T, (\ddot{\eta}_1^2)^T, ..., (\ddot{\eta}_1^N)^T]^T. \tag{70}$$

Substitute Equation (64) into Equation (70), and define state vector

$$X = [\rho_1^T, ..., \rho_{N-1}^T, q_c^T]^T, \tag{71}$$

we obtain the formation dynamics as follows:

$$\ddot{X} = A([\eta_2^i], [v_2^i])\dot{X} + \widetilde{\Gamma}([\eta_2^i])U, \tag{72}$$

where

$$A([\eta_2^i], [v_2^i]) = \Xi\mathbf{G}\Xi^{-1}, \tag{73}$$

$$\mathbf{G} = \mathrm{diag}\{\mathscr{G}_1(\eta_2^1, v_2^1), ..., \mathscr{G}_N(\eta_2^N, v_2^N)\}, \tag{74}$$

$$\widetilde{\Gamma}([\eta_2^i]) = \Xi\mathrm{diag}\{H(\eta_2^1), ..., H(\eta_2^N)\}, \tag{75}$$

$$U = \begin{bmatrix} (\tau_1^1) & \cdots & (\tau_1^N) \end{bmatrix}^T. \tag{76}$$

3.2.2 Decoupling of the Formation Dynamics

We have shown that the formation dynamics are described by Equation (72), in which matrix A is a nonlinear function of $p_i, q_i, r_i, \phi_i, \theta_i$, and ψ_i, for $i = 1, 2, \cdots, N$. We decompose A into two parts: one part is a diagonal matrix, and other remaining part is viewed as a perturbation. Specifically, we define $A_\lambda = \lambda\mathbf{e}_{3N}$, where λ is a constant scalar and $\mathbf{e}_{3N}$ is an identity matrix, and let

$$A([\eta_2^i], [v_2^i]) = A_\lambda + A_\Delta([\eta_2^i], [v_2^i]), \tag{77}$$

then A_Δ is viewed as a perturbation. We can find the value for λ such that the H-infinity norm of A_Δ is bounded and minimized.

Since the angular speed of an AUV is small when the AUV is close to a steady state, we let $p_i \to 0$, $q_i \to 0$, and $r_i \to 0$, and find

$$\mathbf{G} \to \mathbf{G}'. \tag{78}$$

36

See the details of $\mathbf{G}'$ in Appendix C. As $\frac{X_u}{m-X_{\dot{u}}}$, $\frac{Y_v}{m-Y_{\dot{v}}}$, and $\frac{Z_w}{m-Z_{\dot{w}}}$ in $\mathbf{G}'$ are all negative, we select

$$\lambda = \min(\frac{X_u}{m-X_{\dot{u}}}, \frac{Y_v}{m-Y_{\dot{v}}}, \frac{Z_w}{m-Z_{\dot{w}}}), \tag{79}$$

then the H-infinity norm of the perturbation term A_Δ is minimized. Define $\sigma^2 = \lambda_{max}(A_\Delta^T A_\Delta)$, then from Equations (73) and (74) and the expressions of the entries of $\mathbf{G}'$, we can see that σ is a function of p_i, q_i, r_i, and sine and cosine functions of θ_i, ϕ_i, and ψ_i. Because p_i, q_i, and r_i are bounded as the vehicle can not steer infinitely fast, and sine and cosine functions are also bounded, we conclude that $\sigma < \infty$.

Note that the perturbation term A_Δ is caused by asymmetries of an AUV. For AUVs with three planes of symmetry, $X_{\dot{u}} = Y_{\dot{v}} = Z_{\dot{w}}$ and $X_u = Y_v = Z_w$, which will make $\mathbf{G}$ a constant diagonal matrix, and then $A = A_\lambda = \frac{X_u}{m-X_{\dot{u}}}\mathbf{e}_{3N}$, therefore, $A_\Delta = 0_{3N}$, i.e., the perturbation term vanishes.

3.3 Robust Formation Controller Design

3.3.1 Formation System with Time Delay

Define $Z = \dot{X}$ and plug Equation (77) into Equation (72), we get

$$\begin{cases} \dot{X}(t) = Z(t) \\ \dot{Z}(t) = A_\lambda Z + A_\Delta([\eta_2^i], [v_2^i])Z + \widetilde{\Gamma}([\eta_2^i])U. \end{cases} \tag{80}$$

Define error vectors $X_e = X - X_d$ and $Z_e = Z - Z_d$, where X_d and Z_d are desired values, and a perturbation term $W(t) = A_\Delta Z_e(t)$, we get

$$\begin{cases} \dot{X}_e(t) = Z_e(t), \\ \dot{Z}_e(t) = A_\lambda Z_e(t) + W(t) + \mu(t), \end{cases} \tag{81}$$

where

$$\mu(t) = \widetilde{\Gamma}([\eta_2^i])U(t) + A_\lambda Z_d + A_\Delta([\eta_2^i], [v_2^i])Z_d - \dot{Z}_d \tag{82}$$

is the control we need to design, which will be a function of error vectors X_e and Z_e. To compute X_e and Z_e, position and velocity information of all AUVs are needed. As it takes time for the controller to get these information through communication links, the control effort is actually computed based on delayed information, i.e., $\mu(t)$ is a function of $X_e(t-h)$ and $Z_e(t-h)$. Here h is a time delay satisfying $0 \le h \le h^*$ and $\dot{h} = 0$, where h^* is the upper bound of the time delay. We rewrite Equation

(81) as

$$\dot{\bar{X}}(t) = \bar{A}\bar{X}(t) + DW(t) + B_h\mu(t)$$

$$Z_e(t) = L\bar{X}(t), \tag{83}$$

where

$$\bar{X} = \begin{bmatrix} X_e^T & Z_e^T \end{bmatrix}^T, \bar{A} = \begin{bmatrix} 0 & \mathbf{e} \\ 0 & A_\lambda \end{bmatrix}, D = B_h = \begin{bmatrix} 0 \\ \mathbf{e} \end{bmatrix}, L = \begin{bmatrix} 0 & \mathbf{e} \end{bmatrix}. \tag{84}$$

$\mathbf{e}$ is an identity matrix.

Suppose the time delay h satisfies $0 \le h \le h^*$ and $\dot{h} = 0$. Let the control law $\mu(t) = F\bar{X}(t-h)$ be designed where $F = [k_2, \ k_1]$ The goal for the formation system (83) under control μ is to achieve robust stability under the time delay h and the perturbation $W(t)$.

For the robust stability design, given $\|A_\Delta\|_\infty < \infty$, according to the Small Gain Theorem [88], system (83) with perturbation $W(t)$ is well-posed and internally stable for all $A_\Delta \in RH_\infty$ with $\|A_\Delta\|_\infty \le 1/\gamma$ if and only if $\|T_{Z_eW}\|_\infty < \gamma$, where $\gamma > 0$, and T_{Z_eW} is the closed-loop transfer function matrix from W to Z_e. Therefore, the following theorem provides a guidance on how to choose F to achieve robust stability for time delay.

Theorem 1 *The formation system (83) under control μ will be stable under bounded time delay $h \in [0, h^*]$, and the transfer function from W to Z_e satisfies $\|T_{Z_eW}\| \le \gamma$, where $\gamma > 0$, if the following conditions hold:*

(1) There exists a positive symmetry matrix $Y_1 = Y_1^T > 0$ and scalars $\varepsilon_1 > 0, \varepsilon_2 > 0, \varepsilon_3 > 0$, which satisfy $h^*\varepsilon_3 - \gamma^2 < 0$ and the matrix inequality:

$$\begin{bmatrix} \Theta_1 & Y_1\bar{A}^T & Y_1F^TB_h^T & Y_1L^T \\ \bar{A}Y_1 & -\frac{1}{h^*\varepsilon_1}\mathbf{e} & 0 & 0 \\ B_hFY_1 & 0 & -\frac{1}{h^*\varepsilon_2}\mathbf{e} & 0 \\ LY_1 & 0 & 0 & -\mathbf{e} \end{bmatrix} < 0 \tag{85}$$

where $\Theta_1 = (\bar{A} + B_hF)Y_1 + Y_1(\bar{A} + B_hF)^T + h^*(\varepsilon_1^{-1} + \varepsilon_2^{-1} + \varepsilon_3^{-1})B_hFF^TB_h^T + \frac{1}{\gamma^2 - h^*\varepsilon_3}DD^T$.

(2) For any choice of the feedback gain F, there exists a positive symmetry matrix $Y_2 = Y_2^T > 0$

and scalars $\varepsilon_4 > 0, \varepsilon_5 > 0$, which satisfy the following LMI:

$$
\begin{bmatrix}
\Theta_2 & Y_2\bar{A}^T & Y_2F^TB_h^T \\
\bar{A}Y_2 & -\frac{1}{h^*\varepsilon_4}\mathbf{e} & 0 \\
B_hFY_2 & 0 & -\frac{1}{h^*\varepsilon_5}\mathbf{e}
\end{bmatrix} < 0
\tag{86}
$$

Where $\Theta_2 = Y_2(\bar{A}+B_hF)^T + (\bar{A}+B_hF)Y_2 + h^*(\varepsilon_4^{-1}+\varepsilon_5^{-1})B_hFF^TB_h^T$.

Proof The closed-loop system under delay can be converted to the following system by the "time-streching" transform [89]

$$
\dot{\bar{X}}(t) = (A+B_hF)\bar{X}(t) - B_hF\left\{\int_{-h}^0 (A\bar{X}(t+\theta)+B_hF\bar{X}(t-h+\theta)+DW(t+\theta))d\theta\right\} + DW(t)
\tag{87}
$$

Let $P = P^T > 0$. We define a Lyapunov-Krasovskii functional $V[\bar{X}(t)]$ as

$$
\begin{aligned}
V[\bar{X}(t)] =& \bar{X}^T(t)P\bar{X}(t) + \int_{-h}^0\int_{t+\theta}^t \varepsilon_1[\bar{X}^T(s)\bar{A}^T\bar{A}\bar{X}(s)]dsd\theta \\
&+ \int_{-h}^0\int_{t-h+\theta}^t \varepsilon_2[\bar{X}^T(s)F^TB_h^TB_hF\bar{X}(s)]dsd\theta \\
&+ \int_{-h}^0\int_{t-h+\theta}^t \varepsilon_3[W^T(s)D^TDW(s)]dsd\theta.
\end{aligned}
\tag{88}
$$

Then take the time derivative of $V[\bar{X}(t)]$ along the trajectory of (87), we have

$$
\begin{aligned}
\dot{V}[\bar{X}(t)] =& \bar{X}^T(t)[P(A+B_hF)+(A+B_hF)^TP]\bar{X}(t)+W^T(t)D^TP\bar{X}(t)+\bar{X}^T(t)PDW(t) \\
&- 2\bar{X}^T(t)PB_hF\cdot\left\{\int_{-h}^0(A\bar{X}(t+\theta)+B_hF\bar{X}(t-h+\theta)+DW(t+\theta))d\theta\right\} \\
&+ h\varepsilon_1\bar{X}^T(t)A^TA\bar{X}(t) - \int_{-h}^0\varepsilon_1\bar{X}^T(t+\theta)A^TA\bar{X}(t+\theta)d\theta + \int_{-h}^0\varepsilon_2[\bar{X}^T(t)F^TB_h^TB_hF\bar{X}(t) \\
&- \bar{X}^T(t-h+\theta)F^TB_h^TB_hF\bar{X}(t-h+\theta)]d\theta \\
&+ \int_{-h}^0\varepsilon_3[W^T(t)D^TDW(t)-W^T(t+\theta)D^TDW(t+\theta)]d\theta.
\end{aligned}
$$

Next we use the following inequality:

$$
-2a^Tb \le ra^Ta + r^{-1}b^Tb
$$

where a and b are vectors with compatible dimension and $r > 0$ is a positive real number. Apply this inequality to terms $-2\bar{X}^T(t)PB_hF\int_{-h}^0 A\bar{X}(t+\theta)d\theta$, $-2\bar{X}^T(t)PB_hF\int_{-h}^0 B_hF\bar{X}(t-\tau+\theta)d\theta$, and

$-2\bar{X}^T(t)PB_hF\int_{-h}^0 DW(t+\theta)d\theta$, we can then derive the following inequality for the time derivative of V:

$$\dot{V}[\bar{X}(t)] \leq x^T(t)[P(\bar{A}+B_hF)+(\bar{A}+B_hF)^TP+h^*\varepsilon_1\bar{A}^T\bar{A}$$

$$+h^*\varepsilon_2F^TB_h^TB_hF+h^*(\varepsilon_1^{-1}+\varepsilon_2^{-1}+\varepsilon_3^{-1})PB_hFF^TB_h^TP]\bar{X}(t)$$

$$+W^T(t)D^TP\bar{X}(t)+\bar{X}^T(t)PDW(t)+h^*\varepsilon_3W^T(t)D^TDW(t) \tag{89}$$

Next we construct the Hamilton-Jacobi equation to find a sufficient condition for $\|T_{Z_eW}\| \leq \gamma$. We define the Hamiltonian as $H(\bar{X},W,t) = \dot{V}[\bar{X}(t)] + Z_e^T(t)Z_e(t) - \gamma^2W^T(t)D^TDW(t)$. Plug Equation (89) into $H(\bar{X},W,t)$, we get

$$H(\bar{X},W,t) \leq \bar{X}^T(t)\Theta\bar{X}(t)+W^T(t)D^TPW(t)+\bar{X}^T(t)PDW(t)$$

$$+h^*\varepsilon_3W^T(t)D^TDW(t)-\gamma^2W^T(t)D^TDW(t)$$

$$=\hbar^T(t)\Lambda^*(P)\hbar(t) \tag{90}$$

where $\hbar(t) = \begin{bmatrix} \bar{X}^T(t) & W^T(t) \end{bmatrix}^T$, $\Theta = P(\bar{A}+B_hF)+(\bar{A}+B_hF)^TP+h^*\varepsilon_1\bar{A}^T\bar{A}+h^*\varepsilon_2F^TB_h^TB_hF+h^*(\varepsilon_1^{-1}+\varepsilon_2^{-1}+\varepsilon_3^{-1})PB_hFF^TB_h^TP+L^TL$, and

$$\Lambda^*(P) = \begin{bmatrix} \Theta & PD \\ D^TP & (h^*\varepsilon_3-\gamma^2)D^TD \end{bmatrix} \tag{91}$$

We know that the system is robustly stable with a disturbance attenuation γ if $H(\bar{X},W,t) < 0$, which is equivalent to $\Lambda^*(P) < 0$. This will imply $h^*\varepsilon_3 - \gamma^2 < 0$ and the following algebraic Ricatti inequality (ARI)

$$P(\bar{A}+B_hF)+(\bar{A}+B_hF)^TP+h^*\varepsilon_1\bar{A}^T\bar{A}+h^*\varepsilon_2F^TB_h^TB_hF$$

$$+h^*(\varepsilon_1^{-1}+\varepsilon_2^{-1}+\varepsilon_3^{-1})PB_hFF^TB_h^TP+L^TL+\frac{1}{\gamma^2-h^*\varepsilon_3}PDD^TP < 0 \tag{92}$$

Note that $D^TD = \mathbf{e}$ and define $Y_1 = P^{-1}$. We premultiply and postmultiply (92) by Y_1 and get the following equivalent equation:

$$(\bar{A}+B_hF)Y_1+Y_1(\bar{A}+B_hF)^T+h^*\varepsilon_1Y\bar{A}^T\bar{A}Y_1+h^*\varepsilon_2Y_1F^TB_h^TB_hFY_1$$

$$+h^*(\varepsilon_1^{-1}+\varepsilon_2^{-1}+\varepsilon_3^{-1})B_hFF^TB_h^T+Y_1L^TLY_1+\frac{1}{\gamma^2-h^*\varepsilon_3}DD^T < 0. \tag{93}$$

According to the Schur Complement Theorem, the ARI (93) is equivalent to the following matrix inequality:

$$
\begin{bmatrix}
\Theta_1 & Y_1\bar{A}^T & Y_1 F^T B_h^T & Y_1 L^T \\
\bar{A}Y_1 & -\frac{1}{h^*\varepsilon_1}\mathbf{e} & 0 & 0 \\
B_h F Y_1 & 0 & -\frac{1}{h^*\varepsilon_2}\mathbf{e} & 0 \\
L Y_1 & 0 & 0 & -\mathbf{e}
\end{bmatrix} < 0
\tag{94}
$$

where $\Theta_1 = (\bar{A}+B_h F)Y_1 + Y_1(\bar{A}+B_h F)^T + h^*(\varepsilon_1^{-1}+\varepsilon_2^{-1}+\varepsilon_3^{-1})B_h F F^T B_h^T + \frac{1}{\gamma^2 - h^*\varepsilon_3}DD^T < 0$.

Since the matrix inequality (94) only guarantee that $T_{ZeW} \leq \gamma$, if we get the feedback matrix F, we also need check whether F can guarantee the stability of the unperturbed formation system. According the delay-dependent stability of time-delay system [90], we have the LMI (86). ∎

Remark 1 *Given this theorem, one would wish to solve the matrix inequality (85) for F and Y. But this is not easy. The feedback gain matrix F and the unknown matrix Y both need to be computed. But the matrix inequality is not linear, and can not be easily converted to a linear matrix inequality (LMI) due to the term that contains FF^T in Θ. This difficulty is caused by the time delay h^*. To solve for the problem when $h^* > 0$, one may make a guess for F and then try to solve for Y with LMI tools. Note that simply making $h^* = 0$ in (85) to find a guess for F will not work.*

To find the gain F, we propose a procedure that contains two steps. First, we design k_1 such that $\mu'(t) = k_1 Z_e(t-h)$ robustly stabilizes the following velocity control subsystem:

$$
\dot{Z}_e(t) = A_\lambda Z_e(t) + W(t) + \mu'(t)
\tag{95}
$$

with *delay-independent* stability. Then, fix the k_1, we search for k_2 to robustly stabilize the full system with *delay-dependent* stability. The first step is more conservative than necessary, but produces easy to compute solutions. The second step is then used to achieve the final design.

3.3.2 Velocity Controller Design

Theorem 2 *Suppose the time delay h satisfies $0 \leq h \leq h^*$ and $\dot{h} = 0$. Let $P_1 = P_1^T > 0$ and $P_2 < 0$*

be solutions of the following linear matrix inequality (LMI):

$$\Lambda_1^* = \begin{bmatrix} A_\lambda^T P_1 + P_1 A_\lambda + Q + \mathbf{e} & P_2 & P_1 \\ P_2^T & -Q & 0 \\ P_1 & 0 & -\gamma^2 \mathbf{e} \end{bmatrix} < 0. \tag{96}$$

The control law $\mu'(t) = k_1 Z_e(t - h)$, where

$$k_1 = P_1^{-1} P_2 < 0, \tag{97}$$

robustly stabilizes the formation velocity subsystem (95), i.e., $\|T_{Z_e W}\|_\infty < \gamma$, where $\gamma \leq 1/\|A_\Delta\|$.

Proof Let the output of system (95) to be Z_e. We define a Lyapunov-Krasovskii functional as

$$V_1[Z_e(t)] = Z_e^T(t) P_1 Z_e(t) + \int_{t-h}^t Z_e^T(s) Q Z_e(s) ds, \tag{98}$$

where $P_1 = P_1^T > 0$, $Q = Q^T > 0$, and a Hamiltonian function as

$$H(Z_e, W, t) = \dot{V}_1[Z_e(t)] + Z_e^T(t) Z_e(t) - \gamma^2 W^T(t) W(t), \tag{99}$$

then we find the time derivative of the Lyapunov-Krasovskii functional as

$$\dot{V}_1[Z_e(t)] = Z_e^T(t)[A_\lambda^T P_1 + P_1 A_\lambda + Q] Z_e(t) + Z_e^T(t-h) K^T P_1 Z_e(t)$$
$$+ Z_e^T(t) P_1 K Z_e(t-h) - Z_e^T(t-h) Q Z_e(t-h) + W^T(t) P_1 Z_e(t) + Z_e^T(t) P_1 W(t). \tag{100}$$

therefore, by plugging $\dot{V}_1[Z_e(t)]$ into the Hamiltonian function, we get

$$H(Z_e, W, t) = Z_e^T(t)[A_\lambda^T P_1 + P_1 A_\lambda + Q] Z_e(t) + Z_e^T(t-h) K^T P_1 Z_e(t)$$
$$+ Z_e^T(t) P_1 K Z_e(t-h) - Z_e^T(t-h) Q Z_e(t-h) + W^T(t) P_1 Z_e(t)$$
$$+ Z_e^T(t) P_1 W(t) + Z_e^T(t) Z_e(t) - \gamma^2 W^T(t) W(t)$$
$$= \hbar_1^T(t) \Lambda_1^* \hbar_1(t), \tag{101}$$

where $\hbar_1(t) \doteq \begin{bmatrix} Z_e^T(t) & Z_e^T(t-h) & W^T(t) \end{bmatrix}^T$ and

$$\Lambda_1^* \doteq \begin{bmatrix} A_\lambda^T P_1 + P_1 A_\lambda + Q + \mathbf{e} & P_1 K & P_1 \\ K^T P_1 & -Q & 0 \\ P_1 & 0 & -\gamma^2 \mathbf{e} \end{bmatrix}. \tag{102}$$

Let $P_1 = P_1^T > 0$, $P_2 = P_1 K < 0$ and $Q = Q^T > 0$ are feasible solutions for $\Lambda_1^* < 0$, then $H(Z_e, W, t) < 0$. And $H(Z_e, W, t) < 0$ implies that $\|T_{Z_e W}\|_\infty < \gamma$, where $\gamma \leq 1/\|A_\Delta\|_\infty$. ∎

3.3.3 Position Controller Design

The velocity subsystem with perturbations is robustly stable under control $\mu'(t) = k_1 Z_e(t - h)$ for arbitrary bounded delays. Fixing k_1, we design k_2, such that $\mu(t) = k_1 Z_e(t - h) + k_2 X_e(t - h)$ can achieve the control goals for the full system. We choose k_2 as a negative definite symmetric matrix and then check whether we can find $Y, \varepsilon_1, \varepsilon_2$, and ε_3 from (85). Since F is known, the inequality (85) becomes and LMI that can be easily solved. Also, since k_1 is fixed, searching for k_2 is quite straightforward.

Note that the first $3(N - 1)$ elements of control signal μ control the formation shape and the last 3 elements control the formation center trajectory. If the desired formation is constant, i.e., $\dot{\rho}_{jd} = 0$ for $j = 1, 2, ..., N - 1$, the first $3(N - 1)$ elements of $A_\Delta([\eta_2^i], [v_2^i]) Z_d$ vanish. In addition, the first $3(N - 1)$ elements of $A_\lambda Z_d$ and $\dot{Z}_d$ also vanish. Therefore, there is **no extra energy** spent to achieve feedback decoupling for the formation shape dynamics. For the formation center control, since $\dot{q}_{cd}$ is usually not zero, energy is required to cancel the last 3 elements of $A_\Delta([\eta_2^i], [v_2^i]) Z_d$.

Following this approach, controllers implemented on each AUV has an inner-outer-loop structure, shown in Figure 19. The orientation control system is the inner loop, where the orientation of each AUV is driven to a desired value, and the controlled steering dynamics is much faster than translational dynamics. In the outer loop, the formation control signal is computed in Jacobi coordinate according to positions and velocities of all AUVs, and implemented through an invertible transformation.

3.4 Simulation

To demonstrate the effectiveness of the proposed controllers, we carry out simulations for a group of six AUVs, the parameters of which are listed as follows [70, 71].

$$m = 200\text{kg}, X_u = -70, Y_v = -100, Z_w = -50, X_{\dot{u}} = -62.5, Y_{\dot{v}} = -70,$$

$$Z_{\dot{w}} = -50, K_p = -0.8, M_q = -0.9, N_r = -0.4, K_{\dot{p}} = -200, M_{\dot{q}} = -350,$$

$$N_{\dot{r}} = -500, I_x = 203\text{Nms}^2, I_y = 587\text{Nms}^2, I_z = 687\text{Nms}^2.$$
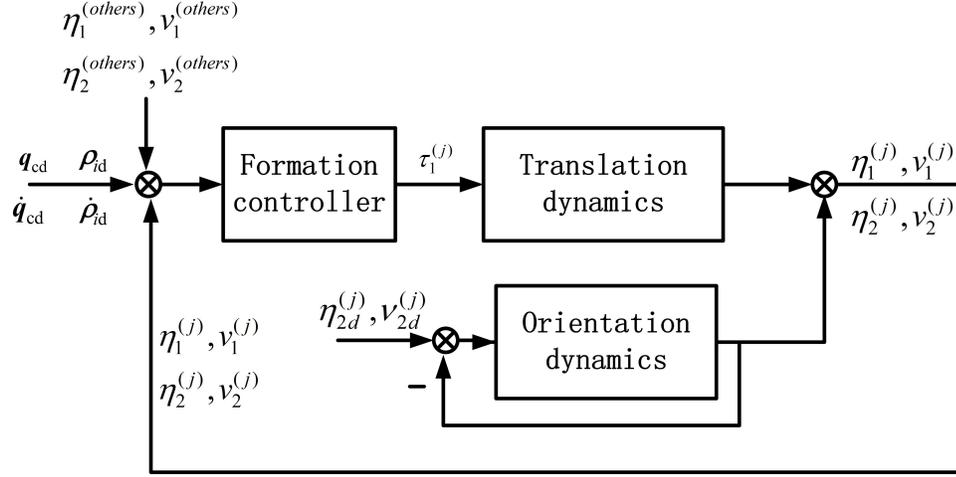
Figure 19: Block diagram for AUV formation control system

The time delay is assumed to be $h^* = 0.5$s.

The simulation is to show six AUVs tracking a sinusoidal line and keeping a hexagon shape. Initial AUV positions are randomly generated in the following interval: $x \in (-10\text{m}, 10\text{m})$, $y \in (-10\text{m}, 10\text{m})$, $z \in (30\text{m}, 50\text{m})$. The desired formation center trajectory is $(5t, 0, 40 + 5\sin t)$. The initial Euler angles of all AUVs are randomly generated around 0.1π and their desired value are $\eta_{2d} = [0.3, 0.1\sin t, 0.1\cos(2t)]^T$. Jacobi vectors are defined as

$$\rho_1 = \frac{1}{\sqrt{2}}(\eta_1^2 - \eta_1^1), \ \rho_2 = \frac{1}{\sqrt{2}}(\eta_1^3 - \eta_1^4), \ \rho_3 = \frac{1}{\sqrt{2}}(\eta_1^5 - \eta_1^6),$$

$$\rho_4 = \frac{1}{2}(\eta_1^4 + \eta_1^3 - \eta_1^1 - \eta_1^2), \ \rho_5 = \frac{1}{4}(\eta_1^1 + \eta_1^2 + \eta_1^3 + \eta_1^4 - 2\eta_1^5 - 2\eta_1^6), \tag{103}$$

and are desired to converge to the following values,

$$\rho_1 \rightarrow [0, 15, 0]^T, \ \rho_2 \rightarrow [0, -15, 0]^T, \ \rho_3 \rightarrow [0, 0, 20]^T, \rho_4 \rightarrow [0, 0, -15]^T, \ \rho_5 \rightarrow [0, 0, 0]^T, \tag{104}$$

which implies that the desired formation shape is a hexagon. During the simulations, a 0.01s time step is used.

To illustrate the effectiveness of both formation controller and orientation controller, we plot the simulation results in Figures 20, 21, and 22. In Figure 20, we plot the trajectories of the six AUVs in black and the trajectory of the formation center in green. The red line is the desired formation center trajectory. We can see that the formation center trajectory converges to the desired sinusoidal curve. We connect the AUV positions with blue lines to highlight the formation shape, which we
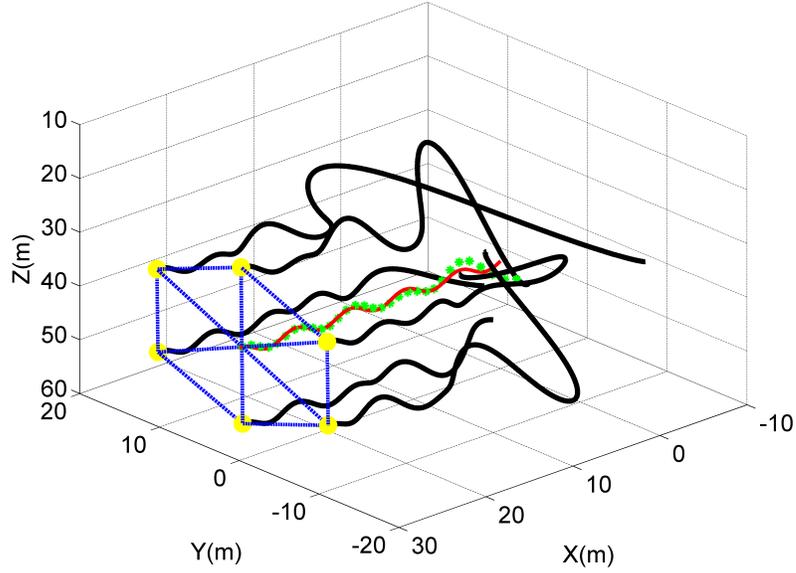
Figure 20: Trajectory and formation of six AUVs.

can see agrees with the desired hexagon. The vertexes of the hexagon are highlighted by yellow dots to show the formation shape. Now Figure 20 suggests that under the formation control, the AUVs form the desired formation and keep this formation while they move along the desired trajectory. To give a close look at the convergence of the Jacobi vectors, we take ρ_2 as an example and plot it in Figure 21, from which we can see that components of ρ_2 converge to their desired values. Other Jacobi vectors also converge to their desired value. To illustrate the convergence of Euler angles of all AUVs, we take yaw angles as an example and plot them in Figures 22, where six colors are used to differentiate the yaw angles for different AUVs. From Figure 22 we can see that the yaw angles all converge to the desired value, i.e., $0.1\cos(2t)$rad as defined in η_{2d}, in about 2 seconds. The roll and pitch angles of all AUVs have the same convergence results. From all the simulation results we can see that our controllers drive each AUV to desired orientation, drive the group of six AUVs to the desired hexagon formation while driving the formation center to the desired sinusoidal curve.

3.5 Conclusions

This chapter presents a robust formation controller design method for AUVs to achieve delay-dependent robust stability. A decoupled design approach is followed to reduce the complexity of
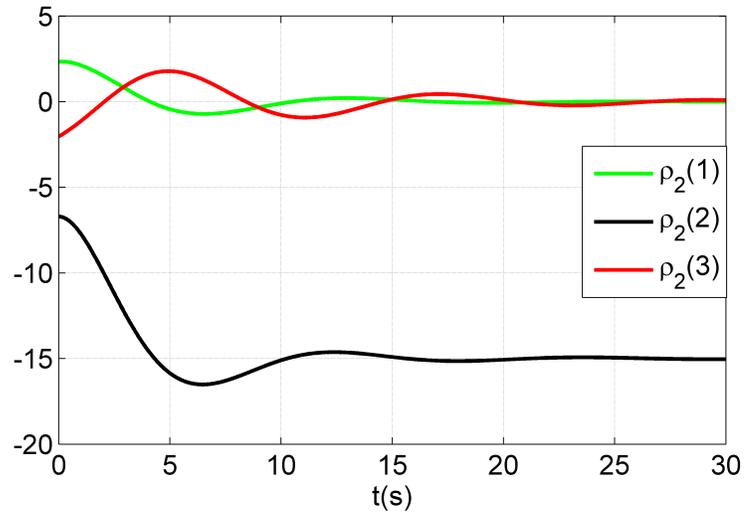
Figure 21: Jacobi vector ρ_2 ($\rho_{2d} = [0, -15, 0]^T$)
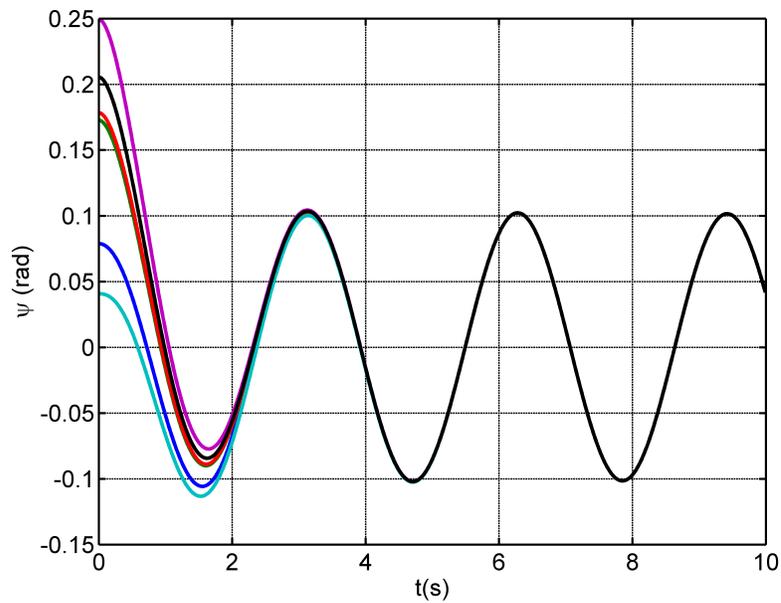
Figure 22: Yaw angles of six AUVs.

the formation control problem caused by 6DOF single AUV dynamics. The Jacobi transform plays an important role to decompose formation shape dynamics and formation center dynamics, so that a simple control design can be pursued.

CHAPTER IV

COOPERATIVE CONTROL OF A HUMAN-AUV SYSTEM

Human and robots are often need to collaborate and interact. As illustrated in [46], a set of semi-autonomous ground vehicles commanded by a human soldier would greatly enhance the militarys flexibility for a range of high-risk scenarios. A group of autonomous underwater vehicles (AUVs) and a human-driven underwater vehicle (HUV) are also often required to collaborate. For example, when an HUV is working on a task that can be performed only under the high intelligence of human, a group of AUVs equipped with tools and devices need to be in specified positions relative to the HUV to assist the human operator. Another example is a situation where an HUV needs to be in some particular position relative to each AUV while they follows a specified path together, so that the human operator of the HUV can acquire most safety. In both cases, the HUV and AUVs need to collaborate by forming a desired formation while moving along a desired trajectory as a team. In our work, we study a system comprised of an HUV and a group of AUVs (Human-AUV system) and focus on cooperative controller design that enables the system to perform formation behavior and execute a curve tracking task.

In human-robot collaboration and interaction, which is an important and growing field ([37–48]), safety of human is an important issue, many results are reported in the literature. For example, in [37] an integrated human-robot strategy to ensure the safety of the human participant is proposed. Using factors influencing the impact force during a human-robot collision, such as the effective robot inertia, the relative distance and velocity between the robot and the human, the level of danger is estimated and used to select a coordinated suite of safety strategies, which make actions of the robot safer and less stressful for the human involved. In [38], an integrated motion synthesis framework designed especially for a robot that interacts with humans is proposed. The framework is composed of Perspective Placement, Human Aware Manipulation Planner and Soft Motion Trajectory Planner, and takes into account human's safety so that the motion of robot is safe to a human. In addition to the safety planning for robots, there are also results in the literature on human's response

in human-robot interaction. In [39], it is shown that hesitation trajectories embodied in an articulated robot arm can be recognized by human observers. In [40], human's emotional arousal is studied based on experiments. Inspired by these work, we consider both the human's response to robots and safety planning of robots in this chapter. Assuming the human operator knows the task of the entire human-AUV system and the role of the HUV in the system, we define an attention function for the human operator to capture the human's response to other vehicles. The attention function is based on the distance between the HUV and AUVs, which is shown related to the safety of the human operator [37]. Based on the attention function, we build a model to describe the motion of an HUV, with which AUVs will be able to predict HUV motions and better interact with the HUV. In this chapter, the safety of the human operator of the HUV is ensured during the controller design, which takes into account constraints of human's attention to avoid scaring the human operator, as well as a mechanism to avoid collisions between the HUV and AUVs.

During the last decade, significant developments have been achieved in formation control of multi-agent systems ([49–55]). In those developments, a common underlying assumption is that the agents in the formation system need to share a common coordinate system, or need to know what coordinate systems other vehicles are using, which requires communications between vehicles. This assumption is difficult to implement to the human-AUV system, as it is not not appropriate to assume that an AUV can estimate to know the coordinate system the human operator is using. In addition, communication between underwater vehicles usually relies heavily on acoustic systems and is plagued with limited communication bandwidth, intermittent failures, latency and multi-path effects. Therefore, we are seeking a cooperative control method so that controllers for each AUV do not rely on frequent communication, but depends only on its sensor measurements, such as the relative positions, velocities, and accelerations of other vehicles, in its own coordinate system. In [31, 91–93], the authors model the entire formation, which is comprised of a group of free particles, as a deformable body. In this work, we use the same methodology to model and decouple the collective dynamics of the Human-AUV system. We apply geometric reduction theory to explicitly decompose the collected motion dynamics of all vehicles into dynamics for the center, the orientation and the shape of the deformable body, so that we can separately take care of the formation center, orientation, and formation shape. We design a gauge invariant formation control law, so

that the control is the same under different gauge convention. This allows each AUV freedom to choose its own coordinate freedom without knowing the coordinate systems used by other vehicles. In addition, the control calculated by each AUV depends only on its measurements of positions, velocities, and accelerations of other vehicles. During the formation shape controller design, we have incorporated mechanisms to avoid scaring the human operator and also to avoid collision between vehicles.

After the collective dynamics decomposition, the formation center can be treated as a free particle and controlled by a curve tracking controller. We have found that some curve tracking techniques for 2 degree-of-freedom smooth curves, including smooth planar curves ([56, 57]) and smooth curves constrained to a sphere surface ([58]), can be extended to apply to a 3 degree-of-freedom smooth curve, i.e., any smooth curve in 3D space. Based on those technologies, in this chapter we develop a curve tracking controller to enable a free particle to track any smooth curve in 3D space, and use it to control the formation center.

The remainder of this chapter is organized as follows. Section 4.1 introduces the dynamics of AUVs and an HUV. In Section 4.2, we decouple the collected dynamics into a formation center, a formation shape, and a formation orientation. In Section 4.3, we design a curve tracking controller for the formation center to track any smooth curve in 3D space. In Section 4.4, we design the formation shape and orientation controller to enable the human-AUV system to converge to a desired formation shape. An example for a system comprised by one HUV and two AUVs and the corresponding simulation results are presented in Section 4.5. We give the conclusion in Section 4.6.

4.1 Vehicle Dynamics

For a system comprised of one HUV and $N-1$ AUVs, we use $\mathbf{r}_1, ... \mathbf{r}_{N-1}, \mathbf{r}_h$ to denote the positions of all the vehicles, where subscript h indicates the HUV with mass m_h and subscripts $1, ..., N-1$ indicates the $N-1$ AUVs. The mass of the i-th AUV is m_i.

4.1.1 AUV dynamics

We use $\mathbf{f}_i$ to denote the force applied on the the i-th AUV, where $i = 1, ..., N-1$, then Newton's law suggests that

$$m_i \ddot{\mathbf{r}}_i = \mathbf{f}_i. \tag{105}$$

Here we use $\ddot{()}$ to denote the second order derivative of the variable in the parenthesis with respect to time, i.e., $\frac{d^2}{dt^2}()$. Similarly, we will use $\dot{()}$ to denote $\frac{d}{dt}()$.

4.1.2 HUV dynamics

For HUV, we also apply Newton's law and get

$$\mathbf{f}_h = m_h \ddot{\mathbf{r}}_h. \tag{106}$$

Here $\mathbf{f}_h$ is the force applied on the HUV and $\ddot{\mathbf{r}}_h$ is the acceleration of the HUV. Unlike forces applied on AUV, i.e., $\mathbf{f}_i$, $i = 1, ..., N-1$, which are control forces calculated by our controller and implemented by each AUV, the forces applied on HUV, i.e., $\mathbf{f}_h$, is the result of the HUV acceleration, which is adjusted by the human operator. We assume the human operator of the HUV knows the desired formation all the vehicles need to form and pays attention to AUV motions when cooperates with AUVs. To form the desired formation with AUVs while moving with them as a team, the human operator has to drive the vehicle in a way so that the HUV tracks the movements of AUVs and meanwhile adjusts its position relative to the AUVs. Therefore, the acceleration of the HUV can be decomposed into two parts: one part, denoted by $(\ddot{\mathbf{r}}_h)_1$, generates the motion to follow all AUVs, the other part, denoted by $(\ddot{\mathbf{r}}_h)_2$, adjust position of the HUV relative to AUVs. Therefore,

$$\ddot{\mathbf{r}}_h = (\ddot{\mathbf{r}}_h)_1 + (\ddot{\mathbf{r}}_h)_2. \tag{107}$$

It's reasonable to assume

$$(\ddot{\mathbf{r}}_h)_1 = \frac{\sum_{i=1}^{N-1} \ddot{\mathbf{r}}_i}{N-1}. \tag{108}$$

It's easy to see that in transient state, the HUV is operated to track the average acceleration of all AUVs, and in steady state, $(\ddot{\mathbf{r}}_h)_1$ will be the same as acceleration of each AUV, as the acceleration of all AUVs will be identical in steady state under our control law.

When cooperate with AUVs, the human operator of HUV pays attention to AUVs. It is reasonable to assume that the human operator will pay more attention to an AUV if the AUV is closer to the HUV, and pay less attention to the AUV if the AUV is further from the HUV. To describe this property, we define the following attention function $\mathscr{A}_i$ to describe how much attention the human operator pays to the i-th AUV:

$$\mathscr{A}_i = \frac{\mathscr{A}_{i0} l_{i0}}{l_i} \tag{109}$$

where l_i is the distance between the HUV and the i-th AUV. l_{i0} and $\mathscr{A}_{i0}$ are constant parameters. l_{i0} is the distance in desired formation and $\mathscr{A}_{i0}$ is the attention when $l_i = l_{i0}$.

Attention is related to how closely the HUV operator is watching an AUV. An attention around 0 means that the human operator is uninterested in the AUV or does not notice the AUV. As the attention increases, the human operator of the HUV will be interested in the AUV and goes towards the AUV, and a bigger attention will cause a bigger acceleration towards the AUV. If the attention is very large, the human operator of the HUV will be fearful of the AUV and go away from it. To describe this property, we define a relative acceleration of the HUV towards the i-th AUV as following:

$$a_i = k_i \mathscr{A}_i (\mathscr{A}_{i0} - \mathscr{A}_i) \frac{\mathbf{r}_i - \mathbf{r}_h}{\|\mathbf{r}_i - \mathbf{r}_h\|}. \tag{110}$$

Here $\| \cdot \|$ is the magnitude of a vector, so $\frac{\mathbf{r}_i - \mathbf{r}_h}{\|\mathbf{r}_i - \mathbf{r}_h\|}$ specifies the direction of the relative acceleration, which is pointing to the i-th AUV from the HUV. The term $k_i \mathscr{A}_i (\mathscr{A}_{i0} - \mathscr{A}_i)$ gives the magnitude of the relative acceleration, where k_i is a positive scalar. Note that when $\mathscr{A}_i = \mathscr{A}_{i0}$, i.e., when the distance between the HUV and the i-th AUV is the same with the one in desired formation, $a_i = 0$, which means the relative acceleration of the HUV towards the i-th AUV is zero, so they will stay at the desired formation. To avoid scaring of the human operator of the HUV, the attention function $\mathscr{A}_i$ of the human operator can not be too large, so we assume

$$\mathscr{A}_i < \mathscr{A}_{i,max}. \tag{111}$$

need to be satisfied all the time. To form and keep the desired formation, the human operator will adjust HUV position through the relative HUV acceleration to all AUVs, so we assume

$$(\ddot{\mathbf{r}}_h)_2 = \sum_{i=1}^{N-1} a_i. \tag{112}$$

51

The acceleration $(\ddot{\mathbf{r}}_h)_2$ will adjust the position of the HUV until all vehicles are in the desired formation. It's easy to see $(\ddot{\mathbf{r}}_h)_2 = 0$ in desired formation, i.e., $(\ddot{\mathbf{r}}_h)_2$ vanishes in steady state.

We can see that the motion of the HUV depends on the motion of the AUVs, while the motion of the AUVs is controlled by the forces applied on them that we will design in the following sections. Given the dependency of HUV motions on AUV motions we just formulated, our objective is to control the forces applied on AUVs, so that the HUV and AUVs form a desired formation and the formation center tracks a desired smooth curve in 3D space, while the distance between vehicles is keeping above some thresholds to avoid vehicle collisions and attentions of the human operator are keeping below some thresholds to avoid human being scared. As we assume no communication between vehicles and each vehicle use its own coordinate system, the designed control force for each vehicle will be based on only sensor measurements from that vehicle in its own coordinate system.

4.2   Collective Dynamics Decomposition

In this section, we decouple the collective dynamics of the human-AUV system. First, by the Jacobi transform [31], the formation shape and orientation dynamics can be decoupled from the dynamics of the center of the formation. We view the entire formation as a deformable body. Let $M = m_h + \sum_{i=1}^{N-1} m_i$, then the center of mass is $q_c = \frac{m_h \mathbf{r}_h + \sum_{i=1}^{N-1} m_i \mathbf{r}_i}{M}$, which describe the position of the entire formation and does not affect the formation shape or orientation. Then the $N-1$ independent Jacobi vectors ($\rho_{fi}, i = 1, ... N-1$) defined below make the kinetic energy of the cluster, originally expressed as $K^{tot} = \frac{1}{2} \sum_{i=1}^{N-1} m_i \|\dot{\mathbf{r}}_i\|^2 + m_h \|\dot{\mathbf{r}}_h\|$, block diagonalized as $K^{tot} = \frac{1}{2} M \|\dot{q}_c\| + \frac{1}{2} \sum_{i=1}^{N-1} \|\dot{\rho}_{fi}\|^2$.

$$[\mathbf{r}_c, \rho_{f1}, ..., \rho_{f(N-1)}] = [\mathbf{r}_1, \mathbf{r}_2, ..., \mathbf{r}_{N-1}, \mathbf{r}_h] \Phi \tag{113}$$

where the linear invertible transform Φ is called Jacobi transform, expressed as a $(3N \times 3N)$ matrix. The Jacobi transform Φ is guaranteed to exist. One implementation is

$$\Phi = \begin{bmatrix} \Phi_{11} & \Phi_{12} & \ldots & \Phi_{1N} \\ \vdots & \vdots & \vdots & \vdots \\ \Phi_{i1} & \Phi_{i2} & \ldots & \Phi_{iN} \\ \vdots & \vdots & \vdots & \vdots \\ \Phi_{N1} & \Phi_{N2} & \ldots & \Phi_{NN} \end{bmatrix} \tag{114}$$

$$\begin{aligned} \Phi_{i1} &= \frac{m_i}{M} \mathbf{e_3} \ (i = 1, 2, ...N) \\ \Phi_{ii} &= \sqrt{\mu_{i-1}} \mathbf{e_3} \ (i > 1) \\ \Phi_{ij} &= -\sqrt{\mu_{j-1}} \frac{m_i}{\sum_{k=1}^{j-1} m_k} \mathbf{e_3} \ (j > 1; i < j) \\ \Phi_{ij} &= \mathbf{0}_{3 \times 3} \ (j > 1; i > j) \end{aligned} \tag{115}$$

where $\mathbf{e}$ is an identity matrix with dimension specified by the subscript, and

$$\frac{1}{\mu_i} = \frac{1}{\sum_{k=1}^i m_k} + \frac{1}{m_{i+1}} \text{ for } i = 1, 2, ...N - 1. \tag{116}$$

Let $\mathscr{R}^{3N}$ be the total configuration space of a formation of N particles in 3D space. After the coordinate of the center $\mathbf{r}_c$ is removed, the space of the Jacobi coordinates is $\mathscr{R}^{3N-3}$. Since the shape of the formation is independent of its orientation, we can view this symmetry as induced by the rigid rotation group SO(3) acting on $\mathscr{R}^{3N-3}$ to the left as $g\rho_{fi}$ for $g \in$ SO(3) and $j = 1, 2, ...N - 1$. After the rotational symmetry is removed from the Jacobi coordinates, we can use $3N - 6$ scalar variables, called shape variables, to describe a shape. The shape variables s_j for $j = 1, 2, ..., (3N - 6)$ are rigid motion invariant. Candidates for s_j are functions of dot products $(\rho_{fi} \cdot \rho_{fk})$ and triple products $(\rho_{fi} \cdot (\rho_{fk} \cdot \rho_{fl}))$, thus, mutual distances, mutual angles, areas and volume formed by the line segments connecting the particles all serves as candidates for shape variables [31]. For a given shape $\mathbf{s} = [s_1, s_2, ..., s_{3N-6}]$, we can measure its orientation by attaching a body coordinate system to the formation, and obtain a group element $g \in$ SO(3) as following:

$$g = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi c\phi s\theta \\ s\psi c\theta & c\psi c\phi + s\phi s\theta s\psi & -c\psi s\phi + s\theta s\psi c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix}$$

Here, $s\cdot = \sin(\cdot)$ and $c\cdot = \cos(\cdot)$. ϕ, θ, and ψ represent the Euler angles for roll, pitch, and yaw, respectively. Now the Jacobi vectors in the body coordinate system, defined as

$$\rho_i = g\rho_{fi}, \tag{117}$$

are vector valued function of the shape variables $\mathbf{s}$ only. Then we can get $\rho_{fi} = g(\Omega \times \rho_i + \sum_{j=1}^{3N-6} \frac{\partial \rho_i}{\partial s_j} \dot{s}_j)$.

Using transforms in Equations (113) and (117), we can decouple the collective dynamics of all the vehicles into formation center dynamics, formation shape dynamics and orientation dyanmics, in the following three equations [31]:

$$M\ddot{\mathbf{r}}_c = \mathbf{u}_c \tag{118}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}(I\Upsilon) = -\Omega \times I\Upsilon + \mathbf{u}_g \tag{119}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}(G\dot{\mathbf{s}}) + \mathbf{A}^T\frac{\mathrm{d}}{\mathrm{d}t}(I\Upsilon) = \frac{1}{2}[\frac{\partial I}{\partial \mathbf{s}}]^* : (\Upsilon, \Upsilon) + ([\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]^* - [\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]) : (\dot{\mathbf{s}}, I\Upsilon) + \frac{1}{2}[\frac{\partial G}{\partial \mathbf{s}}]^* : (\dot{\mathbf{s}}, \dot{\mathbf{s}}) + \mathbf{u}_s \tag{120}$$

where

$$
\begin{aligned}
I(\mathbf{s}) &= \sum_{i=1}^{N-1} (\|\rho_1\|^2 \mathbf{e} - \rho_i \rho_i^T) \\
A_j(\mathbf{s}) &= I^{-1} \sum_{i=1}^{N-1} \rho_i \times \frac{\partial \rho_i}{\partial s_j} \\
\mathbf{A} &= [A_1, A_2, ..., A_{3N-6}] \\
G_{jk} &= -A_j^T I A_k + \sum_{i=1}^{N-1} \frac{\partial \rho_i^T}{\partial s_j} \frac{\partial \rho_i}{\partial s_k} \\
\Upsilon &= \Omega + \sum_{i=1}^{3N-6} A_j \dot{s}_j \\
J &= I\Upsilon
\end{aligned}
\tag{121}
$$

The control on formation center, formation shape, and orientation, i.e., $\mathbf{u}_c$, $\mathbf{u}_s$, and $\mathbf{u}_g$, respectively, have the following one-one correspondence with forces applied on the vehicles, i.e., $\mathbf{f}_1,...,\mathbf{f}_N - 1, \mathbf{f}_h$.

$$
\begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_{N-1} \\ \mathbf{f}_h \end{bmatrix} = \Phi \begin{bmatrix} \mathbf{u}_c \\ \mathbf{u}_{f_1} \\ \vdots \\ \mathbf{u}_{f_{N-1}} \end{bmatrix}, \tag{122}
$$

54

and

$$\mathbf{u}_g = \sum_{j=1}^{N-1} \rho_j \times g^{-1} \mathbf{u}_{f_j},$$

$$\mathbf{u}_{sk} = \sum_{j=1}^{N-1} (\frac{\partial \rho_j}{\partial s_k})^T \times g^{-1} \mathbf{u}_{f_j}. \tag{123}$$

Equation (123) is equivalent to

$$\begin{bmatrix} \mathbf{u}_g \\ \mathbf{u}_s \end{bmatrix} = \Psi \begin{bmatrix} \mathbf{u}_{f_1} \\ \vdots \\ \mathbf{u}_{f_{N-1}} \end{bmatrix} \tag{124}$$

where

$$\Psi = \begin{bmatrix} \widehat{\rho_1} g^{-1} & \widehat{\rho_2} g^{-1} & \cdots & \widehat{\rho_{N-1}} g^{-1} \\ \frac{\partial \rho_1}{\partial s_1}^T g^{-1} & \frac{\partial \rho_2}{\partial s_1}^T g^{-1} & \cdots & \frac{\partial \rho_{N-1}}{\partial s_1}^T g^{-1} \\ \frac{\partial \rho_1}{\partial s_2}^T g^{-1} & \frac{\partial \rho_2}{\partial s_2}^T g^{-1} & \cdots & \frac{\partial \rho_{N-1}}{\partial s_2}^T g^{-1} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial \rho_1}{\partial s_{3N-6}}^T g^{-1} & \frac{\partial \rho_2}{\partial s_{3N-6}}^T g^{-1} & \cdots & \frac{\partial \rho_{N-1}}{\partial s_{3N-6}}^T g^{-1} \end{bmatrix} \tag{125}$$

and $\widehat{(\cdot)}$ denotes the skew symmetric matrix representation of vector $(\cdot)$, i.e., for vector $\omega \doteq [\omega_1, \omega_2, \omega_3]$,

$$\widehat{\omega} \doteq \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}.$$

Equations (122) and (124) can be combined into

$$\begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_{N-1} \\ \mathbf{f}_h \end{bmatrix} = \Gamma \begin{bmatrix} \mathbf{u}_c \\ \mathbf{u}_g \\ \mathbf{u}_s \end{bmatrix} \tag{126}$$

where

$$\Gamma = \Phi \begin{bmatrix} \mathbf{e}_3 & \mathbf{0}_{3 \times (3N-3)} \\ \mathbf{0}_{(3N-3) \times 3} & \Psi^{-1} \end{bmatrix}. \tag{127}$$

Writing Γ in matrix blocks as $\Gamma \doteq \begin{bmatrix} \Gamma_{11} & \Gamma_{12} & \Gamma_{13} \\ \Gamma_{21} & \Gamma_{22} & \Gamma_{23} \end{bmatrix}$, where $\Gamma_{11}, \Gamma_{12}, \Gamma_{13}, \Gamma_{21}, \Gamma_{22}$, and Γ_{23} are

$(3N-3) \times 3$, $(3N-3) \times 3$, $(3N-3) \times (3N-6)$, 3×3, 3×3, and $3 \times (3N-6)$ matrix blocks of Γ,

respectively, we can get

$$
\begin{bmatrix} \mathbf{f}_1 \\ \vdots \\ \mathbf{f}_{N-1} \end{bmatrix} = \Gamma_{11}\mathbf{u}_c + \Gamma_{12}\mathbf{u}_g + \Gamma_{13}\mathbf{u}_s \tag{128}
$$

$$
\mathbf{f}_h = \Gamma_{21}\mathbf{u}_c + \Gamma_{22}\mathbf{u}_g + \Gamma_{23}\mathbf{u}_s \tag{129}
$$

Now we can see that if we can design formation center control $\mathbf{u}_c$, formation shape control $\mathbf{u}_s$, and formation orientation control $\mathbf{u}_g$ under constraint in Equation (129), the forces applied on each AUV can be calculated according to Equation (128). Note Γ_{22} is not invertible in general, so we cannot solve $\mathbf{u}_g$ as a function of $\mathbf{u}_c$, $\mathbf{u}_s$, and $\mathbf{f}_h$, and then plug it into Equation (128) to calculate the forces applied on each AUV. In other words, $\mathbf{u}_c$ and $\mathbf{u}_s$ are coupled, so if an arbitrary force applied on the HUV, we cannot achieve formation shape control and curving tracking control for the formation center at the same time. They can be achieved at the same time only when the human operator of the HUV is notified the required tasks and cooperates with AUVs.

Assuming the human operator of the HUV knows the desired formation all the vehicles need to form and pays attention to AUV motions, we assumed the HUV dynamics in Equation (107), (108), and (112). Plug Equation (105) into (108), we get $(\ddot{\mathbf{r}}_h)_1 = \frac{\mathbf{f}_1 + \cdots + \mathbf{f}_{N-1}}{N-1}$. According to Equations (113) and (117), and the fact that ρ_i, i.e., the Jacobi vectors in the body coordinate system, is a vector valued function of the shape variables $\mathbf{s}$ only, $(\ddot{\mathbf{r}}_h)_2$ in Equation (112) is converted to a function of only the shape variable $\mathbf{s}$ and the orientation g, denoted by

$$
(\ddot{\mathbf{r}}_h)_2 = \chi(\mathbf{s}, g). \tag{130}
$$

We will give χ for an example system comprised of one HUV and two AUVs in Section 4.5. Now the HUV dynamics is rewritten as

$$
\mathbf{f}_h = \frac{\mathbf{f}_1 + \cdots + \mathbf{f}_{N-1}}{N-1} + \chi(\mathbf{s}, g). \tag{131}
$$

From Equation (113), we have

$$
\mathbf{u}_c = \frac{\mathbf{f}_1 + \cdots + \mathbf{f}_{N-1} + \mathbf{f}_h}{N}. \tag{132}
$$

In addition, its easy to calculate Γ_{21} in Equations (126) and (129) and get

$$
\Gamma_{21} = \frac{1}{N}\mathbf{e_3} \tag{133}
$$

Plug Equations (132), (133), and (131) into Equation (129), we get Equation (129) is equivalent to

$$\Gamma_{22}\mathbf{u}_g + \Gamma_{23}\mathbf{u}_s - \frac{N-1}{N}\chi(\mathbf{s},g) = 0 \tag{134}$$

In next two chapters, we will design $\mathbf{u}_c$ to drive the formation center to desired trajectory, and design $\mathbf{u}_s$ and $\mathbf{u}_g$ under the constraint in Equation (134) to drive the formation shape to a desired one. Once we design $\mathbf{u}_c$, $\mathbf{u}_s$, and $\mathbf{u}_g$, the forces applied on each AUV can be calculated according to Equation (128).

4.3 Curve Tracking Controller for Formation Center

In the previous section, the formation center is decoupled from the formation shape and orientation, so that we can view the formation center as a free particle. In this section, we design a curve tracking control law for the formation center, and the control law enables a free particle to track any arbitrary smooth curve in the 3D space.

We adopt the natural Frenet frame to describe any smooth curve in 3D space. Since we only care about the shape of a 3D curve, we assume it is a unit-speed curve, i.e., $\|\frac{d\mathbf{r}}{dt}\| = 1$, where $\mathbf{r}$ denotes a point on the curve. Define $\mathbf{X}$ to be the unit tangent vector, i.e., $\mathbf{X} = \frac{d\mathbf{r}}{dt}$, $\mathbf{Y}$ and $\mathbf{Z}$ to be unit vectors that make $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ a right-handed orthonormal frame, the natural Frenet frame representation of a 3D curve is as follows:

$$\begin{bmatrix} \dot{\mathbf{X}} \\ \dot{\mathbf{Y}} \\ \dot{\mathbf{Z}} \end{bmatrix} = \begin{bmatrix} 0 & \widetilde{v} & \widetilde{u} \\ -\widetilde{v} & 0 & 0 \\ -\widetilde{u} & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \\ \mathbf{Z} \end{bmatrix}, \tag{135}$$

where $\widetilde{u}$ and $\widetilde{v}$ are natural curvatures. Unlike the Frenet-Serret frame, which can not frame a curve with $\|\frac{d^2\mathbf{r}}{dt^2}\| = 0$, the above natural Frenet frame can describe any smooth curve in 3D space. Note that the natural Frenet frame of a curve is not unique. For a curve with $\|\frac{d^2\mathbf{r}}{dt^2}\| = 0$, we have $\widetilde{u} = 0$ and $\widetilde{v} = 0$, and any unit vectors $\mathbf{Y}$ and $\mathbf{Z}$ that make $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ a right-handed orthonormal frame will be valid. For a curve with $\|\frac{d^2\mathbf{r}}{dt^2}\| \neq 0$, a family of valid Natural Frenet frames can be determined from the unique Frenet-Serret frame representation. The implementation is provided in Appendix D.

Using the above Natural Frenet frame representation, we can describe any smooth curve in 3D space a particle needs to track. In similarly way, we can describe the actual trajectory of the particle.

We list them below:

$$\dot{\mathbf{r}}_c = \mathbf{X}_c \tag{136}$$

$$\dot{\mathbf{X}}_c = \widetilde{v}_c \mathbf{Y_c} + \widetilde{u}_c \mathbf{Z}_c \tag{137}$$

$$\dot{\mathbf{Y}}_c = -\widetilde{v}_c \mathbf{X}_c \tag{138}$$

$$\dot{\mathbf{Z}}_c = -\widetilde{u}_c \mathbf{X}_c \tag{139}$$

$$\dot{\mathbf{r}} = \mathbf{X} \tag{140}$$

$$\dot{\mathbf{X}} = \widetilde{v}\mathbf{Y} + \widetilde{u}\mathbf{Z} \tag{141}$$

$$\dot{\mathbf{Y}} = -\widetilde{v}\mathbf{X} \tag{142}$$

$$\dot{\mathbf{Z}} = -\widetilde{u}\mathbf{X} \tag{143}$$

The first four equations with all terms subscripted by "c" belongs to the free particle, and last four equations with all terms having no subscript belongs to the desired curve the particle is going to track. $\mathbf{r}_c$ denotes the trajectory of the free particle and $\mathbf{r}$ denotes the closest point on the desired 3D curve to the free particle.

We define ρ as the closest distance between the formation center and the curve it tracks, and define l as the displacement from the formation center to the 3D curve, i.e., $l = \mathbf{r_c} - \mathbf{r}$, then $\rho = \|l\|$. We define $\widetilde{\theta}$ is the angle between $\mathbf{X}$ and $\mathbf{X_c}$. We will design control law u_c and v_c for the particle so that its trajectory converges to the desired one, i.e., $\rho \to 0$ and $\widetilde{\theta} \to 0$

As $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{Z}$ are orthonormal basis of $\mathbf{R}^3$, we can express $\dot{\mathbf{X}}_c$ by linear combination of $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{Z}$ as

$$\dot{\mathbf{X}}_c = a\mathbf{X} + b\mathbf{Y} + c\mathbf{Z}. \tag{144}$$

From Equation (137) we get

$$\widetilde{u}_c = \dot{\mathbf{X}}_c \cdot \mathbf{Z_c} \tag{145}$$

$$\widetilde{v}_c = \dot{\mathbf{X}}_c \cdot \mathbf{Y_c} \tag{146}$$

Therefore, $\widetilde{u}_c$ and $\widetilde{u}_c$ can be express as

$$\widetilde{u}_c = a\mathbf{X} \cdot \mathbf{Z_c} + b\mathbf{Y} \cdot \mathbf{Z_c} + c\mathbf{Z} \cdot \mathbf{Z_c}, \tag{147}$$

$$\widetilde{v}_c = a\mathbf{X} \cdot \mathbf{Y_c} + b\mathbf{Y} \cdot \mathbf{Y_c} + c\mathbf{Z} \cdot \mathbf{Y_c}. \tag{148}$$

The design of $\widetilde{u}_c$ and $\widetilde{v}_c$ becomes finding the parameters $(a, b, c.)$

Define $g_1 = [\mathbf{X}, \mathbf{Y}, \mathbf{Z}]$ and $g_2 = [\mathbf{X_c}, \mathbf{Y_c}, \mathbf{Z_c}]$. From the fact that $(g_1^T g_2)(g_1^T g_2)^T = \mathbf{e}$, we get

$$(\mathbf{X} \cdot \mathbf{Y_c})^2 + (\mathbf{X} \cdot \mathbf{Z_c})^2 = 1 - (\mathbf{X} \cdot \mathbf{X_c})^2,$$

$$\mathbf{Y} \cdot \mathbf{Y_c} \mathbf{X} \cdot \mathbf{Y_c} + \mathbf{Y} \cdot \mathbf{Z_c} \mathbf{X} \cdot \mathbf{Z_c} = -\mathbf{Y} \cdot \mathbf{X_c} \mathbf{X} \cdot \mathbf{X_c},$$

$$\mathbf{Z} \cdot \mathbf{Y_c} \mathbf{X} \cdot \mathbf{Y_c} + \mathbf{Z} \cdot \mathbf{Z_c} \mathbf{X} \cdot \mathbf{Z_c} = -\mathbf{Z} \cdot \mathbf{X_c} \mathbf{X} \cdot \mathbf{X_c}. \tag{149}$$

From Equations (147) and (148), we obtain

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}(\cos\widetilde{\theta}) &= \frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{X} \cdot \mathbf{X_c}) \\
&= \dot{\mathbf{X}} \cdot \mathbf{X_c} + \mathbf{X} \cdot \dot{\mathbf{X}}_\mathbf{c} \\
&= (\widetilde{v}\mathbf{Y} + \widetilde{u}\mathbf{Z}) \cdot \mathbf{X_c} + \mathbf{X} \cdot (\widetilde{v}_c \mathbf{Y_c} + \widetilde{u}_c \mathbf{Z_c}) \\
&= \widetilde{v}\mathbf{Y} \cdot \mathbf{X_c} + \widetilde{u}\mathbf{Z} \cdot \mathbf{X_c} + \dot{\mathbf{X}}_c \cdot \mathbf{Y_c} \mathbf{X} \cdot \mathbf{Y_c} + \dot{\mathbf{X}}_c \cdot \mathbf{Z_c} \mathbf{X} \cdot \mathbf{Z_c} \\
&= \widetilde{v}\mathbf{Y} \cdot \mathbf{X_c} + \widetilde{u}\mathbf{Z} \cdot \mathbf{X_c} + a[(\mathbf{X} \cdot \mathbf{Y_c})^2 + (\mathbf{X} \cdot \mathbf{Z_c})^2] \\
&\quad + b[\mathbf{Y} \cdot \mathbf{Y_c} \mathbf{X} \cdot \mathbf{Y_c} + \mathbf{Y} \cdot \mathbf{Z_c} \mathbf{X} \cdot \mathbf{Z_c}] \\
&\quad + c[\mathbf{Z} \cdot \mathbf{Y_c} \mathbf{X} \cdot \mathbf{Y_c} + \mathbf{Z} \cdot \mathbf{Z_c} \mathbf{X} \cdot \mathbf{Z_c}].
\end{aligned}
$$

Applying properties in Equation (149), we get

$$\frac{\mathrm{d}}{\mathrm{d}t}(\cos\widetilde{\theta}) = \widetilde{v}\mathbf{Y} \cdot \mathbf{X_c} + \widetilde{u}\mathbf{Z} \cdot \mathbf{X_c} + a[1 - (\mathbf{X} \cdot \mathbf{X_c})^2] - b\mathbf{Y} \cdot \mathbf{X_c} \mathbf{X} \cdot \mathbf{X_c} - c\mathbf{Z} \cdot \mathbf{X_c} \mathbf{X} \cdot \mathbf{X_c}. \tag{150}$$

Later we will use this term to construct Lyapunov function for the curve tracking controller design.

As $\mathbf{r}$ is the closest point, which leads to $l \cdot \mathbf{X} = 0$ (otherwise, ρ will decrease with rate $l \cdot \mathbf{X}$ if the formation center goes in either $\mathbf{X}$ or $-\mathbf{X}$ direction, in other words, ρ is not shortest distance and $\mathbf{r}$ is not the closest point), or equivalently

$$l \cdot \dot{\mathbf{r}} = 0. \tag{151}$$

In addition, $\rho = \|l\|$, which is equivalent to $\rho = (l \cdot l)^{\frac{1}{2}}$, leads to

$$2\rho\dot{\rho} = 2l \cdot \dot{l} = 2l \cdot (\dot{\mathbf{r}}_\mathbf{c} - \dot{\mathbf{r}}). \tag{152}$$

Equations (151) and (152) lead to

$$2\rho\dot{\rho} = 2l \cdot \dot{\mathbf{r}}_\mathbf{c}. \tag{153}$$

As $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{Z}$ are orthonormal basis of $\mathbf{R}^3$, we can express l by linear combination of $\mathbf{X}$, $\mathbf{Y}$, and $\mathbf{Z}$, i.e., $l = (l \cdot \mathbf{X})\mathbf{X} + (l \cdot \mathbf{Y})\mathbf{Y} + (l \cdot \mathbf{Z})\mathbf{Z}$, which simplifies to the follow equation because of the fact $l \cdot \mathbf{X} = 0$:

$$l = \frac{d}{2}\mathbf{Y} + \frac{e}{2}\mathbf{Z}, \tag{154}$$

where

$$d \doteq 2l \cdot \mathbf{Y}, \tag{155}$$

$$e \doteq 2l \cdot \mathbf{Z}. \tag{156}$$

Now Equation (154) further simplifies $2\rho\dot{\rho}$ in Equation (153) to

$$2\rho\dot{\rho} = d\mathbf{Y} \cdot \mathbf{X_c} + e\mathbf{Z} \cdot \mathbf{X_c}. \tag{157}$$

Equation (157) will be used to construct Lyapunov function for the curve tracking controller design later.

Define the following Lyapunov candidate function

$$V_1 = -\ln(\cos\tilde{\theta}) + \rho^2, \tag{158}$$

where the first term aims to align the moving direction of the particle with the tangent direction of the desired curve, the second term serves to control the particle to go to and stay on the desired curve. Take derivative of the Lyapunov candidate function V_1, we get then

$$\dot{V}_1 = -\frac{1}{\mathbf{X} \cdot \mathbf{X_c}}(\dot{\mathbf{X}} \cdot \mathbf{X_c} + \mathbf{X} \cdot \dot{\mathbf{X}}_c) + 2\rho\dot{\rho} \tag{159}$$

Applying Equations (150) and (157), we get

$$\dot{V}_1 = (\frac{-v}{\mathbf{X} \cdot \mathbf{X_c}} + b + d)\mathbf{Y} \cdot \mathbf{X_c} + (\frac{-u}{\mathbf{X} \cdot \mathbf{X_c}} + c + e)\mathbf{Z} \cdot \mathbf{X_c} + \frac{-a(1 - (\mathbf{X} \cdot \mathbf{X_c})^2)}{\mathbf{X} \cdot \mathbf{X_c}} \tag{160}$$

Choose

$$a = k_3\mathbf{X} \cdot \mathbf{X_c}(1 - \mathbf{X} \cdot \mathbf{X_c}) \tag{161}$$

$$b = \frac{v}{\mathbf{X} \cdot \mathbf{X_c}} - d \tag{162}$$

$$c = \frac{u}{\mathbf{X} \cdot \mathbf{X_c}} - e \tag{163}$$

60

and plug them into $\widetilde{u}_c$ and $\widetilde{v}_c$, we get

$$\widetilde{u}_c = k_3(\mathbf{X} \cdot \mathbf{X_c})(1 - \mathbf{X} \cdot \mathbf{X_c})(\mathbf{X} \cdot \mathbf{Z_c}) + (\frac{\widetilde{v}}{\mathbf{X} \cdot \mathbf{X_c}} - d)(\mathbf{Y} \cdot \mathbf{Z_c}) + (\frac{\widetilde{u}}{\mathbf{X} \cdot \mathbf{X_c}} - e)(\mathbf{Z} \cdot \mathbf{Z_c}),$$

$$\widetilde{v}_c = k_3(\mathbf{X} \cdot \mathbf{X_c})(1 - \mathbf{X} \cdot \mathbf{X_c})(\mathbf{X} \cdot \mathbf{Y_c}) + (\frac{\widetilde{v}}{\mathbf{X} \cdot \mathbf{X_c}} - d)(\mathbf{Y} \cdot \mathbf{Y_c}) + (\frac{\widetilde{u}}{\mathbf{X} \cdot \mathbf{X_c}} - e)(\mathbf{Z} \cdot \mathbf{Y_c}). \tag{164}$$

Plug a, b, and c into Equation (160), we get

$$\dot{V}_1 = -k_3(1 - (\mathbf{X} \cdot \mathbf{X_c})^2)(1 - \mathbf{X} \cdot \mathbf{X_c}) \le 0, \tag{165}$$

Theorem 3 *Under control law (164), given $\widetilde{\theta} \in (-\frac{\pi}{2}, \frac{\pi}{2})$ initially, trajectory of the particle in Equations (136-139) converges to the desired 3D curve in Equations (140-143), i.e., $\rho \to 0$ and $\widetilde{\theta} \to 0$.*

Proof Consider the closed-loop dynamics described by Equations (136-143) and (169). According to Equation (165), $\dot{V}_1 = 0$ leads to $\mathbf{X} \cdot \mathbf{X_c} = \pm 1$, i.e., $\widetilde{\theta} = 0$ or $\widetilde{\theta} = \pi$. Later we will show that if $\widetilde{\theta} \in (-\frac{\pi}{2}, \frac{\pi}{2})$ holds initially, it holds at any time instance. Therefore, only $\mathbf{X} \cdot \mathbf{X_c} = 1$, i.e., $\widetilde{\theta} = 0$, is valid. As $\mathbf{X}$ and $\mathbf{X_c}$ are both unit vectors, $\mathbf{X} \cdot \mathbf{X_c} = 1$ is equivalent to $\mathbf{X} = \mathbf{X_c}$. Now by comparing $\mathbf{X}$ and $\mathbf{X_c}$ in Equations (141) and (144), respectively, we get $a\mathbf{X} + b\mathbf{Y} + c\mathbf{Z} = \widetilde{v}\mathbf{Y} + \widetilde{u}\mathbf{Z}$, therefore

$$a = 0, \tag{166}$$

$$b = \widetilde{v}, \tag{167}$$

$$c = \widetilde{u}. \tag{168}$$

Plugging $\mathbf{X} \cdot \mathbf{X_c} = 1$ into Equation (161), we can see $a = 0$, i.e., Equation (166) always holds. Plug Equation (167) into Equation (162), we obtain $d = 0$. Plug Equation (168) into Equation (163) we obtain $e = 0$. Plug $d = 0$ and $e = 0$ into l in Equation (154), we get $l = 0$, which suggests $\rho = 0$. In summary, $\dot{V}_1 = 0$ leads to $\widetilde{\theta} = 0$ and $\rho = 0$. By Lasalle's invariance principle, the state variables converge to the maximal invariant set where $\dot{V}_1 = 0$ is satisfied, and in this invariant set $\rho = 0$ and $\widetilde{\theta} = 0$. ∎

Note: given $\widetilde{\theta} \in (-\frac{\pi}{2}, \frac{\pi}{2})$ initially, the Lyapunov function V_1 is finite initially. We also showed that $\dot{V}_1 \le 0$ and $V_1 \ge 0$, therefore V_1 is finite at any time instance. It suggests that $\widetilde{\theta}$ will never cross $-\frac{\pi}{2}$ or $\frac{\pi}{2}$, otherwise $V_1 \to \infty$ will occur. In other words, if $\widetilde{\theta} \in (-\frac{\pi}{2}, \frac{\pi}{2})$ holds initially, it holds at any time instance.

Based on the above 3D curve tracking control law, the formation center control force $\mathbf{u}_c$ in Equation (118) can be calculated by plugging Equations (164) and (137) into (118), so

$$\mathbf{u}_c = M(\widetilde{v}_c \mathbf{Y_c} + \widetilde{u}_c \mathbf{Z_c}) \tag{169}$$

where $\widetilde{u}_c$ and $\widetilde{v}_c$ are defined in Equation (164).

4.4 Formation Controller

In this section, we develop the formation control to drive the vehicles to desired formation shape. We define the shape variables $\mathbf{s} = [s_1, s_2, ... s_{3N-6}]$ according to Section 4.2, and define their desired value as $\mathbf{s}_d = [s_{1,d}, s_{1,d}, ... s_{3N-6,d}]$, then attention function depends only on the shape variables, i.e.,

$$\mathscr{A}_i = \mathscr{A}_i(\mathbf{s}), \tag{170}$$

and the constraint in Equation (111) can be converted to an equivalent constraint on shape variables, i.e., $\mathbf{s} \geq \mathbf{s}_{min}^1$. To avoid collisions between the vehicles, the relative distances between vehicles have to be greater than a safety threshold, which can also be converted to constrains on the shape variables, so $\mathbf{s} \geq \mathbf{s}_{min}^2$. Define $\mathbf{s}_{min} = \max(\mathbf{s}_{min}^1, \mathbf{s}_{min}^2) \doteq [s_{1,min}, s_{2,min}, ..., s_{3N-6,min}]$, then

$$\mathbf{s} \geq \mathbf{s}_{min} \tag{171}$$

need to be satisfied all the time.

Now we introduce the following Lyapunov candidate function

$$V_2 = \frac{1}{2}\dot{\mathbf{s}}^T G\dot{\mathbf{s}} + \sum_{i=1}^{3N-6} h_i(s_i) \tag{172}$$

where h_i is a bowl-shape function satisfying the following conditions:

- $h_i(s_i) \geq 0$ and $h_i(s_i) = 0$ if and only if $s_i = s_{id}$

- $h_i(s_{i,min}) \rightarrow +\infty$ and $h_i(+\infty) \rightarrow +\infty$,

- $\frac{\mathrm{d}}{\mathrm{d}s_i}h_i(s_i) = 0$ if and only if $s_i = s_{id}$

From Equations (120) and (119), we get

$$\frac{d}{dt}(\frac{1}{2}\dot{\mathbf{s}}^T G\dot{\mathbf{s}}) = \dot{\mathbf{s}}^T(-\mathbf{A}^T\frac{d}{dt}(I\Upsilon) + \frac{1}{2}[\frac{\partial I}{\partial \mathbf{s}}]^* : (\Upsilon,\Upsilon) + ([\frac{\partial \mathbf{A})}{\partial \mathbf{s}}]^* - [\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]) : (\dot{\mathbf{s}},I\Upsilon) + \mathbf{u}_s)$$
$$= \dot{\mathbf{s}}^T(\mathbf{A}^T(\Upsilon - \mathbf{As}) \times I\Upsilon + \frac{1}{2}[\frac{\partial I}{\partial \mathbf{s}}]^* : (\Upsilon,\Upsilon) + ([\frac{\partial \mathbf{A})}{\partial \mathbf{s}}]^* - [\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]) : (\dot{\mathbf{s}},I\Upsilon) + \mathbf{u}_s - \mathbf{A}^T\mathbf{u}_g)$$

(173)

In addition, by defining $\mathbf{H} \doteq \begin{bmatrix} \frac{dh_1}{ds_1} \\ \frac{dh_2}{ds_2} \\ \vdots \\ \frac{dh_{3N-6}}{ds_{3N-6}} \end{bmatrix}$, we get

$$\frac{d}{dt}(\sum_{i=1}^{3N-6} h_i(s_i)) = \dot{\mathbf{s}}^T\mathbf{H}.$$

(174)

Using the simplified $\frac{d}{dt}(\frac{1}{2}\dot{\mathbf{s}}^T G\dot{\mathbf{s}})$ and $\frac{d}{dt}(\sum_{i=1}^{3N-6} h_i(s_i))$, we get the derivative of Lyapunov candidate function V_2 as

$$\dot{V}_2 = \dot{\mathbf{s}}^T(\mathbf{A}^T(\Upsilon - \mathbf{As}) \times I\Upsilon + \frac{1}{2}[\frac{\partial I}{\partial \mathbf{s}}]^* : (\Upsilon,\Upsilon) + ([\frac{\partial \mathbf{A})}{\partial \mathbf{s}}]^* - [\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]) : (\dot{\mathbf{s}},I\Upsilon) + \mathbf{u}_s - \mathbf{A}^T\mathbf{u}_g + \mathbf{H}) \quad (175)$$

We choose $\mathbf{u}_s$ and $\mathbf{u}_g$ satisfying

$$\mathbf{u}_s - \mathbf{A}^T\mathbf{u}_g = -(\mathbf{A}^T(\Upsilon - \mathbf{As}) \times I\Upsilon + \frac{1}{2}[\frac{\partial I}{\partial \mathbf{s}}]^* : (\Upsilon,\Upsilon) + ([\frac{\partial \mathbf{A})}{\partial \mathbf{s}}]^* - [\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]) : (\dot{\mathbf{s}},I\Upsilon) + \mathbf{H} + k_4\dot{\mathbf{s}}) \quad (176)$$

where $k_4 > 0$, then

$$\dot{V}_2 = -k_4\dot{\mathbf{s}}^T\dot{\mathbf{s}} \leq 0.$$

(177)

From Equation (176) and the constraint in Equation (134), we get the formation shape control $\mathbf{u}_s$ and the orientation control $\mathbf{u}_g$ as following:

$$\mathbf{u}_s = \frac{N-1}{N}\mathbf{A}^T(\Gamma_{22} + \Gamma_{23}\mathbf{A}^T)^{-1}\chi(\mathbf{s},g)$$
$$+ (\mathbf{A}^T(\Gamma_{22} + \Gamma_{23}\mathbf{A}^T)^{-1}\Gamma_{23} - \mathbf{e})[\mathbf{A}^T(\Upsilon - \mathbf{As}) \times I\Upsilon$$
$$+ \frac{1}{2}[\frac{\partial I}{\partial \mathbf{s}}]^* : (\Upsilon,\Upsilon) + ([\frac{\partial \mathbf{A})}{\partial \mathbf{s}}]^* - [\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]) : (\dot{\mathbf{s}},I\Upsilon) + \mathbf{H} + k_4\dot{\mathbf{s}}]$$

(178)

$$\mathbf{u}_g = (\Gamma_{22} + \Gamma_{23}\mathbf{A}^T)^{-1}[\frac{N-1}{N}\chi(\mathbf{s},g) - \Gamma_{23}(\mathbf{A}^T(\Upsilon - \mathbf{As}) \times I\Upsilon$$
$$+ \frac{1}{2}[\frac{\partial I}{\partial \mathbf{s}}]^* : (\Upsilon,\Upsilon) + ([\frac{\partial \mathbf{A})}{\partial \mathbf{s}}]^* - [\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]) : (\dot{\mathbf{s}},I\Upsilon) + \mathbf{H} + k_4\dot{\mathbf{s}})]$$

(179)

Theorem 4 *Under control law (178) and (179), formation shape converges to desired ones, i.e.,*

$s \rightarrow s_d$

Proof Equations (178) and (179) are equivalent to Equations (176) and (134). Plug Equation (176) into Equations (119) and (120), we get the closed-loop dynamics of the formation system:

$$\frac{\mathrm{d}}{\mathrm{d}t}(G\dot{\mathbf{s}}) + \mathbf{H} + k_4\dot{\mathbf{s}} = \frac{1}{2}[\frac{\partial G}{\partial \mathbf{s}}]^* : (\dot{\mathbf{s}}, \dot{\mathbf{s}}) \tag{180}$$

By Lasalle's invariance principle, the state variables **s** converge to the maximal invariant set where $\dot{V_2} = 0$ is satisfied. In this invariant set, we have $\dot{\mathbf{s}} = 0$ according to Equation (177), and then according to the closed-loop dynamics (180), we have $\mathbf{H} = 0$, i.e., $\frac{dh_i}{ds_i} = 0$, for $i = 1, 2, ...3N - 6$. Therefore, $s_i = s_{id}$ for $i = 1, 2, ...3N - 6$, i.e., $\mathbf{s} \rightarrow \mathbf{s}_d$. ∎

Note: given $\mathbf{s} \geq \mathbf{s}_{min}$ initially, the Lyapunov function V_2 is finite initially. We also showed that $\dot{V_2} \leq 0$ and $V_2 \geq 0$, therefore V_2 is finite at any time instance. It means that **s** will never reach $\mathbf{s}_{min}$, otherwise $V_2 \rightarrow \infty$ will occur (because $h_i(s_{i,min}) \rightarrow +\infty$). In other words, if $\mathbf{s} \geq \mathbf{s}_{min}$ holds initially, it holds at any time instance. Therefore, collision between vehicles and scare of the human operator is avoided.

Now we convert control law (178) and (179) to get the **gauge invariant form of the formation shape control**. Define

$$\mathbf{U}_g = \mathbf{u}_g, \tag{181}$$

$$\mathbf{U}_s = \mathbf{u}_s - \mathbf{A}^T\mathbf{u}_g. \tag{182}$$

According to Lemma 3.2 in [31], $\mathbf{U}_g$ and $\mathbf{U}_s$ is gauge invariant and independent of the selection of coordinate system. From Equations (179) and (176), we get the following **gauge invariant form of the formation shape control:**

$$\mathbf{U}_g = (\Gamma_{22} + \Gamma_{23}\mathbf{A}^T)^{-1}[\frac{N-1}{N}\chi(\mathbf{s}, g) - \Gamma_{23}(\mathbf{A}^T(\Upsilon - \mathbf{As}) \times \Pi\Upsilon \\ + \frac{1}{2}[\frac{\partial I}{\partial \mathbf{s}}]^* : (\Upsilon, \Upsilon) + ([\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]^* - [\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]) : (\dot{\mathbf{s}}, \Pi\Upsilon) + \mathbf{H} + k_4\dot{\mathbf{s}})] \tag{183}$$

$$\mathbf{U}_s = -(\mathbf{A}^T(\Upsilon - \mathbf{As}) \times \Pi\Upsilon + \frac{1}{2}[\frac{\partial I}{\partial \mathbf{s}}]^* : (\Upsilon, \Upsilon) + ([\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]^* - [\frac{\partial \mathbf{A}}{\partial \mathbf{s}}]) : (\dot{\mathbf{s}}, \Pi\Upsilon) + \mathbf{H} + k_4\dot{\mathbf{s}} \tag{184}$$

This gauge invariant formation shape control is the same under any coordinate system, and allows each vehicle to use its own gauge convention. Therefore, the HUV and AUVs do not have to adopt

the same coordinate system. Each AUV in the formation can calculate the actual force applied on it, i.e., $\mathbf{f}_i$, using its own choice of coordinate system and Jacobi vectors. In its own coordinate system, each AUV can estimate $\mathbf{f}_h$ of HUV according to Equations (106), (107), (108) and (112), and $\mathbf{u}_c$ according to Equations (105) and (169), using only its measurements of each vehicles's location (which completely decide the distance between HUV and AUVs), velocity (will be converted to natural Frenet representation for calculation of $\mathbf{u}_c$), and acceleration. In addition, shape variable $\mathbf{s}$ which can be completely determined by vehicles locations can also be calculated based on the same set of measurements. Therefore, our cooperative control law, including the curve tracking controller and formation shape controller, is based only on measurements, and no communication between vehicles is needed.

4.5   Simulation Results for One Example

In this section, we give an example of a human-AUV system comprised of one HUV and two AUVs and the corresponding numerical simulation results. For simplicity, we assume $m_1 = m_2 = m_h = 1$. The three vehicles are desired to form a isosceles right-angled triangle, with length of cathetus to be 0.2. The formation center is desired to track a helix curve $[\frac{\sin t}{\sqrt{2}}, \frac{\cos t}{\sqrt{2}}, \frac{t}{\sqrt{2}}]$. To describe the triangle, we choose the shape variable as $\mathbf{s} = [s_1, s_2, s_3]$, where $s_1 = \|\mathbf{r}_1 - \mathbf{r}_h\|$, $s_2 = \|\mathbf{r}_1 - \mathbf{r}_h\|$, and s_3 is the angle between $\mathbf{r}_1 - \mathbf{r}_h$ and $\mathbf{r}_2 - \mathbf{r}_h$, so the desired shape variables are $s_{1,d} = 0.2$, $s_{1,d} = 0.2$, and $s_{1,d} = \frac{\pi}{2}$. Using the shape variable, the attention function simplifies to

$$\mathscr{A}_i = \frac{\mathscr{A}_{i0} s_{id}}{s_i} \tag{185}$$

for $i = 1, 2$. The body frame is built as Figure 23.

For $N = 3$, the Jacobi transform becomes

$$\mathbf{r}_1 = \mathbf{r}_c - \frac{\sqrt{2}}{2}\rho_{f1} - \frac{1}{\sqrt{6}}\rho_{f2}$$

$$\mathbf{r}_2 = \mathbf{r}_c + \frac{\sqrt{2}}{2}\rho_{f1} - \frac{1}{\sqrt{6}}\rho_{f2} \tag{186}$$

$$\mathbf{r}_h = \mathbf{r}_c - \frac{\sqrt{2}}{\sqrt{3}}\rho_{f2}$$
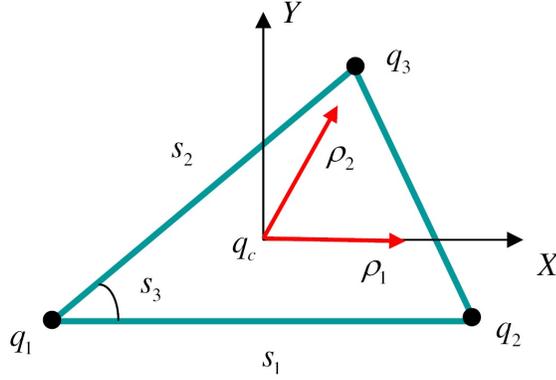
Figure 23: Body Frame

therefore,

$$\Phi = \begin{bmatrix} \frac{1}{3}\mathbf{e}_3 & -\sqrt{\mu_1}\mathbf{e}_3 & -\frac{1}{2}\sqrt{\mu_2}\mathbf{e}_3 \\ \frac{1}{3}\mathbf{e}_3 & \sqrt{\mu_1}\mathbf{e}_3 & -\frac{1}{2}\sqrt{\mu_2}\mathbf{e}_3 \\ \frac{1}{3}\mathbf{e}_3 & 0 & \sqrt{\mu_2}\mathbf{e}_3 \end{bmatrix}, \tag{187}$$

where $\mu_1 = \frac{1}{2}$ and $\mu_2 = \frac{2}{3}$. In addition, we get

$$\mathbf{r}_1 - \mathbf{r}_h = -\frac{\sqrt{2}}{2}\rho_{f1} + (\frac{\sqrt{2}}{\sqrt{3}} - \frac{1}{\sqrt{6}})\rho_{f2}$$
$$\mathbf{r}_2 - \mathbf{r}_h = \frac{\sqrt{2}}{2}\rho_{f1} + (\frac{\sqrt{2}}{\sqrt{3}} - \frac{1}{\sqrt{6}})\rho_{f2}. \tag{188}$$

In the body frame, the vector $(\mathbf{r}_2 - \mathbf{r}_1)$ is $(s_1, 0, 0)$ and the vector $(\mathbf{r}_h - \frac{\mathbf{r}_1 + \mathbf{r}_2}{2})$ is $(s_2 \cos(s_3), s_2 \sin(s_3), 0)$, thus the Jacobi vectors in the body frame are

$$\rho_1 = \sqrt{\mu_1}(s_1, 0, 0)^T$$
$$\rho_2 = \sqrt{\mu_2}(s_2 \cos(s_3) - \mu_1 s_1, s_2 \sin(s_3), 0)^T \tag{189}$$

Now we can calculate the elements for matrix I in Equation (121) and get

$$I_{11} = \mu_2 s_2^2 \sin^2(s_3)$$

$$I_{12} = I_{21} = -\mu_2 s_2 \sin(s_3)(s_2 \cos(s_3) - \mu_1 s_1)$$

$$I_{22} = \mu_1 s_1^2 + \mu_2(s_2 \cos(s_3) - \mu_1 s_1)^2 \tag{190}$$

$$I_{33} = \mu_1 s_1^2 + \mu_2((s_2 \cos(s_3) - \mu_1 s_1)^2 + s_2^2 \sin^2(s_3))$$

$$I_{23} = I_{32} = I_{13} = I_{31} = 0$$

66

and then consequently

$$\frac{\rho_1}{s_1} = \sqrt{\mu_1}(1,0,0)^T$$

$$\frac{\rho_1}{s_2} = (0,0,0)^T$$

$$\frac{\rho_1}{s_3} = (0,0,0)^T$$

$$\frac{\rho_2}{s_1} = \sqrt{\mu_2}(-\mu_1,0,0)^T \tag{191}$$

$$\frac{\rho_2}{s_2} = \sqrt{\mu_2}(\cos(s_3),\sin(s_3),0)^T$$

$$\frac{\rho_2}{s_3} = \sqrt{\mu_2}(-s_2\sin(s_3),s_2\cos(s_3),0)^T$$

which lead to

$$A_1 = \frac{\mu_2}{I_{33}}(0,0,\mu_1 s_2 \sin(s_3))^T$$

$$A_2 = \frac{\mu_2}{I_{33}}(0,0,-\mu_1 s_1 \sin(s_3))^T \tag{192}$$

$$A_3 = \frac{\mu_2}{I_{33}}(0,0,s_2^2 - \mu_1 s_1 s_2 \cos(s_3))^T$$

Now $\mathbf{A}$, G and Υ can be decided by Equation (121), Γ can be decided by Equations (125) and (127). In addition, Equations (188), (117) and (189) lead to

$$\mathbf{r}_1 - \mathbf{r}_h = -\frac{\sqrt{2}}{2}g^{-1}\sqrt{\mu_1}[s_1,0,0]^T + (\frac{\sqrt{2}}{\sqrt{3}} - \frac{1}{\sqrt{6}})g^{-1}\sqrt{\mu_2}[s_2\cos(s_3) - \mu_1 s_1, s_2\sin(s_3),0]^T$$

$$\mathbf{r}_2 - \mathbf{r}_h = \frac{\sqrt{2}}{2}g^{-1}\sqrt{\mu_1}(s_1,0,0)^T + (\frac{\sqrt{2}}{\sqrt{3}} - \frac{1}{\sqrt{6}})g^{-1}\sqrt{\mu_2}[s_2\cos(s_3) - \mu_1 s_1, s_2\sin(s_3),0]^T. \tag{193}$$

Comparing Equations (130) and (112), we get $\chi(\mathbf{s},g) = \sum_{i=1}^{2} a_i$ where a_i, defined in Equation (110), is a function of only $\mathbf{s}$ and g, and is calculated from Equations (193), (185), and (110). Now the formation shape control $\mathbf{U}_s$ and $\mathbf{U}_g$ can be calculated according to Equations (184) and (183), and then converted to $\mathbf{u}_g$ and $\mathbf{u}_s$ according to Equations (181) and (182).

We convert the desired helix curve to Frenet-Serret frame representation as $\mathbf{T} = [\frac{\sqrt{2}\cos t}{2}; -\frac{\sqrt{2}\sin t}{2}; \frac{\sqrt{2}}{2}]^T$, $\mathbf{N} = [-\sin t; -\cos t; 0]^T$, $\mathbf{B} = [\frac{\sqrt{2}\cos t}{2}; -\frac{\sqrt{2}\sin t}{2}; -\frac{\sqrt{2}}{2}]^T$, $\kappa = \frac{\sqrt{2}}{2}$, $\tau = -\frac{\sqrt{2}}{2}$, then one natural Frenet frame representation can be obtained according to Equations (290) and (291), by letting $\zeta = 0$. Now we can calculate formation center control $\mathbf{u}_c$ according to Equations (169) and (164). Using $\mathbf{u}_c$, $\mathbf{u}_g$, and $\mathbf{u}_s$, we can calculate the forces applied on each AUV, i.e., $\mathbf{f}_1$ and $\mathbf{f}_2$, according to Equation (128).

To demonstrate the effectiveness of our method, we carried out numerical simulation for the above example. Simulation results are shown in Figures (24), (25), (26), and (27). Figure (24)

67
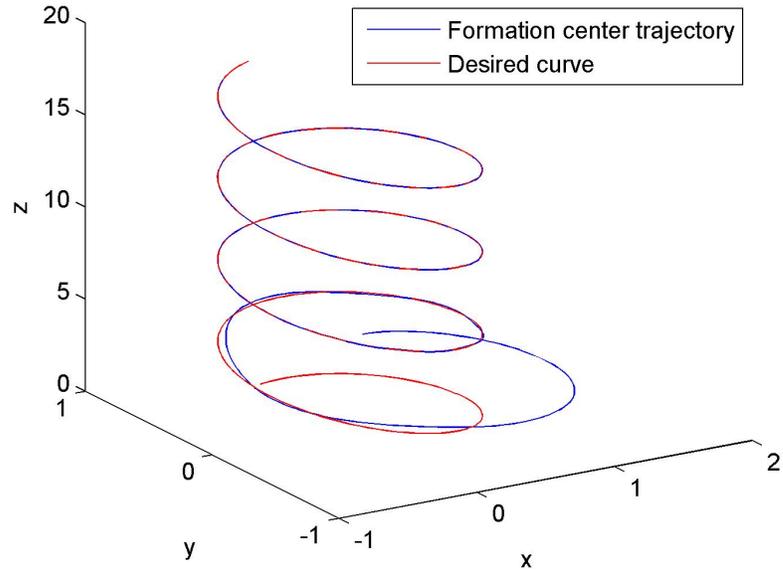
Figure 24: Trajectory of formation center
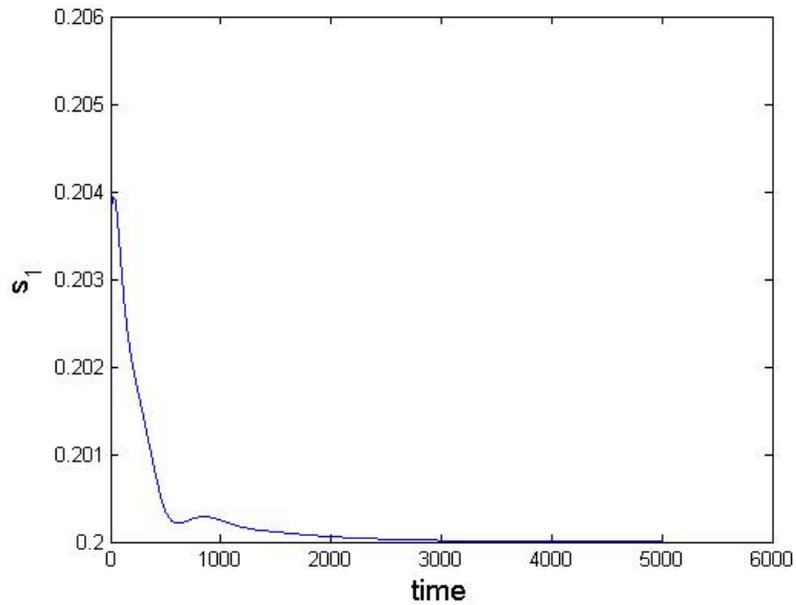
Figure 25: Shape variable s_1

shows the trajectory of the formation center, which we can see converges to the desired curve. Figures (25), (26), and (27) show the three shape variables, which we can see all converge to their desired values.
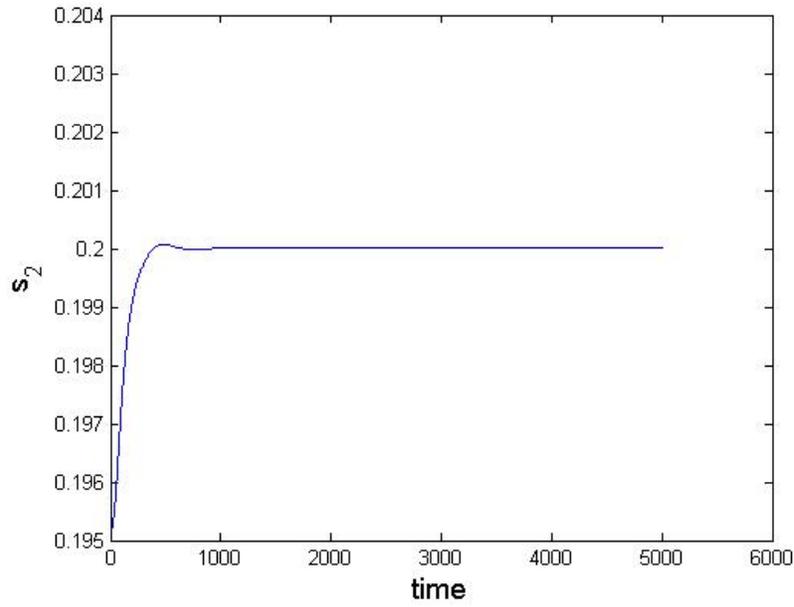
Figure 26: Shape variable s_2
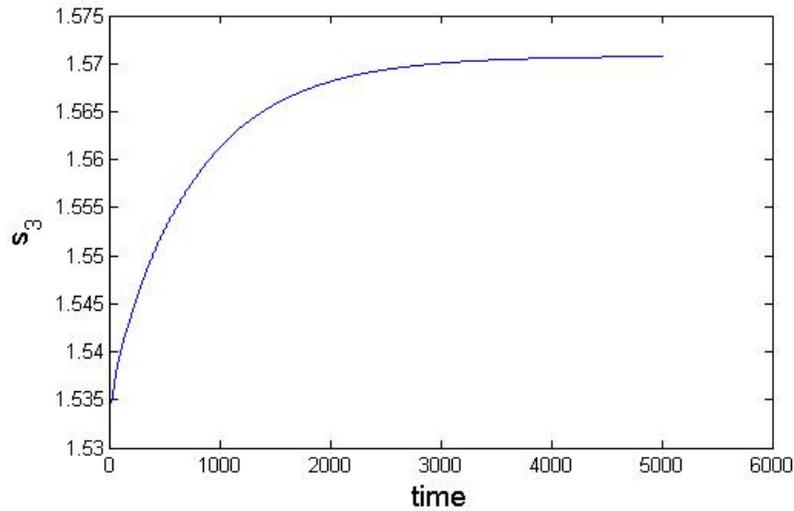
Figure 27: Shape variable s_3

4.6 Conclusion

In this chapter we developed cooperative control laws for a human-AUV system. Using a geometric approach, we explicitly decoupled the collective dynamics of all vehicles into orientation, formation shape, and formation center. We designed a 3D curve tracking controller for the formation center, so that the formation center can tracks any curve in 3D space. We also developed formation

shape controller which enables the system converge to a desired formation shape while avoiding scaring the human operator and collisions between vehicles. The proposed controller does not need communication between vehicles and is gauge invariant, which enables each vehicle to use its own coordinate system.

CHAPTER V

COLLABORATIVE AUTONOMOUS SURVEYS IN MARINE ENVIRONMENTS AFFECTED BY OIL SPILLS

Marine surveys are crucial for assessing risks of maritime disasters like the Deepwater Horizon oil spill in 2010. After the oil spill occurred, research aimed at creating technology and tools to effectively handle such catastrophes in future.

Autonomous surveys are especially attractive in situations where the marine environment is less than ideal for human-based methods. During an autonomous survey, marine robots must be able to move along a desired path in order to gather sensor data along that path, as a result, path following for marine vehicles is very important, and, therefore, has been widely studied [59–65]. Although theoretical work on path following is advanced [56, 94], yet the performance of only few theoretical results has been evaluated and reported in field tests [66]. The path following controllers in our work are based on the Frenet-Serret framework [57] and the robustness of the control laws has been theoretically justified [56]. Satisfactory performance of the controllers has been seen on mobile robots and ocean gliders [67, 68]. Here we are motivated to evaluate the controllers on marine robots for oil spill surveys, where the robots, the environment, and the technical challenges differ significantly from previous mobile robots and ocean gliders experiments.

The main contribution of this work is to demonstrate a simple and effective method to carry out marine surveys on a large area and to reconcile data from various sources to produce meaningful representations. We use a simple unicycle model for our autonomous vehicles. Our controllers enable the vehicle to track lines and curves reliably in the presence of natural disturbances, including wind, water currents, and engineering limitations like sensor inaccuracy, localization errors, and communication delays. Using the data collected by the vehicles during autonomous missions, we demonstrate the effectiveness of a mapping algorithm for generating bathymetric maps and oil concentration maps for the region of interest, where data are not necessarily available at all points. Therefore, our methods for performing autonomous marine surveys are simple and can effectively

reconcile data obtained over multiple different experimental runs from a fleet of autonomous marine vehicles.

To demonstrate the effectiveness of our survey methods, we performed a twenty-one-day survey in July, 2011, for a coastal lagoon in Grand Isle, Louisiana, where heavy pollution had been reported during the DeepWater Horizon oil spill, and large scale cleaning efforts have been performed after the spill was contained. Using a fleet of heterogeneous marine vehicles, we collected a large amount of data to evaluate the concentration level of reminiscent oil one year after the cleanup efforts. This chapter presents our analysis of the survey data and experimental results on parameter identification and path following control for two marine robots we used in the survey. The organization of this chapter is as follows. Section 5.1 describes the hardware and software systems for all the vehicles we used in the survey and explains the dynamic model to describe vehicle motions. The control laws and mapping algorithms are presented in section 5.2, followed by experimental results and data analysis in Section 5.3. Section 5.4 gives the conclusion.

5.1   Marine Robots

In our survey efforts, four vehicles were employed. They were a student-built autonomous surface vehicle (ASV), Victoria, a student-built remote operated vehicle (ROV), β, and autonomous underwater vehicles (AUV's), the Fetch 1, and the EcoMapper, shown in Figure 28. Victoria and the Fetch 1 played a main role in the survey, while the EcoMapper and ROV-β were in an auxiliary role. This section describes the hardware and software for each vehicle, and the mathematical models for vehicle motions.

5.1.1   The ASV-Victoria

5.1.1.1   Hardware

ASV-Victoria (the left one in the second row of Figure 28) is developed and built by a student robotics team supervised by Dr. Fumin Zhang. It weighs $50kg$, and is approximately $1m$ long and $0.75m$ wide. The the trim is $0.5m$ and the overall height is $0.75m$. Victoria has a twin-hull-catamaran design, with the hulls composed of multiple layers of fiberglass sheets. The twin-hull-catamaran design allows for a smooth ride since the turbulence in the center of the boat is reduced. The hull spacing is optimized to produces high stability and high load carrying capability. Victoria uses a

Figure 28: Marine vehicles used in the survey. From top to bottom: AUV-Fetch 1, ASV-Victoria (left), ROV-β (right), and the EcoMapper.

specific layout for electronics, propulsion and power systems inside the hulls to minimize pitching due to sudden changes in acceleration.

The electronic equipment onboard Victoria can be classified into computational units, sensors, actuators and communication systems. Figure 29 shows a high level view of Victoria's electrical systems. Victoria houses two separate computational units, one supporting navigation, vision systems and the other supporting lower level thruster control and data acquisition from all other sensors

onboard. Each computational unit is a Compact RIO (cRIO) produced by National Instruments. The cRIO's are chosen as they combine an embedded real-time processor, a Field Programmable Gate Array (FPGA), and I/O modules. This provides reliability and speed of operation, and makes it easy to swap out onboard sensors depending on survey requirements. Two thrusters from CrustCrawler Robotics form the main actuation units and are mounted on the hulls. They are capable of producing up to 60 *lb* of thrust resulting a maximum linear speed of 2 *m/s* of Victoria. A Microstrain 3DM-GX1 inertial motion unit (IMU), an ethernet camera, a Cyclops-7 oil sensor made by Turner Designs, and a Garmin 16x GPS receiver are the main sensors onboard Victoria during the surveys. A long range wireless link forms the backbone of our communication system and enables remote operation from shore. The wireless communication setup includes a Ubiquiti Rocket-M5 base station on shore, a Ubiquiti Bullet-M5 access point, and an Ethernet router on board Victoria. The vehicle can work in both autonomous mode and remote control mode. In remote control mode, the vehicle can be remotely controlled within a range of approximately 500*m*. We use an Xbox controller to send commands to Victoria during remote operation. An intuitive joystick-based controller coupled with video from Victoria's onboard cameras contribute greatly to the ease of operation. In autonomous mode, curve tracking controller enables Victoria to track specified path autonomously without human intervention.

5.1.1.2 Software

National Instruments (NI) LabVIEW is used onboard ASV-Victoria. A high level schematic of Victoria's software architecture, shown in Figure 30, is composed of three main virtual instruments (VI's): Main PC, Co-operative control, Main RIO, and Main FPGA. The VI Main PC runs on the control laptop and retrieves data from VI's running on the cRIO's on board Victoria, and displays it on the on-shore laptop computer. It also receives commands from the Xbox controller and sends the commands to Victoria, ensuring easy remote operation. The co-operative control VI runs on the main PC and communicates over a network with other autonomous vehicles. The algorithms sub-VI's under the co-operative control VI receive position and orientation data from other autonomous vehicles and send appropriate control commands that enable these vehicles to track desired curves. This makes it very easy to enlarge the size of a survey fleet, as other vehicles can be added to the
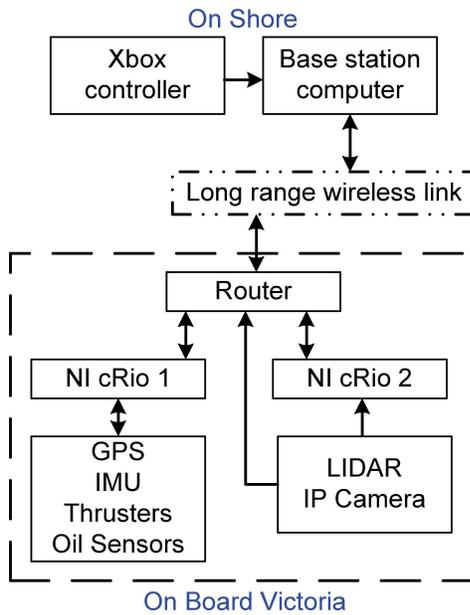
Figure 29: A high level schematic of Victoria's electrical systems
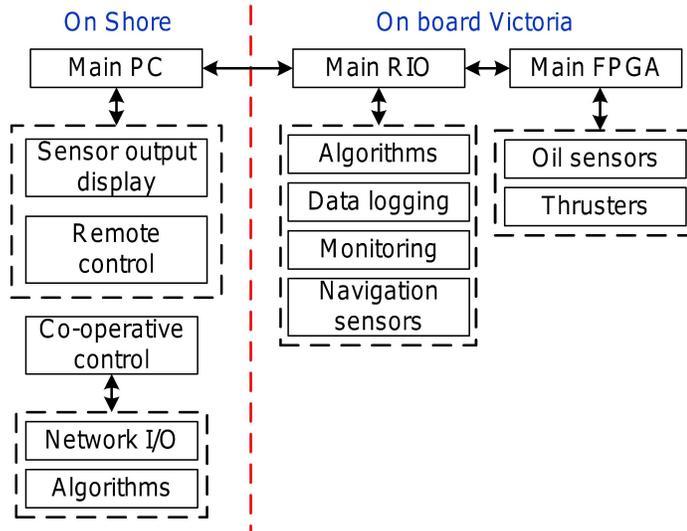
Figure 30: A high level schematic of Victoria's software architecture

fleet by simply adding a new co-operative control VI to the main computer program (one for each added vehicle).

The VI's Main RIO and Main FPGA run on cRIO's onboard Victoria. The VI Main RIO has many sub-VI's. Based on their functions, they can be classified to sub-VI's dealing with algorithms, data logging, monitoring, and sensing. As the name indicates, the sub-VI's dealing with algorithms

perform the calculations that allow vehicle operates in the different states like GPS waypoint navigation and autonomous curve tracking. The data logging VI's record data from the sensing devices into Victoria's onboard memory. The monitoring VI's help monitor critical parameters like CPU temperature, CPU load and execution speed, in order to maintain optimal operating conditions for the computer system. The navigation sensor VI's query navigation equipment like GPS and LIDAR, and provide the data to any other VI, like the algorithm and logging VI's, requiring such data. Simpler lower level sensing and control tasks are handled by the Main FPGA VI running on an FPGA module in a cRIO. This VI handles data acquisition from the oil sensors, receives control commands from the algorithms VI's, and converts them into appropriate commands for the thrusters. The oil sensors and thrusters are dedicated resources which are used on every mission, hence they are architecturally separated to ensure that system-wide code changes do not result in stray bugs that could affect these vital systems. This improves the system reliability.

5.1.2 The Fetch 1

5.1.2.1 Hardware

The Fetch 1 (the first row of Figure 28), developed by Professor Mark Patterson of the College of William and Mary, is an autonomous vehicle that can be used either on surface or underwater, with maximum diving depth $500ft$. It served as an ASV for some of the experiments in our marine survey. The Fetch 1 is aluminum hulled, $220lb$, $6.5ft$ long, driven by a single propeller, and steered by two pairs of single-degree-of-freedom control surfaces. Fetch 1 is outfitted with Wi-Fi as well as a FreeWave RF serial modem that maintains constant contact with a shore station as long as the vehicle is on the surface. The Fetch 1 uses an assortment of sensors including GPS, water temperature and salinity sensors, as well as a crude oil sensor.

5.1.2.2 Software

Fetch 1's main flight computer also runs LabVIEW. For all experiments mentioned in this chapter, Fetch 1 was operated in teleoperation mode where Victoria's shore-side control computer sent commands to Fetch. The communication setup is illustrated in Figure 31. Feedback control algorithms operated at a frequency of $0.5Hz$, with data exchange every two seconds. This is a limitation imposed by the operating speed of the state machine loop in the flight computer in Fetch 1. To deal
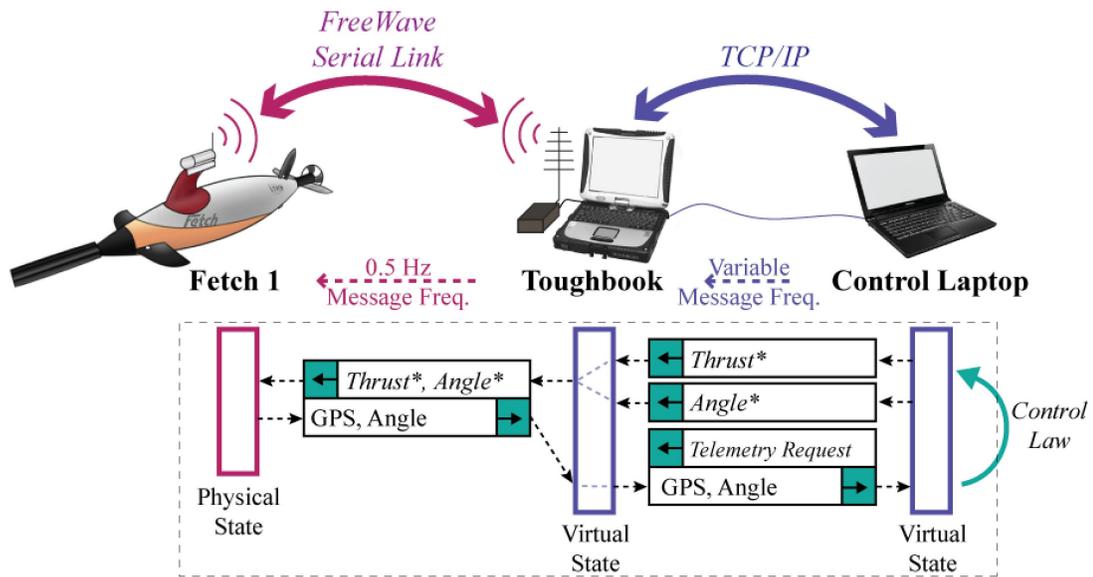
Figure 31: Communicating with the Fetch

with the imposed delay, we use buffers, known as virtual states, on the Toughbook and the control laptop. The GPS positions and control-surface angles are periodically received from the Fetch and stored in a virtual state on the Toughbook. All commands from the control laptop are also stored in this virtual state, so that they can be forwarded to the main flight computer in Fetch 1 when a transmission is relayed to the Fetch 1. On the control laptop, a virtual state is maintained and updated every time the Toughbook is polled for data and every time a new command is generated by the control law. Therefore, the data based on which the control law performs its calculations is at most two-seconds old.

5.1.3 The ROV - β

5.1.3.1 Hardware

The ROV-β, shown on the right in the second row of Figure 28, is an remotely-operated underwater vehicle built bythe student robotics team supervised by Dr. Fumin Zhang. It is $36.45in$ in length, $22.5in$ in width, $18.25in$ in height, weighs $125lb$, and is capable of diving to a depth of $330ft$. The sensors onboard can sample oil, measures depth, and acquire visual data. A system for collecting water samples and a pneumatically powered manipulator are also housed onboard. The major

components of ROV-β are the pressure vessel, frame, propulsion system, buoyancy system, the manipulator, the power and control systems. The body is built of aluminum and carbon steel. Black Rustoleum paint is used to protect the carbon steel from corrosion. The cylindrical pressure vessel has a volume of 160 in^3 and is rated to a depth of 500ft. The pressure vessel has two removable end caps, one on each end of the cylinder. Each cap houses SEACON connectors for through-hull electrical connections. The cap on the front end has a transparent acrylic dome which forms the viewport for the primary camera. The carbon steel frame is welded directly to the pressure vessel. The frame's unique design protects the thrusters from collisions and provides a surface to mount external subsystems like actuators and sampling tubes. The propulsion system includes four oil-compensated thrusters made by CrustCrawler Robotics, each one is capable of producing a thrust of 25lb. The buoyancy system is made of syntactic foam and mounted atop a rack on the frame to offset the negative buoyancy of the ROV and place the center of buoyancy above the center of gravity, making the ROV more stable. ROV-β has a pneumatic manipulator to grab articles of interest while underwater. Power for the ROV is supplied from the surface by two deep-cycle lead acid batteries. The control system comprises of a cRIO, a router, and an Xbox controller. The cRIO is primarily used to send thruster commands and record sensor data. An onboard network switch provides connectivity between the ROV and the shore via a single Cat5e cable. The onshore wireless router allows the pilot to operate away from the launch point. ROV-β improves the effectiveness of a survey team because of its capability to go underwater at points of interest and collect samples.

5.1.3.2 *Software*

Similar to ASV-Victoria, NI-LabVIEW is used onboard ROV-β. The software for ROV-β is a simplified version of the software for ASV-Victoria, as ROV-β is remotely controlled and does not require autonomous control and navigation algorithms. We skip the explicit discussion of software of ROV-β, as it is similar with the one of ASV-Victoria.

5.1.4 The EcoMapper

5.1.4.1 *Hardware*

The EcoMapper (the fourth row of Figure 28) made by YSI Inc. is an AUV for environment mapping. It weighs 45lb and is 152cm in length from bow to the stern. The diameter of the hull is

14.7cm. It features four independent control planes and a two-bladed propeller. The EcoMapper is rated for a depth of 220ft. The maximum speed is designed at 4 knots. Onboard it uses a computer with a processor from the x86 genre and magnetic storage for saving survey data. It communicates with a shore station via a 802.11g Wi-Fi link. The EcoMapper's navigation system is comprised of a GPS for surface operations and a Doppler Velocity Log (DVL) for underwater operations. The nose cone houses the sensors used for surveys. The EcoMapper is equipped with conductivity and temperature sensors, a three-axis digital compass, depth sensor (measures depth from surface), and the depth-sounding sonar (measures height from bottom). It is powered by rechargeable Li-ion batteries rated at 600Wh, which lasts for about 8 hours when the speed of the EcoMapper is around 2.5 knots. The EcoMapper makes it very easy to generate bathymetry maps, which facilitates autonomous-survey-mission design for other marine vehicles.

5.1.4.2 Software

The EcoMapper is operated through Underwater Vehicle Console (UVC) under Windows XP, accessed via "Windows Remote desktop" on a user's computer over a Wi-fi connection. The EcoMapper can be operated in both manual and autonomous modes. When the EcoMapper is on the surface and within Wi-fi range, it can be driven manually. In manual mode, EcoMapper status and sensor readings are displayed on the UVC screen but not recorded by the EcoMapper. In autonomous mode, the EcoMapper follows a predefined course either on or below surface, and does not require assistance from the human user. During missions, the EcoMapper acquires pertinent information useing sensors mounted in the nose cone and saves them to log files.

5.1.5 Mathematical Model for the Vehicles

To describe the motion of our vehicle, like Victoria and Fetch 1, we view them as point particles and use the following unicycle model to describe their kinematics:

$$\dot{x} = v\cos\theta, \tag{194}$$

$$\dot{y} = v\sin\theta, \tag{195}$$

$$\dot{\theta} = w, \tag{196}$$

where $[x, y]$, v, and ω represents the position, linear speed, and angular speed of the robot, respectively. For Victoria, which has two thrusters, the linear and angular speed, i.e., v and ω in Equations (194)-(196), can be written in terms of the left-thruster velocity and the right-thruster velocity:

$$v = \frac{v_l + v_r}{2}, \tag{197}$$

$$\omega = \frac{v_r - v_l}{2l}, \tag{198}$$

$$v_l = K_1 n_l, \; v_r = K_2 n_r, \tag{199}$$

where $2l$ is the distance between the two thrusters. n_l and n_r are the duty cycles of the signals sent to the left and right thrusters, respectively. K_1 and K_2 are constant coefficient that will be estimated in experiments. Substituting Equation (199) into Equations (194)-(196), we obtain

$$\dot{x} = \frac{K_1 n_r + K_2 n_l}{2} \cos\theta, \tag{200}$$

$$\dot{y} = \frac{K_1 n_r + K_2 n_l}{2} \sin\theta, \tag{201}$$

$$\dot{\theta} = \frac{K_1 n_r - K_2 n_l}{2l}. \tag{202}$$

Note that although the above models, which our control algorithms rely on, is simple and make the vehicle-driving code simple and easy, they do not consider the lateral drift or any other effects. However, we will show that our controllers are robust enough to produce satisfactory results.

5.2 Algorithms

In this section, we introduce the algorithm to identify the parameters of the vehicle model (200)-(202), based on which we then develop the curve tracking control law. We also present the algorithm to reconcile survey data and make maps.

5.2.1 Parameter Identification

From Equations (197)-(199), we obtain

$$K_1 = \frac{v + l\omega}{n_r}, \tag{203}$$

$$K_2 = \frac{v - l\omega}{n_l}. \tag{204}$$

According to Equations (203) and (204), we perform open loop tests to identify K_1 and K_2. In each open loop test, thruster commands n_r and n_l stay constant, and v and ω are estimated from GPS

data. Then Equations (203) and (204) give the value of K_1 and K_2. This procedure is simple and easy to implement in experiment in our survey methods, while enable us to generate a model close to reality.

5.2.2 Curve Tracking Control Law

The model in Equations (194)-(196) is easy to implemented in order to drive the vehicle, but not straightforward to use in curve tracking control law derivation. Therefore, we convert it to the following equivalent model in Frenet-Serret frame [57]. Define $\mathbf{r} = [x, y]^T$, $\mathbf{x} = [\cos\theta, \sin\theta]^T$, and $\mathbf{y} = [-\sin\theta, \cos\theta]^T$, then $\mathbf{x}$ and $\mathbf{y}$ are unit orthogonal vectors. Now the unicycle model in Equations (194)-(196) can be rewrite as

$$\dot{\mathbf{r}} = v\mathbf{x} \tag{205}$$

$$\dot{\mathbf{x}} = vu\mathbf{y} \tag{206}$$

$$\dot{\mathbf{y}} = -vu\mathbf{x} \tag{207}$$

where $u \doteq \omega/v$. Noted that Equations (205), (206), and (207) can describe not only motions of a vehicle but also planar curves. Given a smooth curve in the plane, we can imagine a virtual robot moves along this curve with speed v, then $\mathbf{x}$ becomes the tangent vector that it is always tangent to the curve, and $\mathbf{y}$ is the normal vector that it is perpendicular to $\mathbf{x}$. Usually when a curve is concerned, the symbol k is used in place of u and the speed $v = 1$. Now we formulate the curve tracking problem using two Frenet-Serret equations, with $\mathbf{r}_2$ denoting the position vector of the robot and $\mathbf{r}_1$ denoting the position of the the closest point on a curve with respect to the robot, so we get

$$\dot{\mathbf{r}}_1 = v_1\mathbf{x}_1 \tag{208}$$

$$\dot{\mathbf{x}}_1 = \mathbf{y}_1 v_1 k \tag{209}$$

$$\dot{\mathbf{y}}_1 = -\mathbf{x}_1 v_1 k \tag{210}$$

$$\dot{\mathbf{r}}_2 = \mathbf{x}_2 \tag{211}$$

$$\dot{\mathbf{x}}_2 = \mathbf{y}_2 u_2 \tag{212}$$

$$\dot{\mathbf{y}}_2 = -\mathbf{x}_2 u_2 \tag{213}$$

81
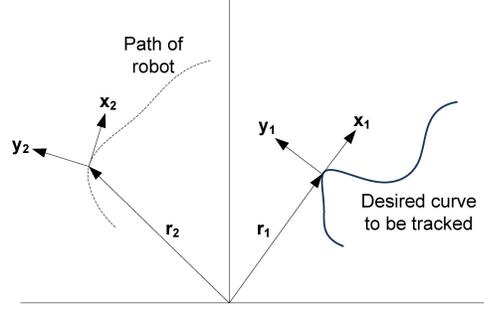
Figure 32: Curve tracking using two Frenet-Serret frames

where k represents the algebraic curvature of the curve that the robot is trying to track, subscript "1" belongs to the curve, and subscript "2" belongs to the robot. Figure 32 shows two particles, i.e., the robot (quantities having subscript 2) and its projection along the curve which the robot is trying to track (quantities having subscript 1).

Define $\mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1$ as the difference between the position of the robot and its projection on the curve, $\rho = \|\mathbf{r}\|$ as the relative distance, and ϕ as the relative bearing between $\mathbf{x}_1$ and $\mathbf{x}_2$, then ϕ satisfies $\cos\phi = \mathbf{x}_1 \cdot \mathbf{x}_2$ and $\sin\phi = \mathbf{x}_1 \cdot \mathbf{y}_2$. We call (ρ, ϕ) the shape variables, take the time derivative of the shape variables, and obtain the following dynamics:

$$\dot{\rho} = -\sin\phi, \tag{214}$$

$$\dot{\phi} = \left(\frac{\mathscr{K}_1}{1 + \mathscr{K}_1\rho}\right)\cos\phi - u_2, \tag{215}$$

where u_2 is the steering control command for the robot. The following control law enables us to perform curve tracking:

$$u_2 = \left(\frac{\pm\mathscr{K}_1}{1 + \mathscr{K}_1\rho}\right)\cos\phi \pm K_p(\rho - \rho_0)\cos\phi + \mu\sin\phi, \tag{216}$$

where ρ_0 is the desired separation between the robot and the curve. The "$\pm$" sign provides different versions of the same control law, depending on whether the initial position of the robot is on the left or right of the curve. It has been theoretically justified that the controller achieves curve tracking [57]. Note that to enable any vehicle to follow a survey path, only calculation of u_2 is required in the control law, and the only position and orientation measurements of the vehicle are required for the control effort calculation. This makes it very easy to write computer code for performing complicated surveys.

5.2.3 Mapping Algorithm

An environment map is important in autonomous survey. A bathymetric map is essential to avoid damage to marine vehicles by accidental deployment in shallow water, a oil-concentration map explicitly shows the oil concentration data out of the autonomous survey, and we have showed a real-time modeling of tidal current helped improving the navigation performance of gliders, in our previous paper [95], where the ocean tidal flow around the glides is approximated by a series of temporal base functions and RBF (radial basis function) spatial basis functions, with the coefficient of the base functions updated in real time. In this chapter, we will use similar method as [95] and make a bathymetric map and a oil-concentration map according to the survey data. As both the depth and oil concentration for a location changes very slowly, if there is any change, within one round survey missions, we neglect the time-varying part and use only spatial basis functions.

In Figure 33, we overlay a square grid on the surface of interest (the lagoon in our case). The cells that lie outside the boundary of the lagoon are not included in our computation. Figure 33 shows two sample trajectories (red and green curves) taken by an AUV (the EcoMapper for example). When an AUV passes through a cell (and the time that the AUV was in the cell is greater than one sampling period), it records the sensor readings (depth or oil concentration in our case). Now we assume at least one reading is recorded in all unshaded cells (we use the position vector of the cell center to denote the cell) in Figure 33, and some of them might have more than one readings, for example, cell $\mathbf{x}_1$ has three measurement readings, i.e., $y_{1,1}$, $y_{1,2}$, and $y_{1,3}$, along the two AUV trajectories. Meanwhile, as our grid is made finer in order to get a high-resolution map, it is tedious to assign a survey path such that a vehicle passes through all the cells on our grid. Hence we may have cells (pockets) with no sensor reading, i.e., shaded (green) cells in Figure 33. Now we will make a map where each cell has one reading (a sensor measurement or a predicted value). In other words, we will assign one reading y_i to cell $\mathbf{x}_i$, $i = 1,...N$, where N is the number of cells on the map. For cells that has only one sensor measurement, y_i is that sensor measurement. For cells that has more than one sensor measurement, y_i is the average of all the sensor measurements in that cell, for example, $y_1 = (y_{1,1} + y_{1,2} + y_{1,3})/3$. For the empty cells, i.e., the cells that have no sensor measurement, we will predict the reading using measurements in non-empty cells. In other words,

Figure 33: The EcoMapper paths on a grid in a region of interest on the lagoon surface. Shaded squares show areas on the grid missed by the EcoMapper.

we need to predict $y_\star$ for an empty cell $\mathbf{x}_\star$, given the sensor readings $y_1,...y_n$ at non-empty cells $\mathbf{x}_1,...\mathbf{x}_n$. To achieve this objective, we build a map to predict the target value for any cell.

Similar with [95], we approximate a target value y (y can be sensor readings $y_1,y_2,...y_M$ or a prediction $y_\star$) for location $\mathbf{x}$ ($\mathbf{x}$ will be $\mathbf{x}_1,\mathbf{x}_2,...\mathbf{x}_M$, or $\mathbf{x}_\star$ corresponding to $y_1,y_2,...y_M$ or $y_\star$, respectively) by M basis functions, i.e., $\boldsymbol{\phi}_1(\mathbf{x})...\boldsymbol{\phi}_M(\mathbf{x})$, with weights $w_1,...,w_M$, so that

$$f(\mathbf{x}) = \boldsymbol{\phi}^T(\mathbf{x})\mathbf{w}, \tag{217}$$

$$y(\mathbf{x}) = f(\mathbf{x}) + \varepsilon, \tag{218}$$

where $\boldsymbol{\phi}(\mathbf{x}) = [\boldsymbol{\phi}_1(\mathbf{x})...\boldsymbol{\phi}_M(\mathbf{x})]^T$ and $\mathbf{w} = [w_1...w_M]^T$ contain unknown parameters which will be specified and solved later. ε is a Gaussian distributed measurement noise with zero mean and variance σ_n, i.e.,

$$\varepsilon \sim \mathcal{N}(0,\sigma_n^2). \tag{219}$$

Variance σ_n is also unknown and will be decided later. Now for location $\mathbf{x}_i$ and its corresponding measurements y_i, the following distribution holds:

$$P(y_i(\mathbf{x}_i)|\mathbf{w}) = \frac{1}{\sqrt{2\pi}\sigma_n}\exp(-\frac{(y_i - \boldsymbol{\phi}^T(\mathbf{x}_i)\mathbf{w})^2}{2\sigma_n^2})$$

84

When making a map using all the survey data, all the data are available for the map making algorithm, which differs from real-time tidal current modeling in [95], where the measurements are updated at some frequency and hence the weight $\mathbf{w}$ for basis functions is updated at the same frequency. Hence, here we put a zero mean Gaussian prior with covariance matrix Σ_p on the weight $\mathbf{w}$, i.e.,

$$\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \Sigma_p). \tag{220}$$

Note here "it is common but by no means necessary to consider Gaussian processes with a zero mean function. The zero-mean assumption is not necessarily a drastic limitation, since the mean of the posterior process is not confined to be zero" [96]. Now we can calculate the mean and the covariance of the Gaussian process $f(x) = \boldsymbol{\phi}(\mathbf{x})^T \mathbf{w}$ as follows:

$$\begin{aligned} k(\mathbf{x}, \mathbf{x}') &\doteq \mathbb{E}[(f(\mathbf{x}) - \mathbb{E}[f(\mathbf{x})])(f(\mathbf{x}') - \mathbb{E}[f(\mathbf{x}')])] \\ &= \mathbb{E}[f(\mathbf{x})f(\mathbf{x}')] \\ &= \boldsymbol{\phi}(\mathbf{x})^T \mathbb{E}(\mathbf{w}\mathbf{w}^T) \boldsymbol{\phi}(\mathbf{x}') \\ &= \boldsymbol{\phi}(\mathbf{x})^T \Sigma_p \boldsymbol{\phi}(\mathbf{x}'). \end{aligned} \tag{221}$$

Note here given the covariance matrix Σ_p, for any set of basis functions $\boldsymbol{\phi}$, we can compute the covariance function as $k(\mathbf{x}, \mathbf{x}') = \boldsymbol{\phi}(\mathbf{x})^T \Sigma_p \boldsymbol{\phi}(\mathbf{x}')$ according to Equation (221), and conversely, for every (positive definite) covariance function k, there exists a (possibly infinite) expansion in terms of basis functions, see section 4.3 in [96]. In this chapter, we specify the covariance function k for the mapping algorithm derivation and omit the calculation of basis functions, since their explicit forms are not directly used in our mapping algorithm. We adopt the standard form of covariance function as follows [97]:

$$k(\mathbf{x}, \mathbf{x}') \doteq \sigma_f^2 \exp\left(-\frac{1}{2}\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{l^2}\right) \tag{222}$$

where l is the length-scale parameter which defines the global smoothness of the function f, σ_f^2 denotes the amplitude or the signal variance.

Now we define $\mathbf{y} = [y_1...y_n]^T$, $\boldsymbol{X} = [\mathbf{x}_1,...,\mathbf{x}_n]^T$, $\Phi \doteq \Phi(\boldsymbol{X}) \doteq [\boldsymbol{\phi}(\mathbf{x}_1),...,\boldsymbol{\phi}(\mathbf{x}_n)]$, then

$$
\begin{aligned}
P(\mathbf{y}(\boldsymbol{X})|\mathbf{w}) &= \prod_{i=1}^{n} P(y_i(\mathbf{x}_i)|\mathbf{w}) \\
&= \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_n} \exp(-\frac{(y_i - \boldsymbol{\phi}^T(\mathbf{x})\mathbf{w})^2}{2\sigma_n^2}) \\
&= \frac{1}{(2\pi\sigma_n^2)^{\frac{n}{2}}} \exp(-\frac{|\mathbf{y} - \Phi^T\mathbf{w}|^2}{2\sigma_n^2})
\end{aligned}
$$

i.e.,

$$
\mathbf{y}(\boldsymbol{X})|\mathbf{w} \sim \mathcal{N}(\Phi^T\mathbf{w}, \sigma_n^2\mathbf{e}),
$$

where $\mathbf{e}$ denotes an identity matrix. As a result,

$$
\begin{aligned}
P(\mathbf{w}|\mathbf{y}(\boldsymbol{X})) &= \frac{P(\mathbf{y}(\boldsymbol{X})|\mathbf{w})P(\mathbf{w})}{P(\mathbf{y}(\boldsymbol{X}))} \\
&\propto P(\mathbf{y}(\boldsymbol{X})|\mathbf{w})P(\mathbf{w}) \\
&\propto \exp(-\frac{1}{2\sigma_n^2}(\mathbf{y} - \Phi^T\mathbf{w})^T(\mathbf{y} - \Phi^T\mathbf{w}))\exp(-\frac{1}{2}\mathbf{w}^T\Sigma_p^{-1}\mathbf{w}) \\
&\propto \exp(-\frac{1}{2}(\mathbf{w} - \bar{\mathbf{w}})^T(\frac{1}{\sigma_n^2}\Phi\Phi^T + \Sigma_p^{-1})(\mathbf{w} - \bar{\mathbf{w}}))
\end{aligned}
$$

Now define $\bar{\mathbf{w}} = \sigma_n^{-2}(\sigma_n^{-2}\Phi\Phi^T + \Sigma_p^{-1})^{-1}\Phi\mathbf{y}$ and $A = \sigma_n^{-2}\Phi\Phi^T + \Sigma_p^{-1}$, we get

$$
\mathbf{w}|\mathbf{y}(\boldsymbol{X}) \sim \mathcal{N}(\bar{\mathbf{w}}, A^{-1})
$$

Therefore, using $P(f(\mathbf{x}_\star)|\mathbf{y}(\boldsymbol{X})) = \int P(f(\mathbf{x}_\star)|\mathbf{w})P(\mathbf{w}|\mathbf{y}(\boldsymbol{X}))\mathrm{d}\mathbf{w}$, we get

$$
\begin{aligned}
f(\mathbf{x}_\star)|\mathbf{y}(\boldsymbol{X}) &\sim \mathcal{N}(\frac{1}{\sigma_n^2}\boldsymbol{\phi}_\star^T A^{-1}\boldsymbol{X}\mathbf{y}, \boldsymbol{\phi}_\star^T A^{-1}\boldsymbol{\phi}(\mathbf{x}_\star)) \\
&= \mathcal{N}(\boldsymbol{\phi}_\star^T\Sigma_p\Phi(\Phi^T\Sigma_p\Phi + \sigma_n^2\mathbf{e})^{-1}\mathbf{y}, \boldsymbol{\phi}_\star^T\Sigma_p\boldsymbol{\phi}_\star - \boldsymbol{\phi}_\star^T\Sigma_p\Phi(\Phi^T\Sigma_p\Phi + \sigma_n^2\mathbf{e})^{-1}\Phi^T\Sigma_p\boldsymbol{\phi}_\star) \\
&\doteq \mathcal{N}(K(\mathbf{x}_\star,\boldsymbol{X})(K(\boldsymbol{X},\boldsymbol{X}) + \sigma_n^2\mathbf{e})^{-1}\mathbf{y}, K(\mathbf{x}_\star,\mathbf{x}_\star) - K(\mathbf{x}_\star,\boldsymbol{X})(K(\boldsymbol{X},\boldsymbol{X}) + \sigma_n^2\mathbf{e})^{-1}K(\boldsymbol{X},\mathbf{x}_\star))
\end{aligned}
$$

$$(223)$$

where K is defined as

$$
K(\boldsymbol{X},\boldsymbol{X})_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j),
$$

$$
K(\mathbf{x}_\star,\boldsymbol{X})_{1,j} = k(\mathbf{x}_\star, \mathbf{x}_j),
$$

$$
K(\boldsymbol{X},\mathbf{x}_\star)_{i,1} = k(\mathbf{x}_i, \mathbf{x}_\star),
$$

$$
K(\mathbf{x}_\star,\mathbf{x}_\star) = k(\mathbf{x}_\star, \mathbf{x}_\star).
$$

Now we get distribution of $y_\star$ for any cell $\mathbf{x}_\star$, given the sensor readings $y_1, ... y_n$ at non-empty cells $\mathbf{x}_1, ... \mathbf{x}_n$. For detailed derivation, calculation and the background theory, please refer [96].

Using Equation (223), we can predict $y_\star$ for an empty cell $\mathbf{x}_\star$, if given parameters $\{\sigma_f, l, \sigma_n\}$. We will calculate $\{\sigma_f, l, \sigma_n\}$ using the available measurement data $\{\boldsymbol{X}, \mathbf{y}\}$. Using maximum-likelihood-estimation method, we choose $\{\sigma_f, l, \sigma_n\} = \underset{\{\sigma_f, l, \sigma_n\}}{\operatorname{argmax}} \prod_{i=1}^n P(y_i)$, therefore, it maximizes the probability that the measurements $\mathbf{y}$ appears if we use the model to predict the target value for cells X. Many optimization algorithm can be applied to solve $\{\sigma_f, l, \sigma_n\}$. In this chapter, we adopt the genetic algorithm, because of its simplicity to implementation, effectiveness for both convex and non-convex problems, and ability to avoid being trapped at a local optimal point. To reduce the computational cost when predict the values for empty cells, we use only 400 cells which are closest to one empty cell. In genetic algorithm, a population size 500 and a search space $\sigma_f \in (0, 100], l \in (0, 500], \sigma_n \in (0, 100]$ are used in this chapter. After the genetic algorithm produces $\{\sigma_f, l, \sigma_n\}$, Equation (223) will give the distribution of the prediction $y_\star$ for any empty cell $\mathbf{x}_\star$, and the mean value will be assigned to $\mathbf{x}_\star$.

5.3 *Experimental Results And Data Analysis*

During the 21-day survey for a tidal lagoon at the Grand Isle, Louisiana, where crude oil has been spotted along the beaches, we carried out field experiments and autonomous surveys. The experiments included parameter identification for ASV-Victoria, curve following control for both ASV-Victoria and AUV-Fetch 1, the success of which enables us to perform the later autonomous surveys. During the surveys, collaboration between the various vehicles shown in Figure 28 was performed. In a marine-robot survey, it is very important to know the environment and a bathymetric map is very helpful for us to decide which spots are safe for deploying our marine vehicles, as our vehicles could get damaged if they were deployed in extremely shallow locations. As the EcoMapper, which is equipped with sonar system for depth measurements, is much lighter and smaller with much smaller chance to get stranded and damaged, comparing with other vehicles in our survey, we first deployed the EcoMapper for autonomous missions over the entire lagoon to acquire the bathymetric data, based on which we made a complete bathymetric map for the lagoon, using the method in previous section. Once a bathymetric map was obtained, ASV-Victoria, which was equipped with

a crude-oil sensor, was deployed in a safe region for autonomous oil survey, in which the ASV was driven by the curve tracking controller described in previous section. We performed intense oil survey in an area and obtained the data for crude oil concentration. Using the mapping-making method in previous section, we made the crude-oil-concentration map to illustrate the oil concentration level. During our surveys, in addition to the surface oil concentration data, we also deployed ROV-β and collected water samples at various depths of a location. In this section, we presents the experiment results, including parameter identification and curve curve tracking, and the survey data analysis, including the bathymetric data and oil-concentration data.

5.3.1 Parameter Identification

We operated ASV-Vicoria in the open loop mode and sent seven sets of thruster speeds to the lower level thruster controllers, and then calculated K_1 and K_2 using Equations (203) and (204). Figures 34(a) and 34(b) show the result for the seven test runs. The dotted lines show the average values, i.e., $K_1 = 37.26$ and $K_2 = 38.4$.

5.3.2 Curve Tracking

To demonstrate the performance of our proposed curve-tracking control law, we carried out experiments for both ASV-Victoria and AUV-Fetch 1, and commanded them to track desired curves. Figures 35(a) and 35(b) show the straight-line-tracking-experiment results on ASV-Victoria, where $\rho_0 = 6m$ and control gains were $\mu = 5$ and $K_p = 1$. The dotted line in Figure 35(a) is the reference


(a) Estimation the parameter K_1 (b) Estimation the parameter K_2
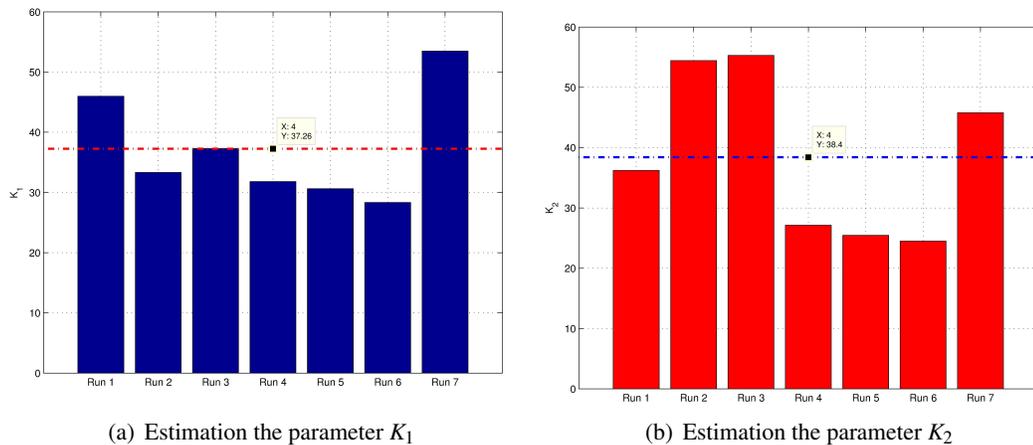
Figure 34: Results of parameter identification tests
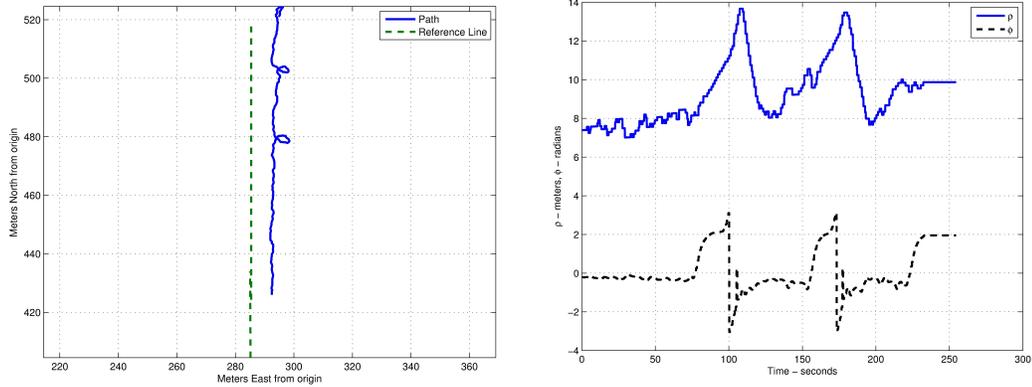
(a) Line following using ASV-Victoria

(b) Error analysis for line following using ASV-Victoria

Figure 35: Line following using ASV-Victoria

line the ASV was trying to follow and the solid curve is the path taken by the ASV. We can see that the ASV follows the line well, but there are some intermittent deviations. The details are shown in Figure 35(b), where the solid line represents ρ and the dotted line represents the errors in orientation ϕ. From Figure 35(b), we observe that the distance ρ is maintained just above 8 on an average, and that ϕ stays close to zero. When disturbances occur, ϕ comes back to zero very quickly. Data analysis suggests that some faults that occurred to the electronic thruster-speed controller caused the disturbances, despite which the vehicle recovered very soon and came back to tracking the line again, and shows the robustness of the control law.

We also carried out circular-path-tracking experiment for ASV-Victoria, where $\rho_0 = 4m$, $\mu = 5$, and $K_p = 1$. The radius of the desired circle was $R = 1m$. The vehicle motion was clockwise. Figure 36(a) shows the experiment result overlayed on a Google map, from which we can see that the ASV tracked the desired circle reasonably well, although the circles had a little displacement to the right, which was caused by the current in the lagoon (approximately 20 cm/s, from southwest to northeast). Figure 36(b) shows separation ρ and the error in orientation ϕ. Although the GPS localization errors, which was on average 3m at Grand Isle, caused some oscillations, we can see that ϕ was maintained close to zero and ρ was maintained around 5m on an average which equals $\rho_0 + R$, which suggests that the Victoria tracked the circular curve successfully in the presence of real environmental disturbances and localization error.

Straight-line-tracking experiment was also performed on Fetch 1, where $\rho_0 = 8m$, $\mu = 0.1$, and

(a) Tracking a circular curve using ASV-Victoria
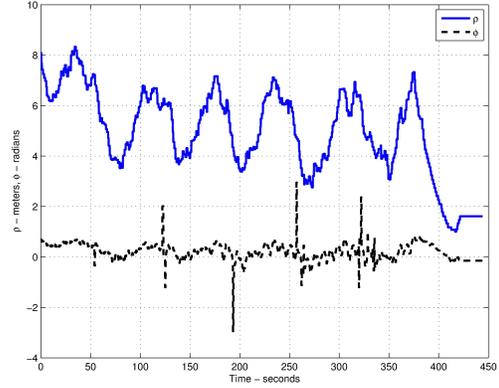
(b) Error analysis for curve tracking using ASV-Victoria

Figure 36: Circular curve following using ASV-Victoria
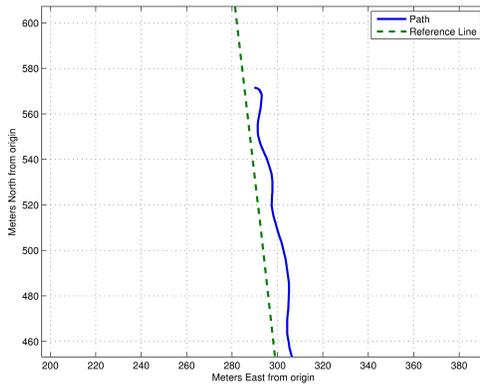
(a) Line following using Fetch 1
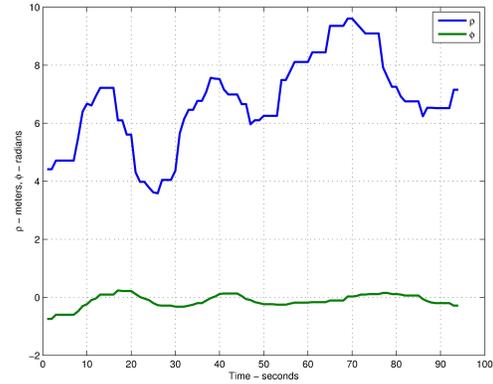
(b) Error analysis for line following using Fetch 1

Figure 37: Line following using AUV-Fetch 1

$K_p = 0.001$. The experimental results are shown in Figure37(a), where the dotted line is the reference line Fetch 1 was trying to follow and the solid line is the actual path of Fetch 1. Because the dynamics of the Fetch were notably slower than that of Victoria, smoother convergence is observed in Figure 37(a). Figure 37(b) shows separation ρ and the errors in orientation ϕ. We can see that the separation ρ converge to the desired value, i.e., $\rho_0 = 8m$, and ϕ stays close to zero.

Form the above experimental results, we can observe that the control laws described in Section 5.2 are robust in the presence of winds, water currents, tides, and engineering constraints such as sensor inaccuracy, localization errors, and network delays. In addition, the effectiveness of the proposed method is also demonstrated by the fact that the algorithms can be easily applied to and works effectively on different vehicles, such as ASV-Victoria and Fetch 1.
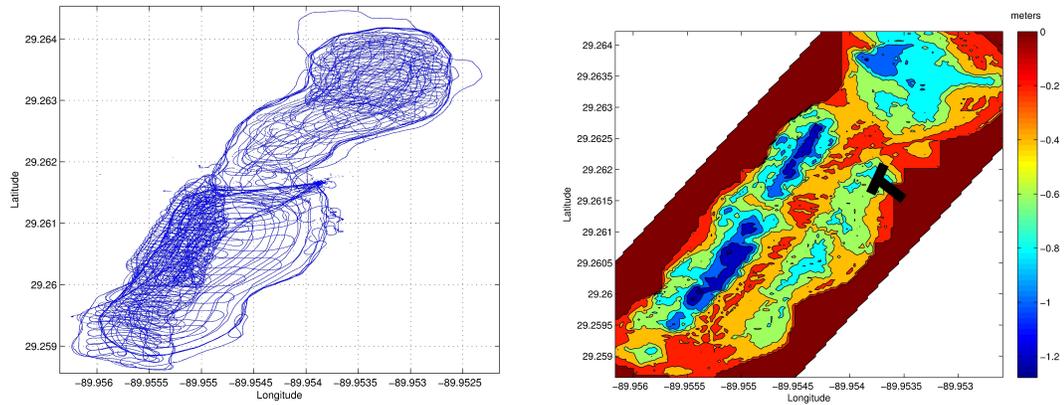
(a) A compilation of actual paths taken by the EcoMap-   (b) A depth map of the lagoon. The color bar shows the
per while on various bathymetric surveys.   depth of the lagoon from the surface in meters.

Figure 38: Generating a smooth bathymetric map

5.3.3 Bathymetry of the Lagoon

In the bathymetric surveys, the EcoMapper followed a path that covered the entire lagoon, and collected bathymetric data at a sample rate of 0.5 seconds along the path. Figure 38(a) shows the path taken by the EcoMapper while it performed a complete bathymetric survey of the lagoon. From Figure 38(a), we can see that the bathymetric data were not continuous (i.e., there were some pockets which were left unsampled). Using the mapping method in previous section, we produced a smooth depth map, by generating a square grid consisting of 22,500 cells, 150 cells along each side, to overlay the survey area, and predicting the depth for umsampled cell by Equation (223). To reduce the computational cost when predict the depth for a unsampled cell, instead of using the entire set of measurements, we use only data from 400 cells which are closest to that empty cell. After predict the depth for all empty cells, we obtain the depth map for the lagoon, shown in Figure 38(b). The dark red portions represent the unsampled parts, including the parts out of the boundary of lagoon, and some parts in the lagoon that are too shallow and too close to the shore, so might get the EcoMapper strand. The regions in blue are the deepest whereas the read areas are relatively shallower. The thick black T-shaped structure shows the location of the dock, where the vehicles were launched.

5.3.4   Surface Crude Oil Concentration Map

The ASV-Victoria intensely surveyed an area and collected surface-crude-oil-concentration data, based on which we generate the crude oil concentration map of that area, shown in Figure 39(a), using the same method that generates the bathymetric map. In Figure 39(a), the oil concentrations in red areas are higher. In order to get a better sense of the distribution of the measured surface crude oil, we overlay a contour plot of the concentration map, shown in Figure 39(a), on top of the depth map of the lagoon, shown in Figure 38(b). The overlays are shown in Figures 39(b) and 39(c). The black and red colored regions correspond to the contours on the depth map, i.e., Figure 38(b), with the black areas being deep and the red areas being shallow. Figure 39(c) provides a zoomed view of Figure 39(b), where the contours of the oil map in Figure 39(a) are shown in yellow and white.

On-field measurements have been perform to verify the bathymetric and oil map. The results suggest that our analysis result are close to the reality, which show the effectiveness of our proposed method in reconciling data from marine-robot survey.

5.4   Conclusions

We have developed a method for marine environmental surveys, using a fleet of heterogeneous marine vehicles. The path following controllers, the convergence and robustness of which we have mathematically proved, enables the vehicles to perform autonomous surveys, where the paths consist of lines and curves. In our control algorithm, we use simple dynamic models and simple control laws, which enables quick deployment of a fleet of autonomous vehicles. The mapping algorithms reconcile survey data from a variety of heterogeneous marine vehicles, and create a high fidelity visual representation of the desired survey data.

Using the proposed method, we performed a 21-day survey at the Grand Isle in Louisiana to evaluate the level of crude oil remaining in the area after the Deepwater Horizon oil spill. We presented the experimental results on controller performance that were tested on ASV-Victoria and AUV-Fetch 1. The results show the effectiveness and robustness of our control law in the presence of natural disturbances like wind, water currents, and engineering constraints such as sensor inaccuracy, localization errors, and network delays. With the help of the fleet of marine robots, we collected
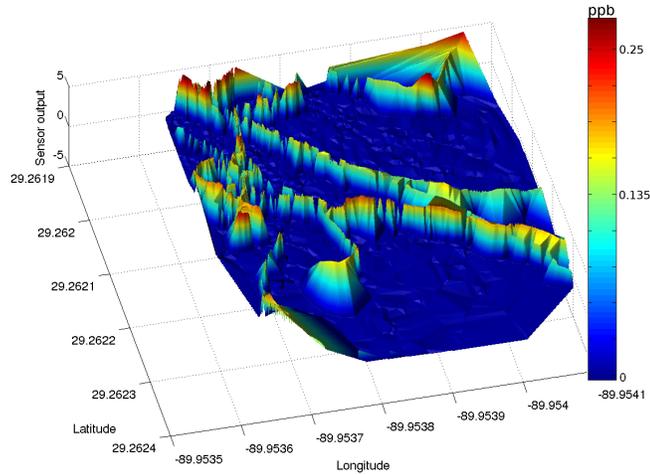
(a) Crude-oil-concentration map
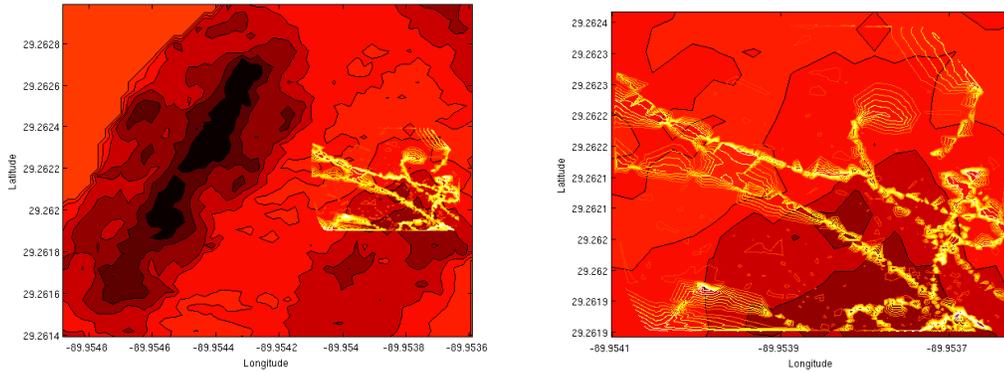
(b) The surface crude oil contour overlayed on the depth map of the lagoon.

(c) The surface crude oil contour overlayed on the depth map - a zoomed view.

Figure 39: Generating a crude-oil-concentration map

large amounts of survey data, including the bathymetric data and crude-oil data. The mapping algorithm successfully reconciled the survey data and created high fidelity visual representations of them.

From the survey data, we can see that there is crude oil remaining in the water in the coastal areas in the Gulf of Mexico after cleanup efforts. Although the concentration is low, there may be long-term unknown effects on the entire coastal ecosystem. The knowledge on the current oil spill disaster can help us get ready to handle possible disasters of a similar scale in the future, so we are grateful to have this chance to assist in this process using our marine vehicles.

CHAPTER VI

CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

The main contributions of this dissertation are summarized as follows:

- *Dynamic modeling of an underwater vehicle.* We developed a method for the dynamic modeling of an underwater vehicle with control surfaces. Applying the geometric properties of a vehicle, we theocratically derived the relationship between fin angles and hydrodynamic damping forces and moments. We proposed a procedure for identifying the hydrodynamic damping coefficients based on CFD calculations. We applied the method to the modeling of the EcoMapper, in which we used strip theory and open-water tests to identify parameters in the motion model. We validated the dynamic motion model of the EcoMapper with field experiments, the results of which were consistent with simulation results generated by the model. Therefore, the proposed model may be used to simulate the EcoMapper motions and to compute desired control input for the EcoMapper in real-time control. The modeling methods in this dissertation also apply to other underwater vehicles with control surfaces.

- *Formation control for AUVs with time delays.* We developed a robust formation controller for a group of fully-actuated AUVs with bounded time delays introduced by communication links so that the AUVs form and maintain a desired formation shape and the formation center tracks a desired trajectory. The controller design employed a generic 6DOF dynamic model for each AUV to describe its motions in the 3D space. We developed a method of decoupling the orientation control and the translation control for each AUV and applied the Jacobi transform to decouple the collective dynamics of all AUVs into formation-center and formation-shape dynamics, which can be controlled separately. Treating coupling in the formation dynamics as perturbations, we designed a robust formation-keeping controller. Comparing to other existing approaches with similar goals, this decoupling procedure simplified the entire design process. We provided both rigorous theoretical analysis and simulation results that justified

94

the effectiveness of the method.

- *Cooperative control for a human-AUV system.* We developed a cooperative control algorithm for a human-AUV system so that the HUV and AUVs form and maintain a desired formation while the formation center tracks a desired trajectory. We proposed a dynamic motion model for an HUV. The model captured human reactions to AUVs through an attention function. Using geometric methods, we decoupled the collective dynamics of the human-AUV system into formation-center, formation-shape, and formation-orientation dynamics. We developed a 3D curve tracking algorithm that enables a free particle to track any smooth curve in 3D space and used it for formation center control. We identified the dynamics constraint in the human-AUV system when the HUV and AUVs collaborate to perform both curve-tracking and formation-keeping tasks. We designed a formation-shape controller and a formation-orientation controller, and proved that they enable the formation shape to converge to the desired one. In our method, control forces are gauge invariant and calculated based only on sensor measurements so that vehicles do not need to communicate and they have freedom to choose their own coordinate systems.

- *Marine environmental survey and data analysis method.* We proposed a marine environmental survey method using a fleet of heterogeneous AUVs and presented a mapping method that reconciles data from heterogeneous marine vehicles on multiple different paths and creates a high-fidelity visual representation of the survey data. We carried out field experiments surveying a coastal lagoon in Grand Isle, Louisiana, which was polluted by crude oil during the Deepwater Horizon oil spill. In the experiments, we validated our methods, evaluated the performance of the controllers, analyzed the bathymetric and crude-oil data collected by our autonomous vehicles in survey missions, and created bathymetric and oil-concentration maps. According to the survey data, crude oil remained in the water in the coastal area in the Gulf of Mexico after cleanup efforts. This information may help evaluate and improve current cleanup efforts.

6.2   Future Work

We developed a method of modeling an underwater vehicle. In future work, we will extend the method to the modeling of a partially immersed surface vehicle that moves in a horizontal plane. In this dissertation, time delay in a multi-AUV system was assumed to be a bounded unknown constant for the formation control design. In future work, to obtain more general results, we will study cases in which time delay is time varying. In addition, while current work investigated cooperative control for a human-AUV system comprised of one HUV and multiple AUVs, future work will include cooperative control of a system comprised of multiple HUVs and multiple AUVs.

APPENDIX A

DERIVATION OF UNDERWATER VEHICLE DYNAMICS

We derive the dynamics in Equations (1) and (2) for underwater vehicles following the books by Thor I. Fossen [70] and Gianluca Antonelli [71].

A.1 Rigid Body Kinematics

A.1.1 Two coordinate frames

Two orthogonal right-handed coordinate frames, shown in Figure (2), are usually used to describe the configuration of an underwater vehicle: one is the earth-fixed frame $\{I\}$, the other is the body-fixed frame $\{B\}$. The earth-fixed frame is denoted as $\Sigma_I, O_I - x_I y_I z_I$. The origin O_I is fixed at a position at the ocean surface, e.g., the position of a mother ship. The body-fixed frame is donated as $\Sigma_B, O_B - x_B y_B z_B$. The origin O_B of the body-fixed frame can be selected as the center of mass (COM) of the underwater vehicle. Note that the vertical axes z_I and z_B are pointing downwards.

The position, orientation, linear velocity, and angular velocity are needed for describing the motion of the rigid body. Each of the four quantities is a 3×1 vector. These vectors can be described in either the earth-fixed frame or the body-fixed frame. The following four vectors are commonly used:

$$\eta_1 = [x, y, z]^T; \eta_2 = [\phi, \theta, \psi]^T; v_1 = [u, v, w]^T; v_2 = [p, q, r]^T. \tag{224}$$

Here η_1 and η_2 describe the position and orientation of the vehicle in the earth-fixed frame, respectively. v_1 and v_2 describe the linear and angular velocity in the body-fixed frame, respectively. The reason for this convention is that we are usually interested in knowing the position and orientation of the vehicle relative to the mother ship. Therefore it is more convenient to describe the position and orientation in the earth-fixed frame. On the other hand, the velocities are often controlled by actuators attached to the vehicle, hence it is more convenient to describe them in the body-fixed frame. The linear velocities $[u, v, w]^T$ are referred as surge, sway, and heave, and the angular velocities $[p, q, r]^T$ are referred as roll, pitch, and yaw.

A.1.2 Coordinate transform for linear velocity

The orientation of an underwater vehicle is described by Euler angles, which are a special selection of a class of angles that can be used to describe rigid body orientation. Three Euler angles used in the earth-fixed frame are roll angle ϕ, pitch angle θ, and yaw angle ψ, so $\eta_2 = [\phi, \theta, \psi]^T$ describes the orientation of a vehicle in the earth-fixed frame. $v_2 = [p, q, r]^T$ denotes the angular velocity in the body-fixed frame. The positive direction of rotation around one of the axes x_I, y_I, and z_I is determined by the following sequence $x \rightarrow y \rightarrow z \rightarrow x$. For example, the rotation from z to x is positive, which corresponds to the vehicle pulling its head up, and the pitch angle θ increasing. Each Euler angle determines a rotation matrix around its corresponding axes as follows:

$$
R_z(\psi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \; R_y(\theta) = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}, \; R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}
$$

$$(225)$$

The coordinate transform between the earth-fixed and the body-fixed frames are described by a rotation matrix ${}^I_B R$. It is customary to describe ${}^I_B R$ by three consecutive rotations determined by the Euler angles. Note that the order of these rotations is not arbitrary. In this dissertation we use the zyx-convention. First rotate a yaw angle ψ about z axis, then rotate a pitch angle θ about y axis, and finally rotate a roll angle ϕ about x axis. These rotations transform the earth-fixed frame into the body-fixed frame. The rotation sequence can be written as ${}^I_B R = R_z(\psi)R_y(\theta)R_x(\phi)$. Now we get

$$
{}^I_B R = R_z(\psi)R_y(\theta)R_x(\phi) = \begin{bmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} .
$$

$$(226)$$

The rotation matrix ${}^I_B R \in SO(3)$ has the property: ${}^I_B R^{-1} = {}^I_B R^T$. As ${}^B_I R = {}^I_B R^{-1}$, we have

$$
{}^B_I R = {}^I_B R^T = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\ \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta \end{bmatrix} . \quad (227)
$$

Now the linear velocity of a vehicle in the earth-fixed frame can be obtained by the coordinate

transformation

$$\dot{\eta}_1 = {}^I_B R v_1. \tag{228}$$

The inverse velocity transformation can be written as:

$$v_1 = {}^I_B R^{-1} \dot{\eta}_1 = {}^B_I R \dot{\eta}_1. \tag{229}$$

A.1.3 Coordinate transform for angular velocity

The Euler angles $\eta_2 = [\phi, \theta, \psi]^T$ are measured in the earth-fixed frame $\{I\}$, but the angular velocity v_2 is measured in the body-fixed frame B. Because $\int_0^t v_2(\tau)d\tau$ does not have immediate physical interpretation, the angular velocity vector $v_2 = [p, q, r]^T$ cannot be integrated directly to obtain the actual angular coordinates in the earth-fixed frame. In other words, $[p, q, r] \neq [\dot{\psi}, \dot{\theta}, \dot{\phi}]$.

For ${}^I_B R \in SO(3)$, we have ${}^I_B R \cdot {}^I_B R^T = \mathbf{e}_3$. Take derivative on the both sides, we get $\frac{d}{dt} {}^I_B R = -{}^I_B R {}^I_B \dot{R}^T {}^I_B R$. As a result, $\frac{d}{dt} {}^I P = \frac{d}{dt} {}^I_B R^B P = -{}^I_B R {}^I_B \dot{R}^T {}^I_B R^B P = -{}^I_B R {}^I_B \dot{R}^T {}^I P$. Here ${}^B P$ is a fixed vector with respect to the body-fixed frame and ${}^I P$ is its description in the earth-fixed frame. In addition, it can be easily verified that

$$- {}^I_B R \cdot {}^I_B \dot{R}^T$$

$$= - R_z(\psi) R_y(\theta) R_x(\phi) R_x^T(\phi) R_y^T(\theta) \dot{R}_z^T(\psi) - R_z(\psi) R_y(\theta) R_x(\phi) R_x^T(\phi) \dot{R}_y^T(\theta) R_z^T(\psi)$$

$$\quad - R_z(\psi) R_y(\theta) R_x(\phi) \dot{R}_x^T(\phi) R_y^T(\theta) R_z^T(\psi)$$

$$= - R_z(\psi) \dot{R}_z^T(\psi) - R_z(\psi) R_y(\theta) \dot{R}_y^T(\theta) R_z^T(\psi) - R_z(\psi) R_y(\theta) R_x(\phi) \dot{R}_x^T(\phi) R_y^T(\theta) R_z^T(\psi)$$

$$= \begin{bmatrix} 0 & -\dot{\psi} & 0 \\ \dot{\psi} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + R_z(\psi) \begin{bmatrix} 0 & 0 & \dot{\theta} \\ 0 & 0 & 0 \\ -\dot{\theta} & 0 & 0 \end{bmatrix} R_z^T(\psi) + R_z(\psi) R_y(\theta) \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -\dot{\phi} \\ 0 & \dot{\phi} & 0 \end{bmatrix} R_y^T(\theta) R_z^T(\psi)$$

$$= \begin{bmatrix} 0 & -\dot{\psi} & 0 \\ \dot{\psi} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & \dot{\theta}\cos\psi \\ 0 & 0 & \dot{\theta}\sin\psi \\ -\dot{\theta}\cos\psi & -\dot{\theta}\sin\psi & 0 \end{bmatrix} + \begin{bmatrix} 0 & \dot{\phi}\sin\theta & \dot{\phi}\sin\psi\cos\theta \\ -\dot{\phi}\sin\theta & 0 & -\dot{\phi}\cos\psi\cos\theta \\ -\dot{\phi}\cos\theta\sin\psi & \dot{\phi}\cos\theta\cos\psi & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & -\dot{\psi}+\dot{\phi}\sin\theta & \dot{\theta}\cos\psi+\dot{\phi}\sin\psi\cos\theta \\ \dot{\psi}-\dot{\phi}\sin\theta & 0 & \dot{\theta}\sin\psi-\dot{\phi}\cos\psi\cos\theta \\ -\dot{\theta}\cos\psi-\dot{\phi}\sin\psi\cos\theta & -\dot{\theta}\sin\psi+\dot{\phi}\cos\psi\cos\theta & 0 \end{bmatrix} \tag{230}$$

We can observe that $-{}_B^I R \cdot {}_B^I \dot{R}^T$ is a skew-symmetric matrix and it is called the *angular-velocity*

matrix. We know that for any skew-symmetric matrix $S = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}$ (note: $S^T = -S$),

if we define a 3×1 column vector $\omega = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix}$, then it can be verified that $SP = \omega \times P$, where P

is any vector and "×" denotes the vector cross product. The 3×1 vector ω, which corresponds to

the 3×3 angular-velocity matrix, is called the *angular-velocity vector*. We use the notion $S(\cdot)$ to

represent an operator that turns a three dimensional vector into a skew-symmetric matrix. In our

case, the skew-symmetric matrix $-{}_B^I R \cdot {}_B^I \dot{R}^T$ corresponds to a vector ω defined by

$$\omega = \begin{bmatrix} -\dot{\theta}\sin\psi + \dot{\phi}\cos\psi\cos\theta \\ \dot{\theta}\cos\psi + \dot{\phi}\sin\psi\cos\theta \\ \dot{\psi} - \dot{\phi}\sin\theta \end{bmatrix}. \tag{231}$$

We call this ω *the angular velocity in the earth-fixed frame.* Therefore, $S(\omega) = -{}_B^I R \cdot {}_B^I \dot{R}^T$.

The relationship between ω and $[\dot{\psi}, \dot{\theta}, \dot{\phi}]^T$ can be obtained as follows:

$$\omega = \begin{bmatrix} \cos\psi\cos\theta & -\sin\psi & 0 \\ \sin\psi\cos\theta & \cos\psi & 0 \\ -\sin\theta & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \doteq J(\eta_1)\dot{\eta}_2, \tag{232}$$

so the relationship between $v_2 = [p, q, r]^T$ and $[\dot{\psi}, \dot{\theta}, \dot{\phi}]^T$ can be established as

$$v_2 = {}_I^B R \omega = {}_I^B R \cdot J(\eta_1)\dot{\eta}_2 = J_{k,o}\dot{\eta}_2, \tag{233}$$

where

$$J_{k,o} = {}_I^B R \cdot J(\eta_1)$$

$$
= \begin{bmatrix}
\cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\
\sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \sin\phi\cos\theta \\
\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi & \cos\phi\cos\theta
\end{bmatrix}
$$

$$
\begin{bmatrix}
\cos\psi\cos\theta & -\sin\psi & 0 \\
\sin\psi\cos\theta & \cos\psi & 0 \\
-\sin\theta & 0 & 1
\end{bmatrix}
$$

$$
= \begin{bmatrix}
1 & 0 & -\sin\theta \\
0 & \cos\phi & \sin\phi\cos\theta \\
0 & -\sin\phi & \cos\phi\cos\theta
\end{bmatrix} . \tag{234}
$$

A.2 Rigid Body Dynamics

In this section we establish the relationship between motions of a rigid body and the external forces. When deriving the equations of motions of a underwater vehicle, we assume: (1) the vehicle is rigid and; (2) the earth-fixed frame is inertial.

Consider the body-fixed coordinate system $x_B y_B z_B$ rotating with an angular velocity $\omega = [\omega_1, \omega_2, \omega_3]^T$ about an earth-fixed coordinate system $x_I y_I z_I$, as shown in Figure 1. p_0 is the position of the origin of the body-fixed frame in the earth-fixed frame. p_c is the position of COM in the earth-fixed frame. r is the position of a point in the body relative to the origin of the body-fixed frame. The mass of a rigid body is defined by $m = \int_V \rho dV$, where ρ is the density of the rigid body, dV is an infinitesimal volume, and V is the entire volume of the rigid body. If the origin of the body-fixed frame is not at the COM, the displacement from the origin of the body-fixed frame to the COM is

$$r_c = \frac{1}{m} \int_V (p - p_0)\rho dV. \tag{235}$$

We can also get that:

$$p_c = \frac{1}{m} \int_V p\rho dV. \tag{236}$$

A.2.1 Inertia matrix

We define a matrix called the *inertia matrix* as:

$$I_0 = \int_V S^T(p - p_0) S(p - p_0) \rho \, dV, \tag{237}$$

where $S(p - p_0)$ is the skew-symmetric matrix corresponding to the vector $p - p_0$, then I_0 is a 3×3 matrix:

$$I_0 \triangleq \begin{bmatrix} I_x & I_{xy} & I_{xz} \\ I_{yx} & I_y & I_{yz} \\ I_{zx} & I_{zy} & I_z \end{bmatrix}. \tag{238}$$

Here I_x, I_y, and I_z are the moments of inertia about x_I, y_I, and z_I axes, respectively. I_{xy}, I_{yx}, I_{xz}, I_{zx}, I_{yz}, and I_{zy} are the products of inertia. They are defined as:

$$I_x = \int_V (y^2 + z^2) \rho \, dV; \quad I_{xy} = \int_V xy \rho \, dV = \int_V yx \rho \, dV = I_{yx}$$

$$I_y = \int_V (x^2 + z^2) \rho \, dV; \quad I_{xz} = \int_V xz \rho \, dV = \int_V zx \rho \, dV = I_{zx}$$

$$I_z = \int_V (x^2 + y^2) \rho \, dV; \quad I_{yz} = \int_V yz \rho \, dV = \int_V zy \rho \, dV = I_{zy} \tag{239}$$

A.2.2 Angular momentum

The angular momentum of a particle with respect to a given origin is defined as $\vec{k} \triangleq (p - p_0) \times m\dot{p}$. In addition, the angular velocity ω and the linear velocity $\dot{p}$ is related by $\dot{p} - \dot{p}_0 = \omega \times (p - p_0)$, i.e., $\dot{p} = \omega \times (p - p_0) + \dot{p}_0$. Therefore, the angular momentum of a rigid body about the origin of the body-fixed frame is:

$$\begin{aligned} \vec{k} &= \int_V (p - p_0) \times \dot{p} \rho \, dV \\ &= \int_V (p - p_0) \times \dot{p}_0 \rho \, dV + \int_V (p - p_0) \times (\dot{p} - \dot{p}_0) \rho \, dV \\ &= \int_V p \rho \, dV \times \dot{p}_0 - \int_V \rho \, dV \, p_0 \times \dot{p}_0 + I_0 \omega \\ &= m(p_c - p_0) \times \dot{p}_0 + I_0 \omega \\ &= I_0 \omega + m(p_c - p_0) \times \dot{p}_0 \\ &= I_0 \omega + m\dot{p}_0 \times (p_0 - p_c) \end{aligned} \tag{240}$$

If the COM coincides with the origin of the body-fixed frame, we have $I_c = I_0 = \int_V S^T(p - p_c)S(p - p_c)\rho dV$. If the center of buoyancy (COB) is selected as the origin of the body-fixed frame, we have $I_b = \int_V S^T(p - p_b)S(p - p_b)\rho dV$. We will use *Steiner's theorem* to calculate I_b. *Steiner's theorem* is as follows:

Steiner's theorem:

$$I_b\omega = I_c\omega + mS^T(p_c - p_b)S(p_c - p_b)\omega. \tag{241}$$

Proof

$$
\begin{aligned}
I_b\omega &= \int_V S^T(p - p_b)S(p - p_b)\rho dV\,\omega \\
&= \int_V (p - p_b) \times (\omega \times (p - p_b))\rho dV \\
&= \int_V (p - p_c + p_c - p_b) \times (\omega \times (p - p_c + p_c - p_b))\rho dV \\
&= \int_V (p - p_c) \times (\omega \times (p - p_c))\rho dV + \int_V (p_c - p_b) \times (\omega \times (p - p_c))\rho dV \\
&\quad + \int_V (p - p_c) \times (\omega \times (p_c - p_b))\rho dV + \int_V (p_c - p_b) \times (\omega \times (p_c - p_b))\rho dV \\
&= I_c\omega + (p_c - p_b) \times (\omega \times \int_V (p - p_c))\rho dV + \int_V (p - p_c)\rho dV \times (\omega \times (p_c - p_b)) \\
&\quad + (p_c - p_b) \times (\omega \times (p_c - p_b)) \int_V \rho dV \\
&= I_c\omega + (p_c - p_b) \times (\omega \times (mp_c - p_c m)) + (mp_c - p_c m) \times (\omega \times (p_c - p_b)) \\
&\quad + (p_c - p_b) \times (\omega \times (p_c - p_b))m \\
&= I_c\omega + mS^T(p_c - p_b)S(p_c - p_b)\omega
\end{aligned}
$$

■

We have derived the angular momentum about an origin that is not necessarily the COM:

$$\vec{k} = I_0\omega + \dot{p}_0 \times m(p_0 - p_c) \tag{242}$$

Using *Steiner's theorem*, we can rewrite it as

$$\vec{k} = I_0 \omega + \dot{p}_0 \times m(p_0 - p_c)$$

$$= I_c \omega + mS^T(p_c - p_0)S(p_c - p_0)\omega + m\dot{p}_0 \times (p_0 - p_c)$$

$$= I_c \omega + m(p_c - p_0) \times (\dot{p}_c - \dot{p}_0) + m\dot{p}_0 \times (p_0 - p_c)$$

$$= I_c \omega + m(p_c - p_0) \times (\dot{p}_c) - m(p_c - p_0) \times \dot{p}_0 + m\dot{p}_0 \times (p_0 - p_c)$$

$$= I_c \omega + m(p_c - p_0) \times (\dot{p}_c) + m\dot{p}_0 \times (p_c - p_0) + m\dot{p}_0 \times (p_0 - p_c)$$

$$= I_c \omega + m(p_c - p_0) \times (\dot{p}_c)$$

$$= I_c \omega + m\dot{p}_c \times (p_0 - p_c) \tag{243}$$

Now we can derive the relationship between I_0 and I_b. Note that $p - p_c = {}_B^I R^B p$ and $S(p - p_c) = S({}_B^I R^B p) = {}_B^I R S(^B p) {}_I^B R$, we get

$$I_0 = \int_V S^T(p - p_0)S(p - p_0)\rho dV$$

$$= \int_V ({}_B^I R S^T(^B p) {}_I^B R {}_B^I R S(^B p) {}_I^B R) dV$$

$$= {}_B^I R I_b {}_I^B R. \tag{244}$$

Therefore, $I_b = {}_I^B R I_0 {}_B^I R$.

A.2.3 Rigid body dynamics

Let f_i and $\sum_{i=1}^n f_i \times (p_i - p_0)$ be the external forces and torques applied to the rigid body, respectively. From $\omega = {}_B^I R v_2$, we get

$$\dot{\omega} = {}_B^I R \dot{v}_2 + {}_B^I \dot{R} v_2$$

$$= {}_B^I R \dot{v}_2 + S(\omega) {}_B^I R v_2$$

$$= {}_B^I R \dot{v}_2 + S(\omega)\omega$$

$$= {}_B^I R \dot{v}_2 \tag{245}$$

Because $p_c - p_0 = {}^I_B R({}^B r_c)$, we get

$$\dot{p}_c - \dot{p}_0 = {}^I_B \dot{R}({}^B r_c) + {}^I_B R({}^B \dot{r}_c)$$

$$= S(\omega) {}^I_B R({}^B r_c)$$

$$= \omega \times {}^I_B R({}^B r_c)$$

$$= {}^I_B R v_2 \times {}^I_B R({}^B r_c)$$

$$= {}^I_B R(v_2 \times {}^I_B R({}^B r_c))$$

$$= {}^I_B R S(v_2)({}^B r_c), \tag{246}$$

in which we have used the fact $({}^B \dot{r}_c) = 0$, as the COM will not move relative to the origin in the body-fixed frame. In addition, since $\dot{p}_0 = {}^I_B R v_1$, we obtain

$$\dot{p}_c = \dot{p}_0 + {}^I_B R S(v_2)({}^B r_c)$$

$$= {}^I_B R v_1 + {}^I_B R S(v_2)({}^B r_c)$$

$$= {}^I_B R[v_1 + S(v_2)({}^B r_c)]. \tag{247}$$

Now Newton's Law yields the following equation

$$\sum_i f_i = m \frac{d}{dt} \dot{p}_c = m \frac{d}{dt} {}^I_B R[v_1 + S(v_2)({}^B r_c)]$$

$$= m \frac{d}{dt} {}^I_B R[v_1 + v_2 \times ({}^B r_c)]$$

$$= m[{}^I_B \dot{R}(v_1 + v_2 \times ({}^B r_c)) + {}^I_B R(\dot{v}_1 + \dot{v}_2 \times ({}^B r_c))]$$

$$= m[{}^I_B R S(v_2)(v_1 + v_2 \times ({}^B r_c)) + {}^I_B R(\dot{v}_1 + \dot{v}_2 \times ({}^B r_c))]$$

$$= m[{}^I_B R(v_2 \times v_1 + v_2 \times (v_2 \times ({}^B r_c))) + {}^I_B R(\dot{v}_1 + \dot{v}_2 \times ({}^B r_c))] \tag{248}$$

Define $\tau_1 = {}^B_I R \sum_i f_i$, then

$$\tau_1 = m v_2 \times v_1 + m v_2 \times (v_2 \times ({}^B r_c)) + m \dot{v}_1 + m \dot{v}_2 \times ({}^B r_c)). \tag{249}$$

We have derived

$$\vec{k} = I_c \omega + m \dot{p}_c \times (p_0 - p_c), \tag{250}$$

and we also know

$$I_c \omega = {}^I_B R I_c^b {}^B_I R \cdot {}^I_B R v_2 = {}^I_B R I_c^b v_2, \tag{251}$$

and

$$mp̈_c \times (p_0 - p_c) = m_B^I R(v_1 + v_2 \times^B r_c) \times^B r_c, \tag{252}$$

so we get

$$\begin{aligned} \vec{k} &= {}_B^I R I_c^b v_2 + m_B^I R(v_1 + v_2 \times^B r_c) \times^B r_c \\ &= {}_B^I R[I_c^b v_2 + m(v_1 + v_2 \times^B r_c) \times^B r_c] \\ &= {}_B^I R[(I_c^b - mS(^B r_c)S(^B r_c))v_2 + mv_1 \times^B r_c]. \end{aligned} \tag{253}$$

Let

$$I_{ob} = I_c^b - mS(^B r_c)S(^B r_c), \tag{254}$$

then

$$\vec{k} = {}_B^I R(I_{ob}v_2 + mv_1 \times^B r_c), \tag{255}$$

as a result

$$\dot{\vec{k}} = \omega \times {}_B^I R[I_{ob}v_2 + mv_1 \times^B r_c] + {}_B^I R[I_{ob}\dot{v}_2 + m\dot{v}_1 \times^B r_c]. \tag{256}$$

We define

$$\tau_2 = ({}_B^I R)^{-1}\dot{\vec{k}}, \tag{257}$$

then

$$\tau_2 = I_{ob}\dot{v}_2 + v_2 \times (I_{ob}v_2) + mv_2 \times (v_1 \times^B r_c) + m\dot{v}_1 \times^B r_c. \tag{258}$$

A.2.4 General form of rigid body dynamics

Equations (249) and (258) are usually written in a component form according to the SNAMA notation ([70]), where $\tau_1 = [X,Y,Z]^T$ denotes the external forces, $\tau_2 = [K,M,N]^T$ denotes the moment of external forces. $v_1 = [u,v,w]^T$ and $v_2 = [p,q,r]^T$ denote the linear and angular velocity in the body-fixed frame, respectively, and $(^B r_c) = [x_G,y_G,z_G]^T$ denotes the center of gravity the body-fixed frame, as before. Using these notations, Equations (249) and (258) become

$$M_{RB}\dot{v} + C_{RB}(v)v = \tau_{RB}, \tag{259}$$

where $v = [v_1, v_2]^T = [u,v,w,p,q,r]^T$ is the body-fixed linear and angular velocity vector and $\tau_{RB} = [\tau_1, \tau_2]^T = [X,Y,Z,K,M,N]^T$ is a generalized vector of external forces and moments. The

parameterization of the rigid-body inertia matrix M_{RB} is

$$M_{RB} = \begin{bmatrix} m\mathbf{e}_{3\times3} & -mS(^B r_c) \\ -mS(^B r_c)) & I_{ob} \end{bmatrix} = \begin{bmatrix} m & 0 & 0 & 0 & mz_G & -my_G \\ 0 & m & 0 & -mz_G & 0 & mx_G \\ 0 & 0 & m & my_G & -mx_G & 0 \\ 0 & mz_G & -my_G & I_x & -I_{xy} & -I_{zx} \\ -mz_G & 0 & mx_G & -I_{yx} & I_y & -I_{yz} \\ my_G & -mx_G & 0 & -I_{zx} & -I_{zy} & I_z \end{bmatrix} \tag{260}$$

We can see that M_{RB} is constant, symmetric, and positive definite, i.e., $\dot{M}_{RB} = \mathbf{0}$, $M_{RB} = M_{RB}^T > \mathbf{0}$.

C_{RB} is given as following:

$$C_{RB}(v) = \begin{bmatrix} mS(v_2) & -mS(v_2)S(^B r_c) \\ -mS(v_2)S(^B r_c) & -S(I_{ob}v_2) \end{bmatrix}$$

$$\tag{261}$$

Noticing

$$\begin{bmatrix} mS(v_2) & -mS(v_2)S(^B r_c) \\ -mS(v_2)S(^B r_c) & -S(I_{ob}v_2) \end{bmatrix} v = \begin{bmatrix} 0_{3\times3} & -mS(v_1) - mS(v_2)S(^B r_c) \\ -mS(v_1) - mS(v_2)S(^B r_c) & -S(I_{ob}v_2) \end{bmatrix} v,$$

we define

$$C'_{RB}(v) \triangleq \begin{bmatrix} 0_{3\times3} & -mS(v_1) - mS(v_2)S(^B r_c) \\ -mS(v_1) - mS(v_2)S(^B r_c) & -S(I_{ob}v_2) \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -m(y_G q + z_G r) & m(y_G p + w) & m(z_G p - v) \\ m(x_G q - w) & -m(z_G r + x_G p) & m(z_G q + u) \\ m(x_G r + v) & m(y_G r - u) & -m(x_G p + y_G q) \end{bmatrix}$$

107

$$\left.\begin{array}{ccc} m(y_Gq+z_Gr) & -m(x_Gq-w) & -m(x_Gr+v) \\ -m(y_Gp+w) & m(z_Gr+x_Gp) & -m(y_Gr-u) \\ -m(z_Gp-v) & -m(z_Gq+u) & m(x_Gp+y_Gq) \\ 0 & I_{yx}q+I_{xz}p+I_zr & -I_{yz}r-I_{xy}p-I_yq \\ -I_{yx}q-I_{xz}p-I_zr & 0 & I_{xz}r+I_{xy}q+I_xp \\ I_{yz}r+I_{xy}p+I_yq & -I_{xz}r-I_{xy}q-I_xp & 0 \end{array}\right], \tag{262}$$

and get

$$C_{RB}(v)v \triangleq C'_{RB}(v)v, \tag{263}$$

so we can use C'_{RB} to substitute C_{RB}. It can be easily seen that $C'_{RB}(v) = -C'^T_{RB}(v)$.

If $^B r_c = 0$, then $M_{RB} = \begin{bmatrix} m\mathbf{e}_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & I_{ob} \end{bmatrix}$ and $C_{RB}(v) = \begin{bmatrix} mS(v_2) & 0_{3\times3} \\ 0_{3\times3} & -S(I_{ob}v_2) \end{bmatrix}$

A.3   Underwater Vehicle Dynamics

In Equation (259), τ_{RB} represents the external forces and moments acting on the vehicle. They include radiation-induced forces and moments, for example, terms caused by the added inertia, i.e., $M_A\dot{v}+C_A(v)v$, terms caused by hydrodynamic damping, i.e., D_vv, and restoring force term $g(\eta)$. They also include environmental forces and moments like ocean currents, waves, and wind. This part is ignored in this dissertation. τ_{RB} also includes propulsion forces and moments, for example, thruster/propeller forces and control surfaces/rudder forces, which are used to control the vehicle. Hence the right-hand side of (259) is $\tau_{RB} = \tau_H + \tau$, where $\tau_H = -M_A\dot{v} - C_A(v)v - D(v)v - g(\eta)$ represents the hydrodynamic forces and moments, and τ represents the propulsion forces and moments. Therefore, 6DOF dynamic model of motions of an underwater vehicle is

$$M\dot{v}+C(v)v+D(v)v+g(\eta) = \tau, \tag{264}$$

where $M \triangleq M_{RB} + M_A$ and $C(v) \triangleq C_{RB}(v) + C_A(v)$. M_A is a 6×6 added inertia matrix defined as

$$M_A \triangleq - \begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ K_{\dot{u}} & K_{\dot{v}} & K_{\dot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & N_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix} \tag{265}$$

For instance, the hydrodynamic added mass force Y_A along y-axis due to an acceleration $\dot{u}$ in the x-direction is written as $Y_A = Y_{\dot{u}}\dot{u}$, therefore, $Y_{\dot{u}} \triangleq \frac{\partial Y}{\partial \dot{u}}$. For a rigid body moving through an ideal fluid, the hydrodynamic Coriolis and centripetal matrix $C_A(v)$ can always be parameterized such that $C_A(v)$ is skew-symmetrical, i.e., $C_A(v) = -C_A^T(v)$, $\forall v \in R^6$. If the body is completely submerged in water, the velocity of the vehicle is low, and the vehicle has three planes of symmetry, which are common for underwater vehicles, M_A and C_A matrices have the following structure

$$M_A = -diag\{X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}\}, \tag{266}$$

$$C_A = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v \\ 0 & 0 & 0 & Z_{\dot{w}}w & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & -Y_{\dot{v}}v & X_{\dot{u}}u & 0 \\ -Z_{\dot{w}}w & Y_{\dot{v}}v & 0 & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ Z_{\dot{w}}w & 0 & -X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ -Y_{\dot{v}}v & X_{\dot{u}}u & 0 & -M_{\dot{q}}q & k_{\dot{p}}p & 0 \end{bmatrix}. \tag{267}$$

The viscosity of the fluid also causes dissipative drag and lift forces on the body. We call them the damping terms. In general the damping terms for an underwater vehicle moving at high speed will be highly nonlinear and coupled. If we approximate them by assuming that the vehicle has three planes of symmetry and that the damping terms higher than second order can be neglected, then $D(v)$ has a diagonal structure with only linear and quadratic damping terms on the diagonal, so

$$D(v) = -\text{diag}\{X_u, Y_v, Z_w, K_p, M_q, N_r\} - \text{diag}\{X_{u|u|}|u|, Y_{v|v|}|v|, Z_{w|w|}|w|, K_{p|p|}|p|, M_{q|q|}|q|, N_{r|r|}|r|\} \tag{268}$$

Now we look into the gravity and buoyancy. We know that the weight of the body is $W = mg$ and the buoyancy force is $-B = -\rho g \nabla$ (the positive direction of B points upwards) in the earth-fixed frame, where g is the acceleration of gravity, ρ is the fluid density, and ∇ is the volume of fluid displaced by the vehicle. We can transform the weight and buoyancy to the body-fixed coordinate system using $f_G(\eta_2) = J_1^{-1}(\eta_2) \begin{bmatrix} 0 \\ 0 \\ W \end{bmatrix}$ and $f_B(\eta_2) = -J_1^{-1}(\eta_2) \begin{bmatrix} 0 \\ 0 \\ B \end{bmatrix}$. Consequently, the restoring force and moment vector in the body-fixed frame is:

$$g(\eta) = - \begin{bmatrix} f_G(\eta) + f_B(\eta) \\ r_G \times f_G(\eta) + r_B \times f_B(\eta) \end{bmatrix} = \begin{bmatrix} (W-B)\sin\theta \\ -(W-B)\cos\theta\sin\phi \\ -(W-B)\cos\theta\cos\phi \\ -(y_G W - y_B B)\cos\theta\cos\phi + (z_G W - z_B B)\cos\theta\sin\phi \\ (z_G W - z_B B)\sin\theta + (x_G W - x_B B)\cos\theta\cos\phi \\ -(x_G W - x_B B)\cos\theta\sin\phi - (y_G W - y_B B)\sin\theta \end{bmatrix}.$$

(269)

APPENDIX B

HYDRODYNAMIC PARAMETER ESTIMATION
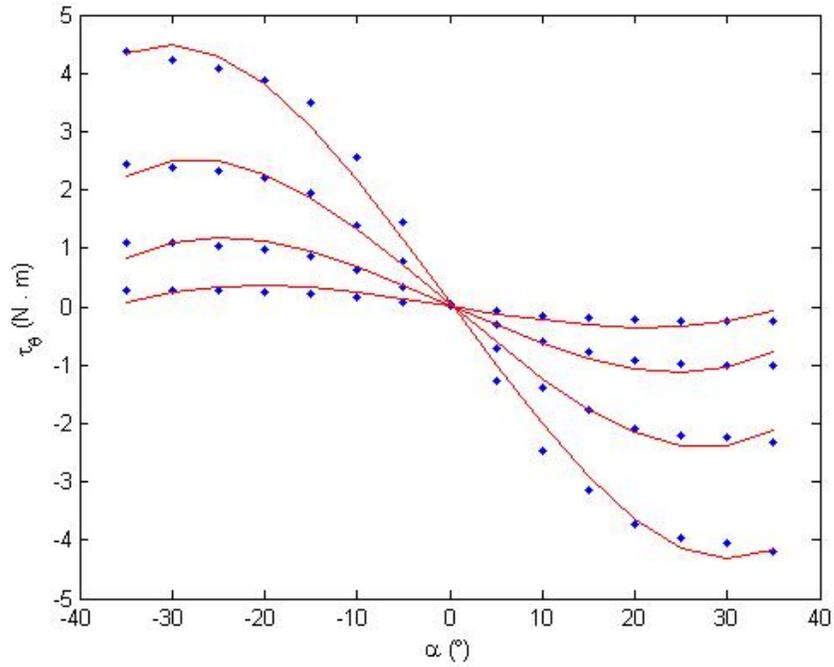
Figure 40: Horizontal-fin angle α VS τ_θ
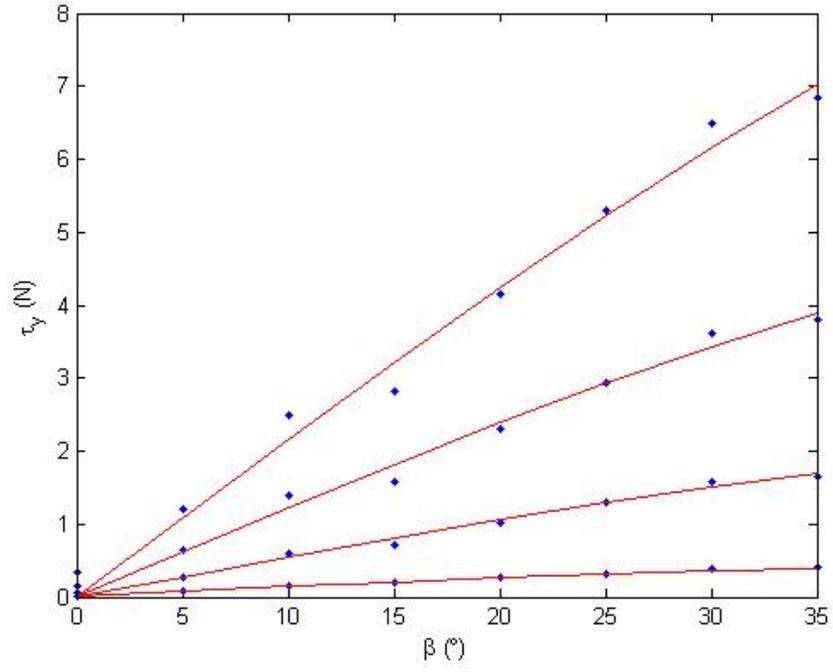
Figure 41: Vertical-fin angle β VS τ_y
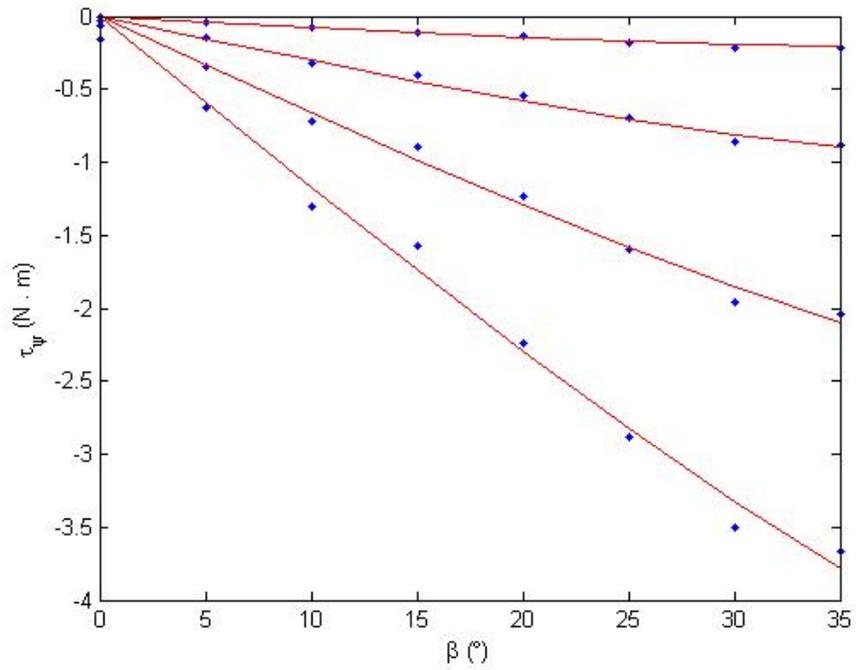
Figure 42: Vertical-fin angle β VS τ_ψ
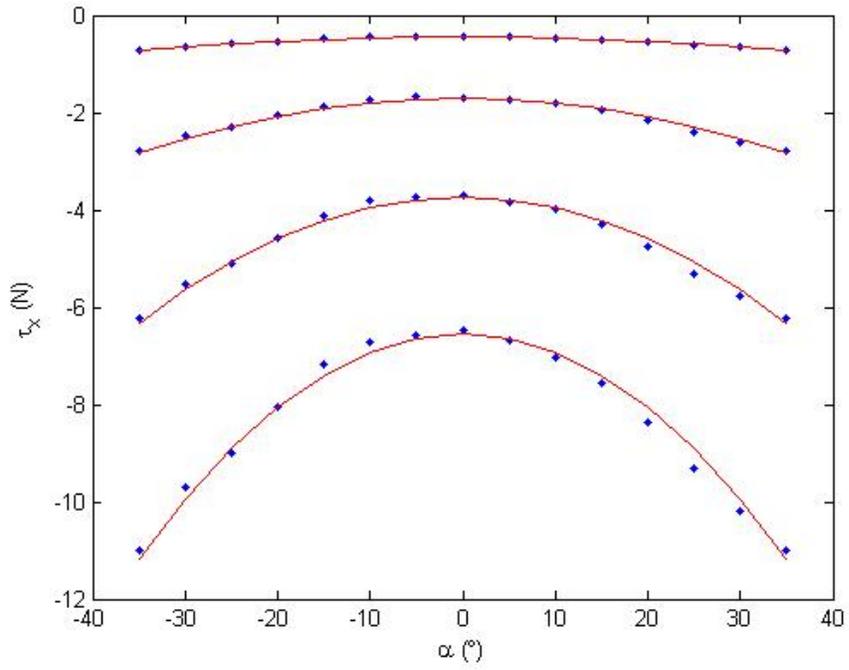
Figure 43: Horizontal-fin angle α VS τ_x
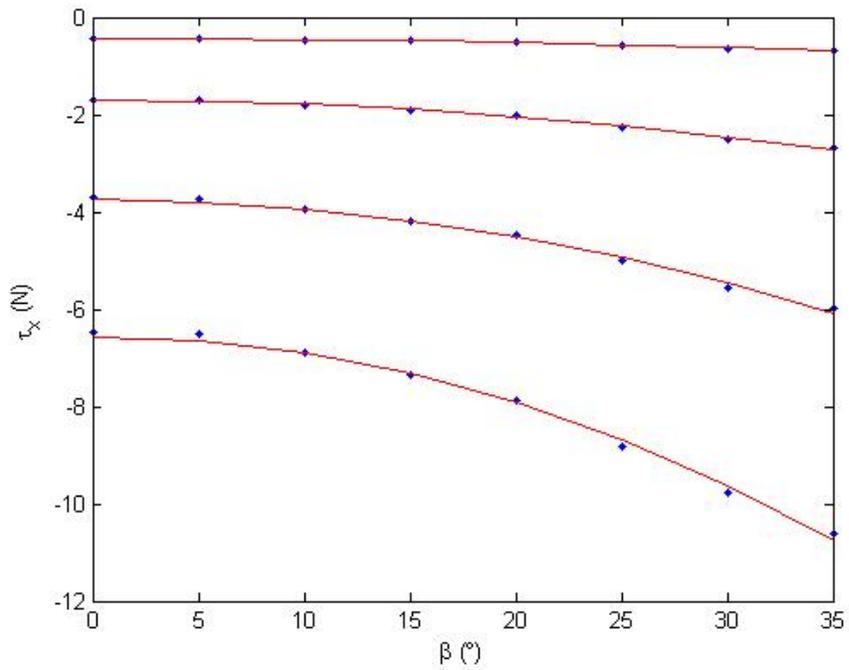
Figure 44: Vertical-fin angle β VS τ_x

113

APPENDIX C

$\mathbf{G}'$ IN EQUATION (78)

$$\mathbf{G}' = \begin{bmatrix} g'_{11} & g'_{12} & g'_{13} \\ g'_{21} & g'_{22} & g'_{23} \\ g'_{31} & g'_{32} & g'_{33} \end{bmatrix},$$

where

$$
\begin{aligned}
g'_{11} &= \frac{X_u}{m - X_{\dot{u}}}(\cos\theta\cos\psi)^2 \\
&\quad + \frac{Y_v}{m - Y_{\dot{v}}}(\cos\psi\sin\theta\sin\phi - \cos\phi\sin\psi)^2 \\
&\quad + \frac{Z_w}{m - Z_{\dot{w}}}(\cos\phi\cos\psi\sin\theta + \sin\phi\sin\psi)^2, \\
g'_{12} &= \frac{X_u}{m - X_{\dot{u}}}(\cos^2\theta\sin\psi\cos\psi) \\
&\quad + \frac{Y_v}{m - Y_{\dot{v}}}(\cos\phi\cos\psi + \sin\theta\sin\phi\sin\psi)(\cos\psi\sin\theta\sin\phi - \cos\phi\sin\psi) \\
&\quad + \frac{Z_w}{m - Z_{\dot{w}}}(\cos\phi\cos\psi\sin\theta + \sin\phi\sin\psi)(\cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi), \\
g'_{13} &= -\frac{X_u}{m - X_{\dot{u}}}\cos\theta\sin\theta\cos\psi \\
&\quad + \frac{Y_v}{m - Y_{\dot{v}}}\cos\theta\sin\phi(\sin\theta\cos\psi\sin\phi - \sin\psi\cos\phi) \\
&\quad + \frac{Z_w}{m - Z_{\dot{w}}}\cos\theta\cos\phi(\sin\theta\cos\psi\cos\phi + \sin\psi\sin\phi), \\
g'_{21} &= \frac{X_u}{m - X_{\dot{u}}}\cos^2\theta\cos\psi\sin\psi \\
&\quad + \frac{Y_v}{m - Y_{\dot{v}}}(\sin\theta\sin\phi\sin\psi + \cos\phi\cos\psi)(\cos\psi\sin\theta\sin\phi - \cos\phi\sin\psi) \\
&\quad + \frac{Z_w}{m - Z_{\dot{w}}}(\cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi)(\cos\phi\cos\psi\sin\theta + \sin\phi\sin\psi), \\
g'_{22} &= \frac{X_u}{m - X_{\dot{u}}}(\cos\theta\sin\psi)^2 \\
&\quad + \frac{Y_v}{m - Y_{\dot{v}}}(\cos\phi\cos\psi + \sin\theta\sin\phi\sin\psi)^2 \\
&\quad + \frac{Z_w}{m - Z_{\dot{w}}}(\cos\psi\sin\phi - \cos\phi\sin\theta\sin\psi)^2,
\end{aligned}
$$

$$g'_{23} = -\frac{X_u}{m - X_{\dot{u}}} \cos\theta \sin\theta \sin\psi$$
$$+ \frac{Y_v}{m - Y_{\dot{v}}} \cos\theta \sin\phi (\sin\theta \sin\phi \sin\psi + \cos\phi \cos\psi)$$
$$+ \frac{Z_w}{(m - Z_{\dot{w}})} \cos\theta \cos\phi (\sin\theta \sin\psi \cos\phi - \sin\phi \cos\psi),$$

$$g'_{31} = -\frac{X_u}{m - X_{\dot{u}}} \cos\theta \cos\psi \sin\theta$$
$$+ \frac{Y_v}{m - Y_{\dot{v}}} (\sin\phi \cos\psi \sin\theta - \cos\phi \sin\psi) \cos\theta \sin\phi$$
$$+ \frac{Z_w}{m - Z_{\dot{w}}} (\cos\phi \cos\psi \sin\theta + \sin\phi \sin\psi) \cos\theta \cos\phi,$$

$$g'_{32} = -\frac{X_u}{m - X_{\dot{u}}} \sin\theta \cos\theta \sin\psi$$
$$- \frac{Z_w}{m - Z_{\dot{w}}} (\sin\phi \cos\psi - \cos\phi \sin\theta \sin\psi) \cos\theta \cos\phi$$
$$+ \frac{Y_v}{m - Y_{\dot{v}}} (\cos\phi \cos\psi + \sin\phi \sin\theta \sin\psi) \cos\theta \sin\phi,$$

$$g'_{33} = \frac{X_u}{m - X_{\dot{u}}} \sin^2\theta$$
$$+ \frac{Y_v}{m - Y_{\dot{v}}} (\cos\theta \sin\psi)^2$$
$$+ \frac{Z_w}{m - Z_{\dot{w}}} (\cos\theta \cos\psi)^2.$$

APPENDIX D

IMPLEMENTATION OF NATURAL FRENET FRAME

For a curve with non-zero curvature in 3D space, we use $\mathbf{r}$ to denote a point on the curve and use $\mathbf{T}$, $\mathbf{N}$, and $\mathbf{B}$ to denote the tangent vector, the principal normal vector, and the binormal vector, respectively, i.e.,

$$\mathbf{T} = \frac{\mathrm{d}\mathbf{r}}{\mathrm{d}s} \tag{270}$$

$$\mathbf{N} = \frac{\frac{\mathrm{d}\mathbf{T}}{\mathrm{d}s}}{\left\| \frac{\mathrm{d}\mathbf{T}}{\mathrm{d}s} \right\|} \tag{271}$$

$$\mathbf{B} = \mathbf{T} \times \mathbf{N} \tag{272}$$

where s denotes the length of the curve, then $(\mathbf{T}, \mathbf{N}, \mathbf{B})$ is the unique Frenet-Serret frame for the curve, and the 3D curve can be describe by the following equation:

$$\frac{\mathrm{d}\mathbf{T}}{\mathrm{d}s} = \kappa \mathbf{N} \tag{273}$$

$$\frac{\mathrm{d}\mathbf{N}}{\mathrm{d}s} = -\kappa \mathbf{T} + \widetilde{\tau} \mathbf{B} \tag{274}$$

$$\frac{\mathrm{d}\mathbf{B}}{\mathrm{d}s} = -\widetilde{\tau} \mathbf{N} \tag{275}$$

where κ and $\widetilde{\tau}$ are curvature and torsion, respectively. In our work, we only care about the shape of the curve, so that we assume the curve to be a unit speed curve, i.e., $\frac{\mathrm{d}s}{\mathrm{d}t} = 1$, therefore,

$$\begin{bmatrix} \dot{\mathbf{T}} \\ \dot{\mathbf{N}} \\ \dot{\mathbf{B}} \end{bmatrix} = \begin{bmatrix} 0 & \kappa & 0 \\ -\kappa & 0 & \widetilde{\tau} \\ 0 & -\widetilde{\tau} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{T} \\ \mathbf{N} \\ \mathbf{B} \end{bmatrix} \tag{276}$$

where

$$\mathbf{T} = \dot{\mathbf{r}} \tag{277}$$

$$\kappa = \|\dot{\mathbf{T}}\| \tag{278}$$

$$\mathbf{N} = \frac{\dot{\mathbf{T}}}{\kappa} \tag{279}$$

$$\mathbf{B} = \mathbf{T} \times \mathbf{N} \tag{280}$$

$$\widetilde{\tau} = -\dot{\mathbf{B}} \cdot \mathbf{N} \tag{281}$$

Here "·" denotes the inner product. Note that a curve with zero curvature, i.e., $\kappa = 0$, does not have Frenet-Serret representation. For any curve with non-zero curvature, Frenet-Serret representation always exists and is unique.

For the natural Frenet frame defined in Equation (135), we have $\mathbf{X} = \mathbf{T}$, which means that we can rotate the Frenet-Serret frame around the direction of $\dot{\mathbf{r}}$ for an angle $\widetilde{\alpha}$ to get the Natural Frenet frame, i.e.,

$$\begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \\ \mathbf{Z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\widetilde{\alpha} & s\widetilde{\alpha} \\ 0 & -s\widetilde{\alpha} & c\widetilde{\alpha} \end{bmatrix} \begin{bmatrix} \mathbf{T} \\ \mathbf{N} \\ \mathbf{B} \end{bmatrix} \tag{282}$$

or equivalently

$$\begin{bmatrix} \mathbf{T} \\ \mathbf{N} \\ \mathbf{B} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\widetilde{\alpha} & -s\widetilde{\alpha} \\ 0 & s\widetilde{\alpha} & c\widetilde{\alpha} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \\ \mathbf{Z} \end{bmatrix}. \tag{283}$$

Plug Equation (283) into left part of Equation (276), we get

$$\begin{bmatrix} \dot{\mathbf{T}} \\ \dot{\mathbf{N}} \\ \dot{\mathbf{B}} \end{bmatrix} = \begin{bmatrix} 0 & \kappa & 0 \\ -\kappa & 0 & \widetilde{\tau} \\ 0 & -\widetilde{\tau} & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\widetilde{\alpha} & -s\widetilde{\alpha} \\ 0 & s\widetilde{\alpha} & c\widetilde{\alpha} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \\ \mathbf{Z} \end{bmatrix} = \begin{bmatrix} 0 & \kappa c\widetilde{\alpha} & -\kappa s\widetilde{\alpha} \\ -\kappa & \widetilde{\tau} s\widetilde{\alpha} & \widetilde{\tau} c\widetilde{\alpha} \\ 0 & -\widetilde{\tau} c\widetilde{\alpha} & \widetilde{\tau} s\widetilde{\alpha} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \\ \mathbf{Z} \end{bmatrix}$$

$$\tag{284}$$

From Equation (283) we also get

$$
\begin{bmatrix} \dot{\mathbf{T}} \\ \dot{\mathbf{N}} \\ \dot{\mathbf{B}} \end{bmatrix} = \frac{d}{dt}\left(\begin{bmatrix} 1 & 0 & 0 \\ 0 & c\widetilde{\alpha} & -s\widetilde{\alpha} \\ 0 & s\widetilde{\alpha} & c\widetilde{\alpha} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \\ \mathbf{Z} \end{bmatrix} \right) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\widetilde{\alpha} & -s\widetilde{\alpha} \\ 0 & s\widetilde{\alpha} & c\widetilde{\alpha} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{X}} \\ \dot{\mathbf{Y}} \\ \dot{\mathbf{Z}} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & -s\widetilde{\alpha}\dot{\widetilde{\alpha}} & -c\widetilde{\alpha}\dot{\widetilde{\alpha}} \\ 0 & c\widetilde{\alpha}\dot{\widetilde{\alpha}} & -s\widetilde{\alpha}\dot{\widetilde{\alpha}} \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \\ \mathbf{Z} \end{bmatrix}
$$

(285)

Plug equation (284) to the above equation, we get

$$
\begin{bmatrix} \dot{\mathbf{X}} \\ \dot{\mathbf{Y}} \\ \dot{\mathbf{Z}} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\widetilde{\alpha} & -s\widetilde{\alpha} \\ 0 & s\widetilde{\alpha} & c\widetilde{\alpha} \end{bmatrix}^{-1} \left(\begin{bmatrix} 0 & \kappa c\widetilde{\alpha} & -\kappa s\widetilde{\alpha} \\ -\kappa & \widetilde{\tau}s\widetilde{\alpha} & \widetilde{\tau}c\widetilde{\alpha} \\ 0 & -\widetilde{\tau}c\widetilde{\alpha} & \widetilde{\tau}s\widetilde{\alpha} \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 \\ 0 & -s\widetilde{\alpha}\dot{\widetilde{\alpha}} & -c\widetilde{\alpha}\dot{\widetilde{\alpha}} \\ 0 & c\widetilde{\alpha}\dot{\widetilde{\alpha}} & -s\widetilde{\alpha}\dot{\widetilde{\alpha}} \end{bmatrix} \right) \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \\ \mathbf{Z} \end{bmatrix}
$$

$$
= \begin{bmatrix} 0 & \kappa c\widetilde{\alpha} & -\kappa s\widetilde{\alpha} \\ -\kappa c\widetilde{\alpha} & 0 & \widetilde{\tau}+\dot{\widetilde{\alpha}} \\ \kappa s\widetilde{\alpha} & -\widetilde{\tau}-\dot{\widetilde{\alpha}} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{X} \\ \mathbf{Y} \\ \mathbf{Z} \end{bmatrix}
$$

(286)

Compare Equation (286) with Equation (135) we get

$$
u = -\kappa s\widetilde{\alpha} \tag{287}
$$

$$
v = \kappa c\widetilde{\alpha} \tag{288}
$$

$$
\widetilde{\tau} + \dot{\widetilde{\alpha}} = 0 \tag{289}
$$

Therefore,

$$
\widetilde{\alpha} = -\widetilde{\tau}t + \zeta \tag{290}
$$

and then

$$
u = \kappa s\big(\widetilde{\tau}t - \zeta\big),
$$

$$
v = \kappa c\big(\widetilde{\tau}t - \zeta\big). \tag{291}
$$

Here ζ is an arbitrary constant.

REFERENCES

[1] T. Balch and R.C. Arkin. Behavior-based formation control for multirobot teams. *Robotics and Automation, IEEE Transactions on*, 14(6):926–939, 1998.

[2] R.S. Smith and F.Y. Hadaegh. Distributed estimation, communication and control for deep space formations. *IET Control Theory Applications*, 1(2):445–451, 2007.

[3] 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 *Automation and Logistics, 2007 IEEE International Conference on*, pages 569–573, Aug. 2007.

[4] S. Emrani, A. Dirafzoon, and H.A. Talebi. Adaptive distributed formation control of multiple autonomous underwater vehicles. In *Control Applications, 2011 IEEE International Conference on*, pages 693–698, Sep. 2011.

[5] E. Yang and D. Gu. Nonlinear formation-keeping and mooring control of multiple autonomous underwater vehicles. *Mechatronics, IEEE/ASME Transactions on*, 12(2):164–178, 2007.

[6] S. Fan, Z. Feng, and L. Lian. Collision free formation control for multiple autonomous underwater vehicles. In *2010 IEEE OCEANS*, pages 1–4, May 2010.

[7] R. Cui, D. Xu, and W. Yan. Formation control of autonomous underwater vehicles under fixed topology. In *Control and Automation, IEEE International Conference on*, pages 2913–2918, June 2007.

[8] W. Yan, R. Cui, and D. Xu. Formation control of underactuated autonomous underwater vehicles in horizontal plane. In *Automation and Logistics, IEEE International Conference on*, pages 822–827, Sep. 2008.

[9] W. Dong and J. Farrell. Decentralized cooperative control of multiple nonholonomic dynamic systems with uncertainty. *Automatica*, 45(3):706–710, 2009.

[10] P.K.C. Wang. Navigation strategies for multiple autonomous mobile robots moving in formation. In *Intelligent Robots and Systems, IEEE/RSJ International Workshop on*, pages 486–493, Sep. 1989.

[11] N. E. Leonard, D. A. Paley, F. Lekien, R. Sepulchre, D. M. Fratantoni, and R. E. Davis. Collective motion, sensor networks and ocean sampling. *Proceedings of IEEE*, 95(1):48–74, 2007.

[12] R. Ellison and D. Slocum. High spatial resolution mapping of water quality and bathymetry with a person-deployable, low cost autonomous underwater vehicle. In *OCEANS 2008*, pages 1–7, Sep. 2008.

[13] G. Hollinger, U. Mitra, and G. Sukhatme. Active and adaptive dive planning for dense bathymetric mapping. In *Proc. International Conference on Experimental Robotics (ISER)*, pages 1–7, 2012.

[14] R.R. Murphy, E. Steimle, M. Hall, M. Lindemuth, D. Trejo, S. Hurlebaus, Z. Medina-Cetina, and D. Slocum. Robot-assisted bridge inspection after hurricane IKE. In *Safety, Security Rescue Robotics (SSRR), 2009 IEEE International Workshop on*, pages 1–5, Nov. 2009.

[15] S. Mukhopadhyay, C. Wang, S. Bradshaw, S. Maxon, L. Hicks, M. Patterson, and F. Zhang. Controller performance of marine robots in reminiscent oil surveys. In *Proc. 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2012)*, pages 1766–1771, 2012.

[16] Y. Hwang. Hydrodynamic modeling of LMRS unmanned underwater vehicle and tow tank test validation. In *Proceedings of OCEANS 2003*, pages 1425–1430, Sep. 2003.

[17] E.A. de Barros, J.L.D. Dantas, A.M. Pascoal, and E. Desa. Investigation of normal force and moment coefficients for an AUV at nonlinear angle of attack and sideslip range. *Oceanic Engineering, IEEE Journal of*, 33(4):538–549, 2008.

[18] C.D. Williams, T.L. Curtis, J.M. Doucet, M.T. Issac, and F. Azarsina. Effects of hull length on the hydrodynamic loads on a slender underwater vehicle during manoeuvres. In *OCEANS 2006*, pages 1–6, Sep. 2006.

[19] T. Prestero. Verification of a Six-Degree of Freedom Simulation Model for the REMUS Autonomous Underwater Vehicle. Master's thesis, Massachusetts Institute of Technology and Woods Hole Oceanographic Institution, Cambridge, Massachusetts, 2001.

[20] P. Jagadeesh and K. Murali. Investigation of alternative turbulence closure models for axis symmetric underwater hull forms. *The Journal of Ocean Technology*, 1(2):37–57, 2006.

[21] N. Mulvany, L. Tu, J.Y.and Chen, and B. Anderson. Assessment of two-equation turbulence modeling for high reynolds number hydrofoil flow. *International Journal for Numerical Methods in Fluids*, 45:275–299, 2004.

[22] D. A. Boger and J. J. Dreyer. Prediction of hydrodynamic forces and moments for underwater vehicles using overset grids. In *Proceeding of 44th AIAA Aerospace Sciences Meeting and Exhibit*, pages 1–13, 2006.

[23] B. J. Racine and E. G. Paterson. CFD-based method for simulation of marine-vehicle maneuvering. In *Proceeding of 35th AIAA Fluid Dynamics Conference and Exhibit*, pages 1–22, 2005.

[24] C. Chin and M. Lau. Modeling and testing of hydrodynamic damping model for a complex-shaped remotely-operated vehicle for control. *Journal of Marine Science and Application*, 11:150–163, 2012.

[25] Y. Wang, W. Yan, and J. Li. Passivity-based formation control of autonomous underwater vehicles. *IET Control Theory and Applications*, 6(4):518–525, 2012.

[26] P.L. Kempker, A.C.M. Ran, and J.H. van Schuppen. A formation flying algorithm for autonomous underwater vehicles. In *2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, pages 1293–1298, 2011.

[27] S. Kalantar and U.R. Zimmer. Distributed shape control of homogeneous swarms of autonomous underwater vehicles. *Autonomous Robots*, 22(1):37–53, 2007.

[28] S.P. Hou and C.C. Cheah. Can a simple control scheme work for a formation control of multiple autonomous underwater vehicles? *IEEE Transactions on Control Systems Technology*, 19(5):1090–1101, 2011.

[29] R. Olfati-Saber, J. A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95:215–233, 2007.

[30] M. Porfiri, D. G. Roberson, and D.J. Stilwell. Tracking and formation control of multiple autonomous agents: A two-level consensus apporach. *Automatica*, 43:1318–1328, 2007.

[31] F. Zhang. Geometric cooperative control of particle formations. *Automatic Control, IEEE Transactions on*, 55(3):800–803, 2010.

[32] R. Olfati-Saber and R.M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

[33] P.A. Bliman and G. Ferrari-Trecate. Average consensus problems in networks of agents with delayed communiations. *Automatica*, 44:1985–1995, 2008.

[34] R. Carli, F. Fagnani, A. Speranzon, and S. Zampieri. Communication constraints in the average consensues problem. *Automatica*, 44:671–684, 2008.

[35] W. Wang and J.E. Slotine. Contraction analysis of time-delayed communications and group cooperation. *IEEE Transactions on Automatic Control*, 51(4):712–717, 2006.

[36] P.F. Hokayem, D. Stipanović, and M.W. Spong. Reliable control of multi-agent formations. In *2007 American Control Conference*, pages 1882–1887, 2007.

[37] D. Kulić and E. Croft. Pre-collision safety strategies for human-robot interaction. *Auton. Robots*, 22(2):149–164, 2007.

[38] E. Sisbot, L. Marin-Urias, X. Broquere, D. Sidobre, and R. Alami. Synthesizing robot motions adapted to human presence. *International Journal of Social Robotics*, 2(3):329–343, 2010.

[39] A. Moon, C.A.C. Parker, E.A. Croft, and H. F M Van der Loos. Did you see it hesitate? - empirically grounded design of hesitation trajectories for collaborative robots. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 1994–1999, 2011.

[40] D. Kulic and E. Croft. Anxiety detection during human-robot interaction. In *Intelligent Robots and Systems, 2005 IEEE/RSJ International Conference on*, pages 616–621, 2005.

[41] C.C. Wang, H.C. Hsu, and M.S. Zeng. Hand signal recognition for diver-machine interface. In *OCEANS '02 MTS/IEEE*, pages 1231–1236, 2002.

[42] G. Dudek, J. Sattar, and A. Xu. A visual language for robot control and programming: A human-interface study. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 2507–2513, 2007.

[43] A. Xu, G. Dudek, and J. Sattar. A natural gesture interface for operating robotic systems. In *Robotics and Automation, IEEE International Conference on*, pages 3557–3563, 2008.

[44] B. Verzijlenberg and M. Jenkin. Swimming with robots: Human robot communication at depth. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 4023–4028, 2010.

[45] P. Robinette, P.A. Vela, and A.M. Howard. Information propagation applied to robot-assisted evacuation. In *Robotics and Automation, IEEE International Conference on*, pages 856–861, 2012.

[46] L.T. Parker and A.M. Howard. Assistive formation maintenance for human-led multi-robot systems. In *Systems, Man and Cybernetics, IEEE International Conference on*, pages 2350–2355, 2009.

[47] D. Brooks and A.M. Howard. Quantifying upper-arm rehabilitation metrics for children through interaction with a humanoid robot. *Journal of Applied Bionics and Biomechanics*, 9(2):157–172, 2012.

[48] A.M. Howard, C.H. Park, and S. Remy. Using haptic and auditory interaction tools to engage students with visual impairments in robot programming activities. *IEEE Transactions on Learning Technologies*, 5(1):87–95, 2012.

[49] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

[50] W. Ren and R.W. Beard. *Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications*. Springer Publishing Company, Inc., 1st edition, 2007.

[51] P. Ogren, E. Fiorelli, and N.E. Leonard. Cooperative control of mobile sensor networks:adaptive gradient climbing in a distributed environment. *Automatic Control, IEEE Transactions on*, 49(8):1292–1302, 2004.

[52] M. Egerstedt and X. Hu. Formation constrained multi-agent control. *Robotics and Automation, IEEE Transactions on*, 17(6):947 –951, 2001.

[53] D.V. Dimarogonas, S.G. Loizou, K.J. Kyriakopoulos, and M.M. Zavlanos. A feedback stabilization and collision avoidance scheme for multiple independent non-point agents. *Automatica*, 42:229–243, 2006.

[54] Z. Lin, B. Francis, and M. Maggiore. Necessary and sufficient graphical conditions for formation control of unicycles. *Automatic Control, IEEE Transactions on*, 50(1):121 – 127, 2005.

[55] F. Fahimi. Full formation control for autonomous helicopter groups. *Robotica*, 26:143–156, 2008.

[56] M. Malisoff, F. Mazenc, and F. Zhang. Stability and robustness analysis for curve tracking control using input-to-state stability. *IEEE Transactions on Automatic Control*, 57(5):1320–1326, 2012.

[57] F. Zhang, E. Justh, and P. S. Krishnaprasad. Boundary following using gyroscopic control. In *Proceedings of 43rd IEEE Conference on Decision and Control*, pages 5204–5209. IEEE, 2004.

[58] W. Wu and F. Zhang. Cooperative exploration of level surfaces of three dimensional scalar fields. *Automatica, the IFAC Journall*, 47(9):2044–2051, 2011.

[59] M.F. Geen. Advances in Marine Survey Products and Platforms. In *OCEANS 2007*, pages 984–989, 2007.

[60] A.J. Hausler, R. Ghabcheloo, I. Kaminer, A.M. Pascoal, and A.P. Aguiar. Path planning for multiple marine vehicles. In *OCEANS 2009*, pages 423–431, 2009.

[61] K.Y. Pettersen and O. Egeland. Exponential stabilization of an underactuated surface vessel. In *Proceedings of the 35th IEEE Conference on Decision and Control*, pages 967–972, 1996.

[62] K.Y. Pettersen and E. Lefeber. Way-point tracking control of ships. In *Proceedings of the 40th IEEE Conference on Decision and Control*, pages 940–945, 2001.

[63] K.D. Do, Z.P. Jiang, and J. Pan. Universal controllers for stabilization and tracking of underactuated ships. *Systems & Control Letters*, 47(4):299–317, 2002.

[64] J. Ghommam, F. Mnif, A. Benali, and N. Derbel. Nonsingular Serret-Frenet Based Path Following Control for an Underactuated Surface Vessel. *Journal of Dynamic Systems, Measurement, and Control*, 131(2):021006, 2009.

[65] X. Xiang, L. Lapierre, C. Liu, and B. Jouvencel. Path tracking: Combined path following and trajectory tracking for autonomous underwater vehicles. In *Proceedings of the International Conference on Intelligent Robots and Systems*, pages 3558 –3563, 2011.

[66] K.Y. Pettersen and T.I. Fossen. Underactuated dynamic positioning of a ship-experimental results. *IEEE Transactions on Control Systems Technology*, 8(5):856–863, 2000.

[67] F. Zhang, D. M. Fratantoni, D. Paley, J. Lund, and N. E. Leonard. Control of coordinated patterns for ocean sampling. *International Journal of Control*, 80(7):1186–1199, 2007.

[68] F. Zhang, A. O'Connor, D. Luebke, and P.S. Krishnaprasad. Experimental study of curvature-based control laws for obstacle avoidance. In *Proceedings of 2004 IEEE International Conference on Robotics and Automation*, volume 4, pages 3849–3854, 2004.

[69] J. DeArruda. Oceanserver IVER2 autonomous underwater vehicle remote helm functionality. In *OCEANS 2010*, pages 1–5, 2010.

[70] T.I. Fossen. *Gaudiance and Control of Ocean Vehicles*. John Wiley and Sons, Chichester, 1 edition, 1994.

[71] G. Antonelli. *Underwater Robots-Motion and Force Control of Vehicle-Manipulator Systems*. Springer-Verlag, Berlin, 2 edition, 2006.

[72] C.J. Von Alt and J.F. Grassle. Leo-15 an unmanned long term environmental observatory. In *OCEANS 1992*, pages 849–854, 1992.

[73] S. Tang, T. Ura, T. Nakatani, B. Thornton, and T. Jiang. Estimation of the hydrodynamic coefficients of the complex-shaped autonomous underwater vehicle tuna-sand. *Journal of Marine Science and Technology*, 14(3):373–386, 2009.

[74] T. Nakatani, T. Ura, Y. Ito, J. Kojima, K. Tamura, T. Sakamaki, and Y. Nose. AUV "TUNA-SAND" and its exploration of hydrothermal vents at kagoshima bay. In *OCEANS 2008*, pages 1–5, 2008.

[75] T. Perez. *Ship Motion Control: Course Keeping and Roll Stabilisation Using Rudder and Fins*. Advances in Industrial Control. Springer, 2010.

[76] K. DeMarco, M.E. West, and T.R. Collins. An implementation of ROS on the yellowfin autonomous underwater vehicle (AUV). In *OCEANS 2011*, pages 1–7, 2011.

[77] A. Melim and M. West. Towards autonomous navigation with the yellowfin AUV. In *OCEANS 2011*, pages 1–5, 2011.

[78] A.P. Aguiar and João P.H. Trajectory-tracking and path-following of underactuated autonomous vehicles with parametric modeling uncertainty. *IEEE Transactions on Automatic Control*, 52(8):1362–1379, 2007.

[79] E. Borhaug and K.Y. Pettersen. Formation control of 6-dof euler-lagrange systems with restricted inter-vehicle communication. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 5718–5723, 2006.

[80] D.J. Klein, P. Lee, K.A. Morgansen, and T. Javidi. Integration of communication and control using discrete time kuramoto models for multivehicle coordination over broadcast networks. *IEEE Journal on Selected Areas in Communications*, 26(4):695 – 705, 2008.

[81] R. Ghabcheloo, A. P. Aguiar, A. Pascoal, C. Silvestre, I. Kaminer, and J. Hesphanha. Coordinated path-following control of multiple underactuated autonomous vehicles in the presence of communication failures. In *Proceedings of the 45th IEEE Conference on Decision and Control*, pages 4345–4350, 2006.

[82] J. Ghommam and F. Mnif. Coordinated path-following control for a group of underactuated surface vessels. *IEEE Transactions on Industrial Electronics*, 56(10):3951–3963, 2009.

[83] W. Dong and J. A. Farrell. Formation control of multiple underactuated surface vessels. *IET Control Theory and Applications*, 2(12):1077–1085, 2008.

[84] H. Yang and F. Zhang. Robust control of formation dynamics for autonomous underwater vehicles in horizontal plane. *Journal of Dynamic Systems, Measurement, and Control*, 134(3):031009, 2012.

[85] A. Abdessameud and A. Tayebi. Formaion control of vtol unmmaned aerial vehicles with communication delays. *Automatica*, 47(11):2383–2394, 2011.

[86] N. Chopraa, M.W. Spong, and R. Lozano. Synchronization of bilateral teleoperators with time delay. *Automatica*, 44(8):2142–2148, 2008.

[87] S. Chung, U. Ahsun, and J.E. Slotine. Application of synchronization to formation flying spacecraft: Lagrangian approach. *Journal of Guidance, Control, and Dynamics*, 32(2):512–526, 2009.

[88] K. Zhou and J.C. Doyle. *Essentials of Robust Control*. Prentice Hall, NJ,USA, 1 edition, 1997.

[89] K. Gu, V.L. Kharitonov, and J. Chen. *Stability of Time-Delay Systems*. Birkhauser, Boston, 2003.

[90] M.S. Mahmoud. *Robust Control and Filtering for Time-Delay Systems*. Marcel-Dekker, New York, 2000.

[91] F. Zhang, M. Goldgeier, and P.S. Krishnaprasad. Control of small formations using shape coordinates. In *Robotics and Automation, IEEE International Conference on*, pages 2510–2515, 2003.

[92] F. Zhang. Cooperative shape control of particle formations. In *Decision and Control, 46th IEEE Conference on*, pages 2516–2521, 2007.

[93] F. Zhang and N.E. Leonard. Cooperative filters and control for cooperative exploration. *Automatic Control, IEEE Transactions on*, 55(3):650–663, 2010.

[94] K.D. Do and J. Pan. Robust path following of underactuated ships using Serret-Frenet frame. In *Proceedings of the American Control Conference*, pages 2000–2005, 2003.

[95] X. Liang, W. Wu, D. Chang, and F. Zhang. Real-time modelling of tidal current for navigating underwater glider sensing networks. *Procedia Computer Science*, 10:1121–1126, 2012.

[96] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[97] C. Stachniss, C. Plagemann, and A.J. Lilienthal. Gas distribution modeling using sparse gaussian process mixtures. *Autonomous Robots*, 26(2-3):187–202, 2009.