MODELING, DESIGN AND CONTROL OF GLIDING ROBOTIC FISH

By

Feitian Zhang

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Electrical Engineering – Doctor of Philosophy

2014

ABSTRACT

MODELING, DESIGN AND CONTROL OF GLIDING ROBOTIC FISH

By

Feitian Zhang

Autonomous underwater robots have been studied by researchers for the past half century. In particular, for the past two decades, due to the increasing demand for environmental sustainability, significant attention has been paid to aquatic environmental monitoring using autonomous underwater robots. In this dissertation, a new type of underwater robots, gliding robotic fish, is proposed for mobile sensing in versatile aquatic environments. Such a robot combines buoyancy-driven gliding and fin-actuated swimming, inspired by underwater gliders and robotic fish, to realize both energy-efficient locomotion and high maneuverability. Two prototypes, a preliminary miniature underwater glider and a fully functioning gliding robotic fish, are presented. The actuation system and the sensing system are introduced. Dynamic model of a gliding robotic fish is derived by integrating the dynamics of miniature underwater glider and the influence of an actively-controlled tail. Hydrodynamic model is established where hydrodynamic forces and moments are dependent on the angle of attack and the sideslip angle. Using the technique of computational fluid dynamics (CFD) water-tunnel simulation is carried out for evaluating the hydrodynamic coefficients. Scaling analysis is provided to shed light on the dimension design.

Two operational modes of gliding robotic fish, steady gliding in the sagittal plane and tail-enabled spiraling in the three-dimensional space, are discussed. Steady-state equations for both motions are derived and solved numerically. In particular, for spiral motion, recursive Newton's method is adopted and the region of convergence for this method is numerically examined. The local asymptotic stability of the computed equilibria is established through checking the Jacobian

matrix, and the basins of attraction are further numerically explored. Simulation and experiments are conducted to validate steady-state models and calculated equilibria for both motions.

Tail-enabled feedback control strategies are studied in both sagittal-plane glide stabilization and three-dimensional heading maintenance. A passivity-based controller and a sliding mode controller are designed and tested in experiments for those two problems, respectively. In sagittal-plane glide stabilization, a nonlinear observer is designed and implemented to estimate velocity-related states. A three-dimensional curve tracking problem is also discussed and a two-degree-of-freedom control scheme is proposed by integrating static inverse mapping and H_∞ control technique. The differential geometric features, such as the torsion and curvature, are explored for planning the trajectory.

Finally, the field tests with the lab-developed prototype of gliding robotic fish are conducted in the Kalamazoo River, Michigan and the Wintergreen Lake, Michigan for detecting oil spill and sampling harmful algal blooms, respectively. Both gliding and spiraling motions are tested in the experiments as well as the fish-like swimming. The field test results are presented to show the effectiveness of the designed robot in environmental monitoring tasks.

Dedicated to
  my wife Mi Zhou
   and our son
     Andrew Anbang Zhang
       with all my love.

ACKNOWLEDGMENTS

First, I would like to express my warmest gratitude to my advisor, the director of Smart Microsystems Lab at Michigan State University, Prof. Xiaobo Tan. Five years ago, Prof. Tan provided me a valuable opportunity of starting my Ph.D. journey here in the field of underwater robotics, and now upon the completion of my Ph.D. training, with experience and knowledge gained during the past five years, I am excitedly looking forward to my postdoctoral research life at University of Maryland. I genuinely thank him for his advising in my research and mentoring in my career path.

I want to thank my academic committee members, Prof. Hassan Khalil, Prof. Guoming Zhu and Prof. Mukherjee at Michigan State University, and our research collaborator, Prof. Fumin Zhang at Georgia Institute of Technology, for their insightful comments in the area of control and robotics. I also thank Prof. Elena Litchman at Michigan State University and Prof. Derek Paley at University of Maryland for their generosity and kindness in providing experimental facilities and testing fields for the project.

In Smart Microsystems Lab, I owe many people a debt of gratitude. The research in gliding robotic fish needs diverse expertise and extensive collaboration. I want to thank John Thon and Cody Thon for their continuous contribution in mechanical design and robot manufacturing, and Jianxun Wang and Hong Lei for their help in circuit development, and Bin Tian, Osama En-Nasr, Scott O'Connor, Suriya Madhan, Tingyuan Zhang and Yujie Hao for their help in conducting experiments and collecting data.

I would also like to thank the other students in Smart Microsystems Lab and the faculty, students and post-docs in my department, whose comments and discussions have helped my research. Thanks also to all of the administrators and staff members in the ECE department who have helped me over the years.

Most of all, I want to thank my mother, Dongmei He, my father, Guangbo Zhang and my wife, Mi Zhou for their unconditional love in both good times and bad times. I could not have finished my Ph.D. program in the past five years without their understanding, support and encouragement.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Gliding robotic fish, a new type of underwater robots, is proposed for monitoring aquatic environment in this dissertation. Such a robot combines the advantages of both underwater gliders and robotic fish, and features long operation duration and high maneuverability.

In this introduction, I will first discuss the existing methods in aquatic environmental monitoring, and conduct a brief literature review on two important underwater robots, the underwater glider and robotic fish. Afterwards, gliding robotic fish is introduced as a new type of underwater robots, inspired by the above two well-known classes of robots. Locomotion mechanism and advantages of the gliding robotic fish are briefly discussed. The structure of this dissertation is then clarified. At last, an overview of the contributions is presented.

1.1   Technology in Aquatic Environmental Monitoring

There is a growing interest in monitoring aquatic environments, due to the emerging problems of environmental pollution and expanding demand for sustainable development. Such pollutions involve various types contaminants, including industrial waste, chemicals, and bacteria, etc. The pollution could happen in different aquatic environments, such as ponds, rivers, lakes, and even the ocean. Due to ongoing industrialization and expanding exploration of aquatic resources, water pollution problems have become increasingly frequent and severe, which has drawn a global attention [5–7]. In particular, the massive 2010 oil spill in the Gulf of Mexico has brought world-wide

1

attention and attracted intensive research into this urgent issue. The cleanup and recovery will take many years and millions of dollars. Throughout this process, monitoring the water quality and detecting the remaining contaminants in the water are an essential task [8, 9]. Similar contamination incidents include the 2010 oil spill in Kalamazoo River, Michigan, [10], 2010 Xingang Port oil spill in China Yellow Sea [11], 2011 Nigeria oil spill [12], and 2012 Arthur Kill storage tank spill in New Jersey [13].

Throughout the development in technology for aquatic environmental monitoring, a variety of different water sampling methods have been studied and employed. Manual sampling, via boat/-ship or with handheld devices, is still a common practice in aquatic environmental monitoring. This approach is labor-intensive and has difficulty capturing dynamic and spatially distributed phenomena of interest. An alternative is in-situ sensing with fixed or buoyed/moored sensors [14], including vertical profilers that can move up and down along a water column [15–19]. However, since buoyed sensors have little or no freedom to move around, it could take a prohibitive number of them to capture spatially inhomogeneous information. The past decade or so has seen great progress in the use of robotic technology in aquatic environmental sensing [20–36]. Predominant examples of these technologies include remotely operated vehicles (ROVs) [20, 24, 27, 33], autonomous surface vehicles (ASVs) [22, 23, 25, 31, 35], propeller-powered autonomous underwater vehicles (AUVs) [24, 30, 34, 36–39], and underwater gliders [1–3]. ROVs typically have limited spatial access and autonomy due to their tethered nature, while the sampling space of ASVs is limited to the two-dimensional (2D) water surface. AUVs, on the other hand, can operate freely and autonomously in the 3D water body, but their high price tags (upward of $150K per vehicle) presents a huge barrier to their deployment in large numbers for high-resolution spatiotemporal coverage. Besides, there has been a growing interest in bio-inspired underwater robots [40–46], such as robotic fish, which holds great potential in wide application in water monitoring. The

shortcoming of this kind of robots is the limited operational duration and thus area coverage, due to the energy constraints and the requirement of constant actuation for propulsion..

1.2 Gliding Robotic Fish

1.2.1 Design Concept

The design concept of a gliding robotic fish comes from energy-efficient underwater gliders and highly-maneuverable robotic fish.

Underwater seagliders are known for their great energy-efficiency and long-duration operation in oceanographic applications [47]. An underwater glider utilizes its buoyancy and gravity to enable motion without any additional propulsion, and adjusts its center of gravity to achieve certain attitude, which results in glide and thus horizontal travel. Since energy is needed only for buoyancy and center-of-gravity adjustment when switching the glide profile, underwater gliders are very energy-efficient, as proven by the great success of the Seaglider [1], Spray [2] and Slocum [3] (Fig. 1.1). The maneuverability of underwater gliders, however, is quite limited. The large size (1– 2 m long), heavy weight (50 kg and above), and high cost [47] of these vehicles also impede their adoption in the application of networked sensing and in versatile environments like ponds and inland lakes.

On the other hand, over the past two decades, there has been significant interest in developing robots that propel and maneuver themselves like real fish do. Often called *robotic fish* (Fig. 1.2), they accomplish swimming by deforming the body and fin-like appendages [48–68]. In many designs, a fish-like flapping tail is used to provide propulsion force and a biased tail angle is applied to realize turning. Due to the similarity to the real fish and the fact that the rotation of the tail fin is usually enabled by a motor that is easy to control and fast in response, robotic fish typically

3
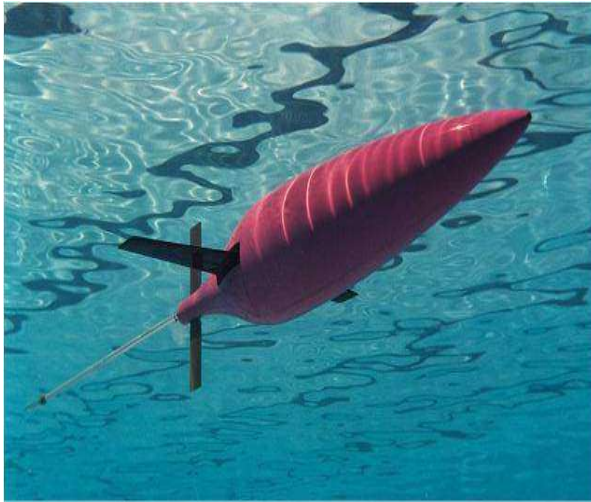
(a)   (b)

(c)

Figure 1.1: Classic underwater gliders. (a) Seaglider [1]; (b) Spray [2]; (c) Slocum [3].

(a)
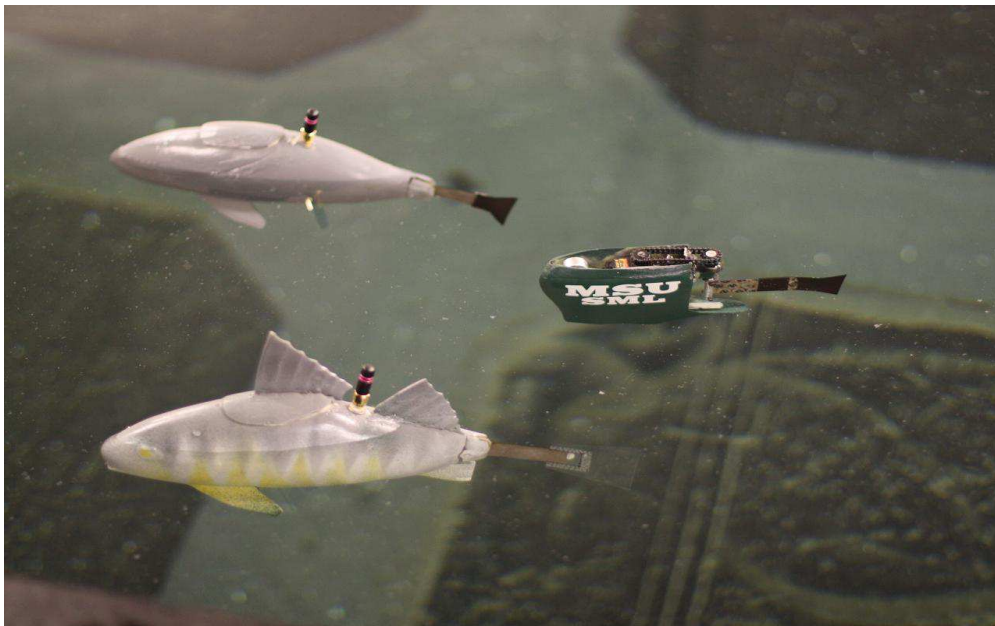
(b)

Figure 1.2: Examples of robotic fish. (a) From Mechatronics Research laboratory at Massachusetts Institute of Technology [4]; (b) From Smart Microsystems Lab at Michigan State University.

have high maneuverability (e.g., small turning radius); However, as the forward propelling force is generated from the flapping motion of the fins, such robots require constant actuation for swimming and cannot work for extended periods of time without battery recharge.

In this dissertation, inspired by the design and merits of both underwater gliders and robotic fish, a new type of underwater robots, *gliding robotic fish*, is proposed. Gliding robotic fish combines both mechanisms of gliding and swimming and features energy efficiency and high maneuverability at the same time. Such a robot would realize most of its locomotion through gliding like underwater gliders, by utilizing its buoyancy and gravity to enable motion without any additional propulsion, and adjusting its center of gravity to achieve a certain attitude. It would use actively controlled fins to achieve high maneuverability, during turning and orientation maintenance. Of course, fins can also provide additional propulsive power during locomotion, if needed.

The dimension of the gliding robotic fish is supposed to be smaller than traditional underwater gliders, and one use of such a robot is to provide a mobile sensing platform in relatively shallow waters, such as lakes, rivers, and even ponds, where larger gliders are not quite suitable due to their large size and high cost. The small size and low cost of gliding robotic fish also facilitate the research of networked sensing and operative control.

1.2.2 Motion and Control

For gliding robotic fish, there could be various interesting motions generated by integrating the gliding and swimming mechanisms. In this dissertation, two main steady motion profiles are discussed as the regular working patterns for sampling water. One is steady-state gliding in the sagittal plane, and the other is steady-state spiraling in the three-dimensional (3D) space.

The steady-state gliding is also the common operating mode for traditional underwater gliders, used for sampling the water field 1.3. In the zig-zag trajectory, the gliding robotic fish only

6
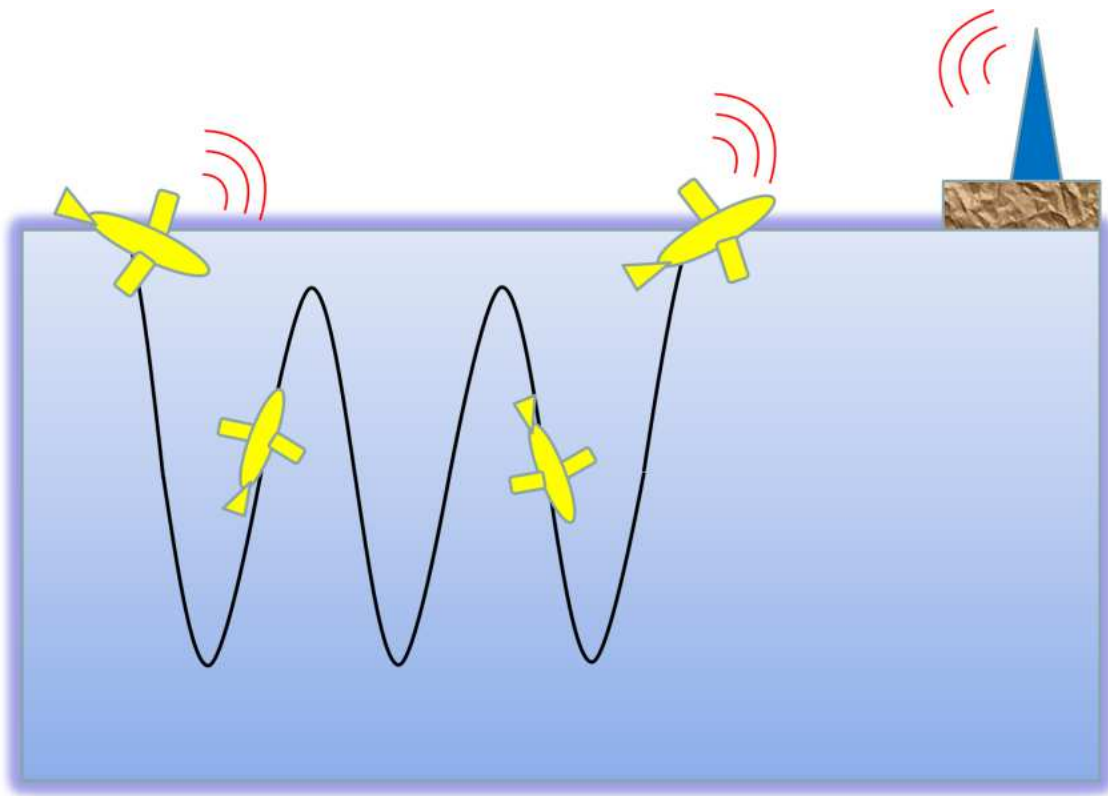
Figure 1.3: Schematics of working pattern "gliding in sagittal plane".
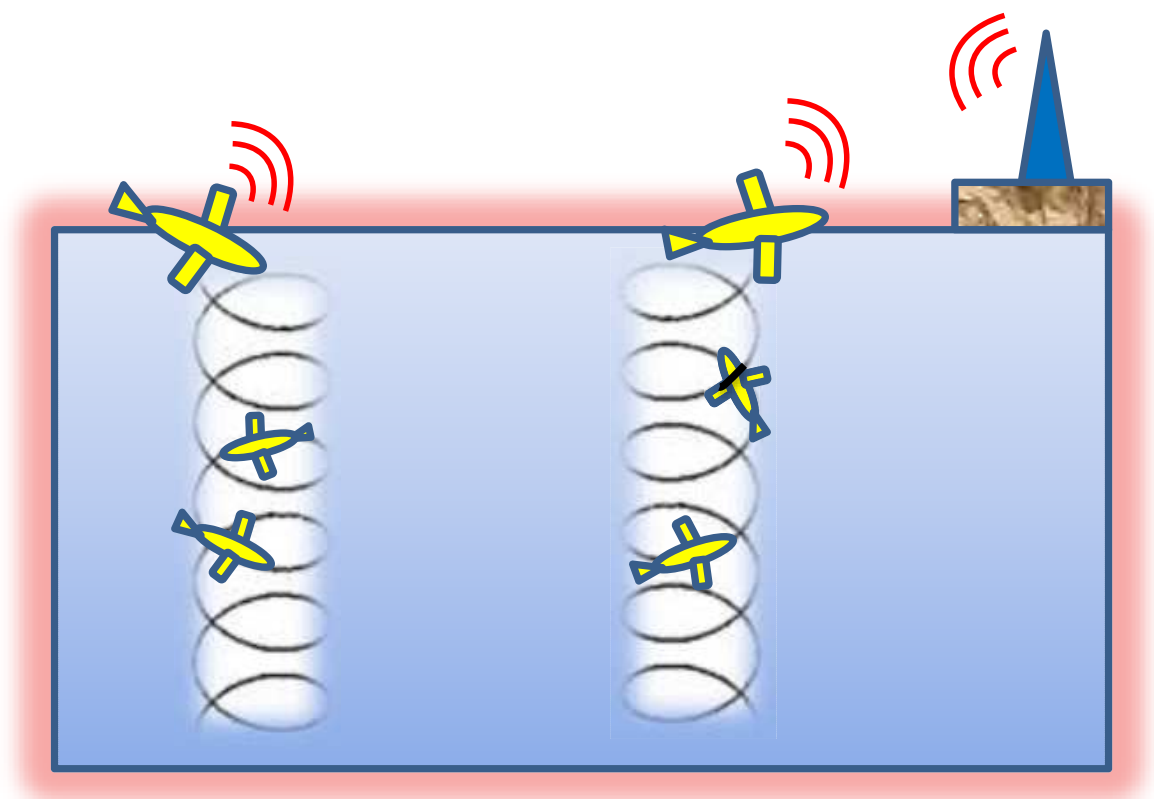
Figure 1.4: Schematics of working pattern "spiraling in 3D space".

consumes energy during the transition between descending and ascending. In the phase of steady glide, balances of forces and moments are achieved. If we interpret the motion intuitively, the net buoyancy of the robot enables a propelling force to counteract the hydrodynamic resistance. Zero energy consumption in the steady gliding period makes the gliding robotic fish highly energy-efficient.

The other motion, steady spiral in the three-dimentional space, is proposed as a novel method of sampling a water column for gliding robotic fish 1.4. A water column is a conceptual narrow volume (like a narrow cylinder) of water stretching vertically from the surface to the bottom. Water column sampling is a routine surveying method in environmental studies to evaluate the stratification or mixing of water layers [69]. This motion is achieved by incorporating the steady gliding with a deflected fish-like tail. The non-zero angled tail fin will introduce a turning moment to the steady gliding robot and lead to a 3D spiraling motion. With the actively-controlled tail, the gliding robotic fish is capable of spiral motions with tight turning radius. If needed, a gradually changing tail angle will form spiral-in or spiral-out 3D trajectories.

For most of its operation time, a gliding robotic fish holds a preset steady gliding path without any energy consumption. The gliding angle is calculated before deployment so that the robot will follow a designed trajectory, which either minimizes the energy cost, or maximizes the field mapping capability [70]. However, a gliding robot fish is subject to many non-negligible uncertainties from the aquatic environment (e.g. current disturbance), which results in additional energy cost because counteracting the deviation from the preset course requires re-calibration and control effort to keep the robot on or bring it back to the designed path [71] [29] [72]. The stability property of the steady gliding path and fast convergence to the path, are very important for the gliding robotic fish to reduce the energy expenditure on path correction. The stabilization involves both sagittal and lateral dynamics, corresponding to sagittal and lateral disturbances.

For the gliding angle stabilization in the sagittal plane, there are several related results in literature for underwater gliders. Most of them use the net buoyancy and internal mass displacement as control variables. For instance, in [73] and [74] the authors presented an LQR (linear quadratic regulator) controller and a PID controller, respectively, but both based on a linearized model. Although in the LQR method energy is used as a cost function, the approach does not consider the additional cost for the course correction and guidance. Both LQR and PID controller can not fully address the nonlinear gliding characteristics. Nonlinear controllers involving torque control and buoyancy control are proposed by Bhatta [75], but those methods ignore the dynamics of the control surface and also require full state feedback for the controller implementation, which will increase the complexity of the software and hardware. A Lyapunov-based control design is reported in [76]; however, for the elevator control, it only deals with a fixed-value control input to achieve a certain equilibrium gliding path. In this dissertation, a novel, passivity-based nonlinear controller is proposed for the sagittal-plane stabilization problem using only a whale-like tail fin of the gliding robotic fish. The singular perturbation results from [76] are utilized to reduce the full-order system to a slow-mode second-order system. A passivity-based controller is designed based on the approximated reduced system, and the controller is applied back into the original full-order system. Through checking the Jacobian matrix, the local stability of the full-order closed-loop system is established given sufficient time-scale separation. Furthermore, a nonlinear observer is also proposed to estimate velocity-related system states, which are used in controller implementation. Both open-loop and closed-loop experiments are carried out to illustrate the effectiveness of the designed controller and observer.

For the lateral motion, attitude stabilization is of great importance to underwater robots, which are subjected to various environmental disturbances. There has been extensive work on pitch motion stabilization in the longitudinal plane for underwater gliders as discussed in the previous

10

paragraph, where typically the absence of lateral motion is assumed [73, 76–78]. There has also been some limited research into three-dimensional gliding involving lateral motion, most of which focuses on the steady-state turning or spiraling [79–81]. However, little work has been reported on the yaw angle stabilization where ambient flow disturbances could easily push the robot away from its desired heading orientation. In this dissertation, the relative degree of the system dynamics is first identified and error dynamics is then derived based on the system normal form. A sliding mode controller is designed for the error dynamics to achieve attitude stabilization. Both simulation and experiments are conducted to evaluate the effectiveness of the proposed control scheme.

Three-dimensional curve tracking is also crucial for underwater robots, including in particular, gliding robotic fish. For example, it is critical in sampling water columns, seeking pollutant sources, and mapping the whole aquatic environment. Precise tracking control is very challenging due to dynamic nonlinearity and strong coupling among multiple control inputs. As stated previously, there has been some limited literature that covers three-dimensional gliding involving lateral motion, most of which focuses on the steady-state turning or spiraling [79–81]. However, it is still an unexplored area of three-dimensional curve tracking control for buoyancy-driven underwater robots. In this dissertation, I propose a novel two degree-of-freedom (DOF) control strategy for gliding robotic fish to track three-dimensional curves based on the differential geometric features of steady spirals. The control strategy includes a feedforward controller that is designed through inverse mapping of steady spiral motion, and a feedback controller designed using the robust H_∞ framework based on local, linearized dynamics. The effectiveness of the proposed 2-DOF control scheme is demonstrated in simulation, where PI control and open-loop inverse mapping control are also conducted for the purpose of comparison.

There are a number of challenges in aquatic environmental monitoring, especially in field experiments using autonomous underwater robots. Gliding robotic fish is proposed to carry out the

11

water sampling work with great energy efficiency, high maneuverability and adaptability to versatile environments. As the final test of the design concept and robot development, the gliding robotic fish prototype is taken to Kalamazoo River, Michigan, to detect oil spill, and Wintergreen Lake, Michigan, to sample harmful algal blooms. The external swappable environmental sensor reads the field data of interest and the on-board micro-controller sends back the information wirelessly to the base station. The GPS-based positioning system facilitates the water sampling task with precise location information, where the motions of swimming, gliding and spiraling are tested and integrated. The field test results are presented to demonstrate the functionality and usefulness of the proposed gliding robotic fish as a novel platform for solving the real-world problem of aquatic environmental monitoring.

1.3　Overview of Contributions

The contributions of this research reside mainly on dynamics analysis and tail-enabled control of a new type of underwater robot, the gliding robotic fish. The details are as follows.

First, the design idea of combining buoyancy-driven propulsion and tail-enabled maneuverability is novel. The advantages of long operation duration and high maneuverability from underwater gliders and robotic fish, make the gliding robotic fish a suitable underwater platform in shallow water environmental sensing.

Second, the discussion about the two water-sampling working patterns, the sagittal-plane glide and three-dimensional spiral, provides insights of the energy-efficient feature of gliding robotic fish. Particularly, the tail-enabled steady spiral, is a novel motion proposed for sampling water columns, and potential path planning tasks. The turning radius could be as tight as 0.5 m, providing much higher sampling resolution compared to the 30-40 m radius from traditional underwater

gliders.

Third, the steady glide stabilization is usually realized through buoyancy control or mass-distribution control in literature. In this dissertation an actively-control tail fin is adopted as the control input to stabilize the steady gliding motion. For the first time in literature of the sagittal-plane motion of buoyancy-driven underwater robots, a passivity-based feedback controller is designed to obtain a fast convergence speed with partial state feedback, which is convenient for implementation. For the heading maintenance in lateral dynamics, a sliding mode controller is proposed with only a requirement on the yaw angle feedback information. Moreover, both simulation and experimental results are presented to provide insight into the stabilization problem for buoyancy-driven underwater robots.

Fourth, three-dimensional curve tracking for buoyancy-driven underwater robots is very challenging and little relevant work could be found. In this dissertation, A novel two-degree-of-freedom control strategy, which consists of a feedforward controller designed through inverse mapping of steady spiral motion, and a feedback controller designed using the robust H_∞ framework based on local, linearized dynamics, is proposed for the curve tracking problem using the differential geometric features of steady spirals. The study of the geometric features of the spiral motion, such as torsion and curvature, also shows a promising motion planning and navigation method.

Finally, the successful development of gliding robotic fish prototype and the field tests in the Kalamazoo River and the Wintergreen Lake, proves the concept of the design, and demonstrates the functions of the robot as a platform for aquatic environmental monitoring. This leads to more research directions, such as adaptive sampling, networked control and cooperative control with multiple agents, and it opens the door to further collaboration with biologists and sociologists to solve bigger real-world water-pollution problems.

Chapter 2

Implementation of Gliding Robotic Fish

During the research of the gliding robotic fish, two prototypes have been developed. The first one is a miniature underwater glider, reported in [82, 83], built to study the gliding component of gliding robotic fish and help to evaluate the steady glide model in the sagittal plane (Fig. 2.1). The second one is a fully functioning gliding robotic fish prototype, named "Grace" (Fig. 2.2) [84]. In this section, I will focus on the implementation of the gliding robotic fish "Grace", which naturally explains the case for the miniature underwater glider.

2.1 Actuation System

Integrating an actively-controlled tail into the miniature underwater glider, the research team has developed a fully functioning gliding robotic fish, named "Grace". The robot has three actuation systems for locomotion, including the buoyancy system, the mass distribution system, and the actively-controlled tail fin system.

In the buoyancy system, water is pumped in and out of the robot's body to change the net buoyancy. When the robot is heavier than the water it displaces (negatively buoyant), the robot will descend (Fig. 2.2); and when it is lighter than the water it displaces (positively buoyant), the robot will ascend. The pumping system of "Grace" is enabled by a linear actuator with integrated feedback, which allows the precise control of water volume despite the pressure differences at different depths, while a DC pump is used for the miniature underwater glider prototype.

14
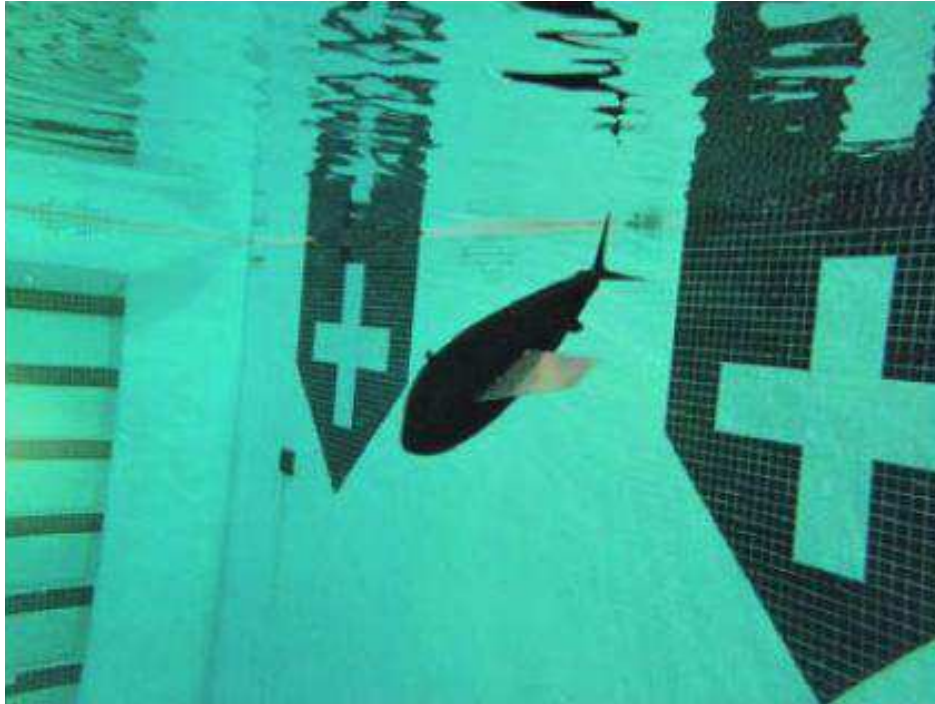
Figure 2.1: Protoype I, the miniature underwater glider operating in the swimming pool.
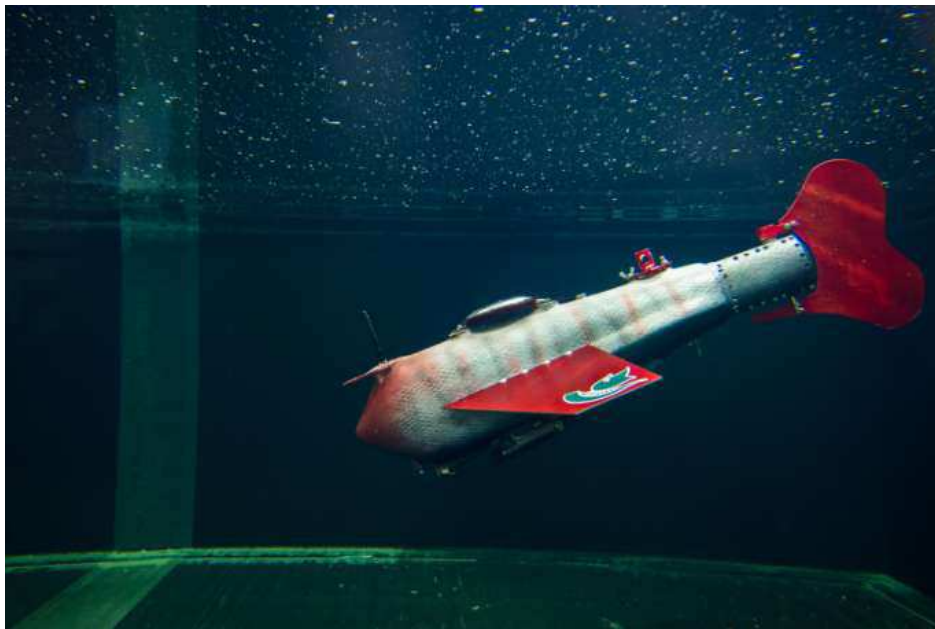
Figure 2.2: Gliding robotic fish "Grace" gliding in the large indoor tank in the Smart Microsystems Lab.

Figure 2.3: Gliding Rototic fish "Grace" swimming in Kalamazoo river.

For the mass distribution system, a linear actuator is used to push a mass (battery pack) back and forth along a guiding rail to change the center of the mass, for the purpose of manipulating the pitch angle, in both miniature underwater glider and "Grace".

The fish-like tail fin system in "Grace" is driven by a servo motor through a chain transmission. In three-dimensional gliding, a deflected tail can be used to control the turning motion and heading orientation. Like a real fish, the robot can also flap the tail to realize the swimming motion (Fig. 2.3).

2.2   Gliding Robotic Fish Components

A schematic of selected components for the gliding robotic fish is shown in Fig. 2.4. In the figure, the components of the three actuation systems can be identified. Besides, there are two physically separated printed circuit boards (PCBs). One is the *ctrl PCB* containing the micro-controller and navigational sensors, such as gyroscopes, accelerometers and a digital compass; the other is the

16
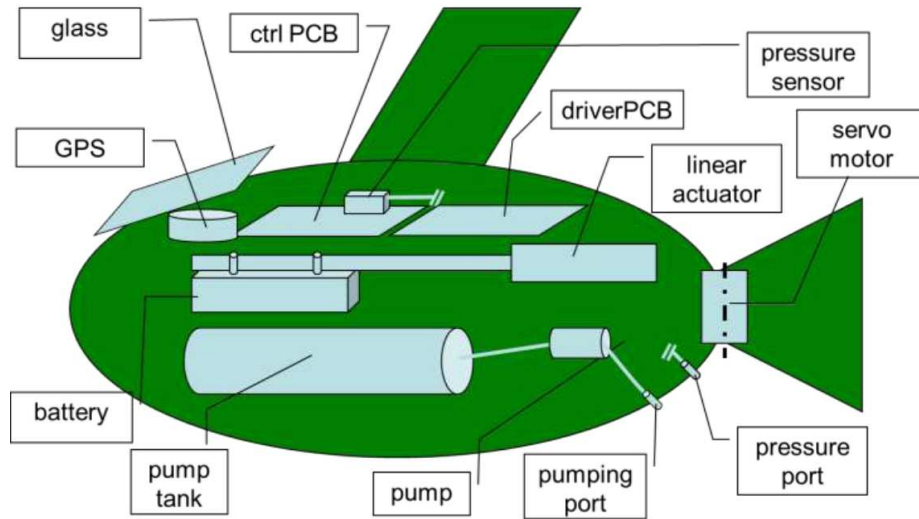
Figure 2.4: The schematic of internal configuration for gliding robotic fish.

driver PCB containing regulators and driver components for the actuators, including the linear actuators and the servo motor. A pressure sensor is used to measure the depth of the current location of the robot, with one port connecting to the ambient water.

There are some other components equipped on the shell of the gliding robotic fish, including the wireless communication antenna, environmental sensors and the GPS receiver, as shown in Fig. 2.5. "Grace" is equipped with a crude oil sensor and a temperature sensor, and has been tested in the Kalamazoo River, Michigan, to sample the oil concentration near the site of a 2010 oil spill (Fig. 2.3). The sensor can be easily swapped to measure other environmental processes, such as chlorophyll, harmful algae, turbidity, rhodamine. The GPS unit is used to measure the global position and provide the universal time, when the gliding robotic fish surfaces. Table 2.1 lists the details of the used components mentioned above.
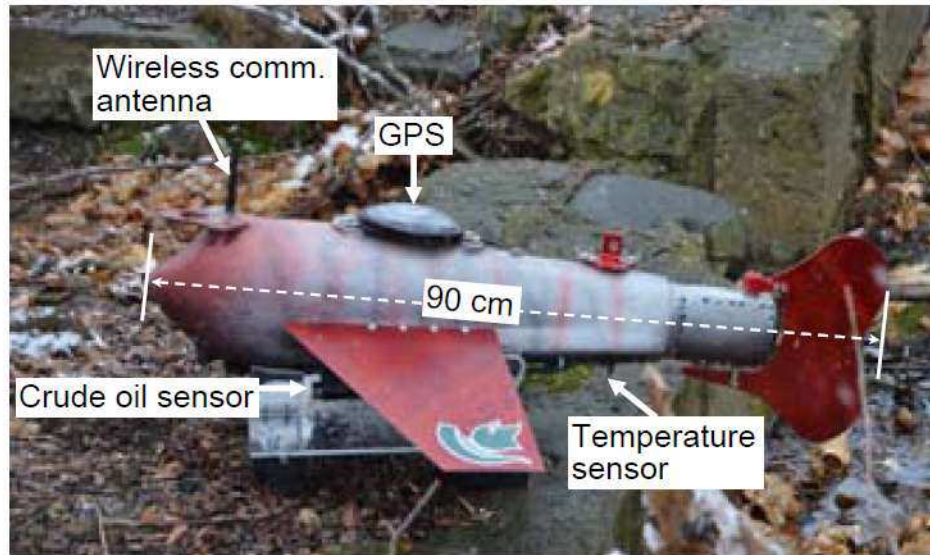
Figure 2.5: Components on the outside of prototype "Grace".

Table 2.1: Selected components used in "Grace".

	Component name	Component model
1	Micro-controller	Microchip dsPIC6014A
2	Battery	Batteryspace 18.5V Polymer-Li-Ion battery pack
3	Linear actuator	Firgelli L16-140-63-12-P
4	Pump 1 (miniature underwater glider)	Flight Works Model 300C
5	Pump 2 ("Grace")	Servocity 180 lbs thrust linear actuator
6	Servo motor	Hitec Servo HS-7980TH
7	Pressure sensor	Honeywell 40PC100G2A
8	GPS	Garmin GPS 18x LVC
9	Gyro	ST LPY503AL
10	Accelerometer+Compass	ST LSM303DLH
11	Wireless module	XBee Pro 900 XSC RPSMA
12	Wireless antenna	900MHz Duck Antenna RP-SMA
13	Crude oil sensor	Turner Designs Cyclops-7 Crude Oil Sensor
14	Temperature sensor	TMP36

2.3 Mechanical Design

The mechanical structure of the robot is designed considering the requirements of compactness, low cost, and energy efficiency, and the limitation of having to house all necessary electrical components. The outer shell is designed to have a fish-like shape with a streamline profile for the purpose of energy efficiency. The pair of wings is designed to provide enough hydrodynamic forces and moments for a satisfactory glide. A guide rail is designed to help push the battery pack using the linear actuator to change the mass distribution.

SolidWorks is used to draw schematics of most of the mechanical components. A computer numerical control (CNC) machine is used to manufacture parts of the system, such as the mold of the shell. 3D printer is used to print other parts of the system, such as the tail compartment.

The robot shell is made of carbon-fiber, which is strong enough to sustain the underwater pressure. The tail compartment is made of composite polymer via 3D printing, which enables rapid prototyping. The guide rail is made of stainless steel, which is rigid enough to provide a straight traveling path for the battery pack.

Some physical parameters of "Grace" are as the follows. Length is 65 cm (body) / 90 cm (total); width is 15 cm (body) / 75 cm (with wings); and height is 18 cm (body) / 34 cm (including tail). Here, the term "total" includes the body, the tail, the antenna and the environmental sensor. The wings are in a trapezoidal shape with a wingspan of 30 cm (one side) and an aspect ratio of 1.45. The weight is 9 kg in total. The tail with the servo compartment weighs 0.8 kg itself.

The tail is special in the design in that it is the only moving part seen from the outside, so it must be designed properly to be waterproof, robust, and reliable with respect to different flapping amplitude and frequency. The tail flapping motion is transformed from the rotation of a servo by a chain system, in which two identical gears are used as shown in Fig. 2.6.

Figure 2.6: Components of the tail compartment for "Grace".
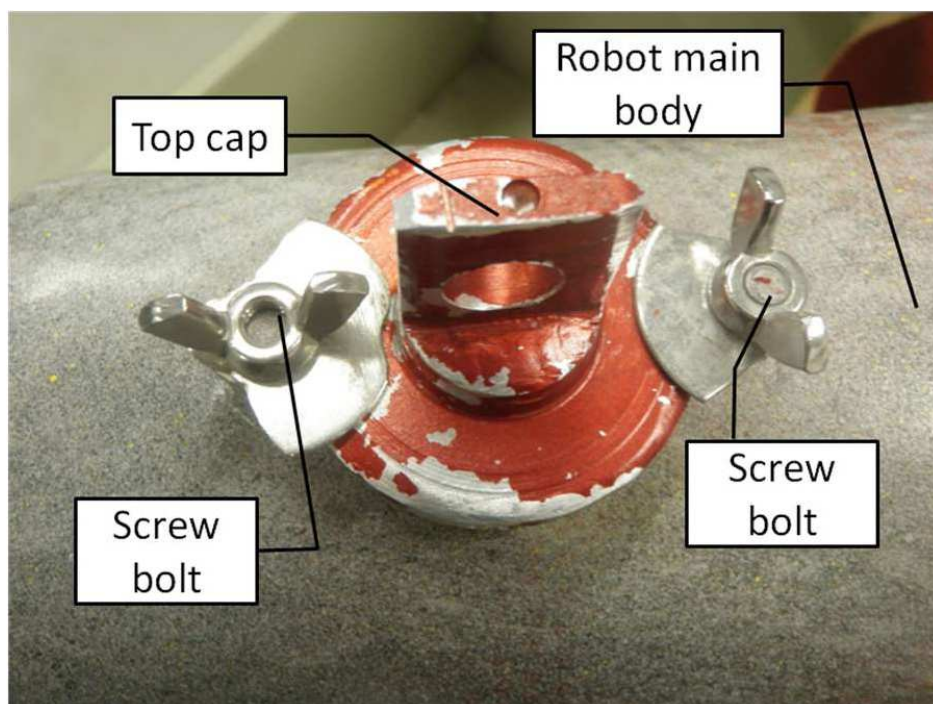
Figure 2.7: Top cap interface of Grace to the outside when cap covered.
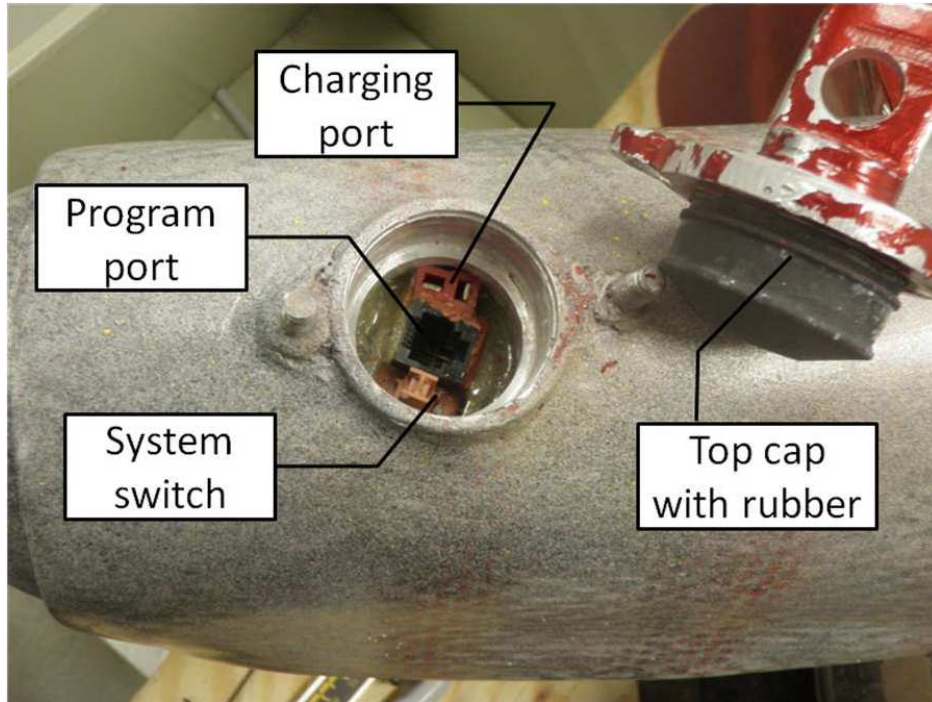
Figure 2.8: Top cap interface of Grace to the outside when cap open.

The robot must be waterproofed for all electrical/electronic components. In "Grace", all electrical/electronic components are placed in the main body. In particular, the system switch, charging port, and programming port are located under the top cap. A plunger-like cap covered by rubber is used to seal the top interface, with two screw bolts to ensure the sealing, as shown in Figs. 2.7 and 2.8.

In addition, the tail is detachable from the main body, enabling us to replace the tail if something goes wrong. The detachable tail enforces the reliability of the sealing of the main body with double O-rings. Furthermore, the tail can be configured to flap sideways (like a shark) or up and down (like a whale).

Chapter 3

Dynamic Model of Gliding Robotic Fish

3.1   Full Dynamic Model

A gliding robotic fish is a hybrid of a miniature underwater glider and a robotic fish, and its modeling will need to incorporate the effects of both. The tail of gliding robotic fish is used to control the lateral motion for underwater gliding, and used to propel the robot for swimming. We treat it as a control surface and a source for external forces and moments. The robot is modeled as a rigid-body system, including an internal movable mass for center-of-mass control and a water tank for buoyancy adjustment [73, 82, 83]. On the other hand, the deflected tail provides external thrust force and side force as well as the yaw moment.

Fig. 3.1 shows the mass distribution within the robot. The stationary body mass m_s (excluding the movable mass) has three components: hull mass m_h (assumed to be uniformly distributed), point mass m_w accounting for nonuniform hull mass distribution with displacement $\boldsymbol{r_w}$ with respect to the geometry center (GC), and ballast mass m_b (water in the tank) at the GC, which is a reasonable simplification since the effect on the center of gravity caused by the water in the tank is negligible compared with the effect from the movable mass. The movable mass $\bar{m}$, which is located at $\boldsymbol{r_p}$ with respect to the GC, provides a moment to the robot. The motion of the movable mass is restricted to the longitudinal axis. The robot hull displaces a volume of fluid of mass m . Let $m_0 = m_s + \bar{m} - m$ represents the excess mass (negative net buoyancy). The robot will sink if $m_0 > 0$ and ascend if $m_0 < 0$.
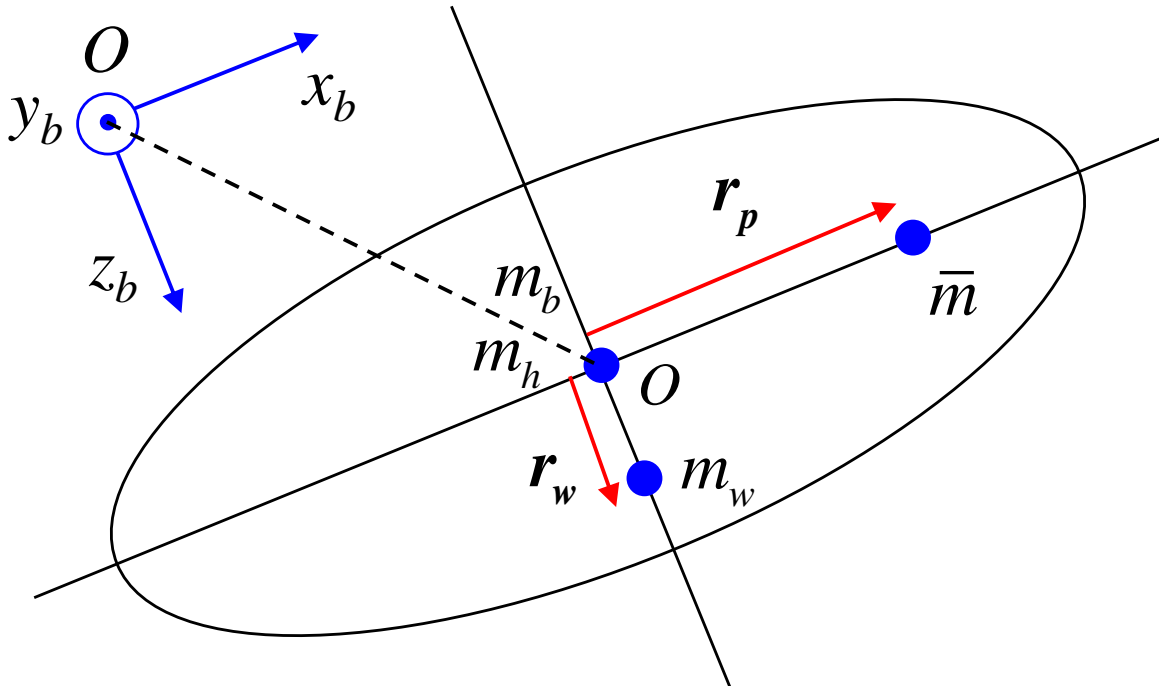
Figure 3.1: The mass distribution of the gliding robotic fish (side view).
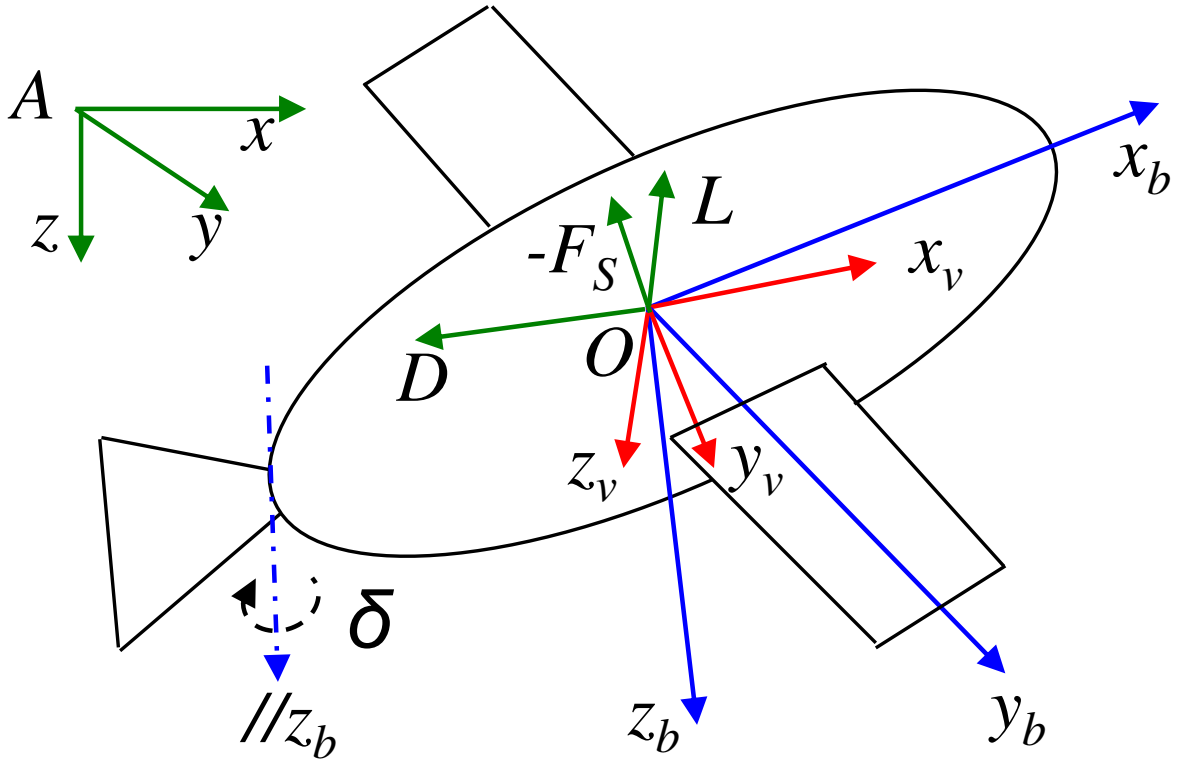
Figure 3.2: Illustration of the reference frame and hydrodynamic forces.

23

The relevant coordinate reference frames are defined following the standard convention. The body-fixed reference frame, denoted as $Ox_by_bz_b$ and shown in Fig. 3.2, has its origin O at the geometry center, so the origin will be the point of application for the buoyancy force. The Ox_b axis is along the body's longitudinal axis pointing to the head; the Oz_b axis is perpendicular to Ox_b axis in the sagittal plane of the robot pointing downward, and Oy_b axis is automatically formed by the right-hand orthonormal principle. In the inertial frame $Axyz$, Az axis is along gravity direction, and Ax/Ay are defined in the horizontal plane, while the origin A is a fixed point in space.

As commonly used in the literature, $\boldsymbol{R}$ represents the rotation matrix from the body-fixed reference frame to the inertial frame. $\boldsymbol{R}$ is parameterized by three Euler angles: the roll angle ϕ, the pitch angle θ, and the yaw angle ψ. Here

$$\boldsymbol{R} = \begin{pmatrix} c\theta c\phi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{pmatrix} \tag{3.1}$$

where $s(\cdot)$ is short for $\sin(\cdot)$ and c for $\cos(\cdot)$. Let $\boldsymbol{v_b} = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^T$ and $\boldsymbol{\omega_b} = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 \end{bmatrix}^T$ represent the translational velocity and angular velocity, respectively, expressed in the body-fixed frame. The subscript b indicates that the vector is expressed in the body-fixed frame, and this notation is applied throughout this dissertation.

We assume that the tail fin is rigid and pivots at the junction between the body and the tail about the Oz_b axis. The tail induces an external thrust force $\boldsymbol{F_t}$ on the robot when it flaps. There are also other hydrodynamic forces and moments generated because of the relative movement between the tail and the surrounding water, like the side force and the yaw moment.

By extending the previous modeling work for underwater gliders [81, 82], we obtain the dy-

namic model for a gliding robotic fish with an actively-controlled tail fin as the following,

$$\dot{\boldsymbol{b_i}} = \boldsymbol{R}\boldsymbol{v_b} \tag{3.2}$$

$$\dot{\boldsymbol{R}} = \boldsymbol{R}\hat{\boldsymbol{\omega}}_b \tag{3.3}$$

$$\dot{\boldsymbol{v_b}} = \boldsymbol{M}^{-1}\left(\boldsymbol{M}\boldsymbol{v_b}\times\boldsymbol{\omega_b}+m_0g\boldsymbol{R}^T\boldsymbol{k}+\boldsymbol{F}_{ext}\right) \tag{3.4}$$

$$\dot{\boldsymbol{\omega}}_b = \boldsymbol{J}^{-1}\left(-\dot{\boldsymbol{J}}\boldsymbol{\omega_b}+\boldsymbol{J}\boldsymbol{\omega_b}\times\boldsymbol{\omega_b}+\boldsymbol{M}\boldsymbol{v_b}\times\boldsymbol{v_b}+\boldsymbol{T}_{ext}\right.$$
$$\left.+m_wg\boldsymbol{r_w}\times\left(\boldsymbol{R}^T\boldsymbol{k}\right)+\bar{m}g\boldsymbol{r_p}\times\left(\boldsymbol{R}^T\boldsymbol{k}\right)\right) \tag{3.5}$$

Here $\boldsymbol{M} = (m_s + \bar{m})\boldsymbol{I} + \boldsymbol{M}_f = diag\{m_1, m_2, m_3\}$, where $\boldsymbol{I}$ is the 3×3 identity matrix, and $\boldsymbol{M}_f$ is the added-mass matrix, which can be calculated via strip theory [85]. $\boldsymbol{J} = diag\{J_1, J_2, J_3\}$ is the sum of the inertia matrix due to the stationary mass distribution and the added inertia matrix in water. In addition, $\boldsymbol{k}$ is the unit vector along the Az direction in the inertial frame, $\boldsymbol{r_w}$ is a constant vector, and $\boldsymbol{r_p}$ is the controllable movable mass position vector, which has one degree of freedom in the Ox_b direction, $\boldsymbol{r_p} = \begin{bmatrix} r_{p1} & 0 & 0 \end{bmatrix}^T$. $\boldsymbol{b_i} = \begin{bmatrix} x & y & z \end{bmatrix}^T$ is the position vector of the robot in the inertial reference frame, $\hat{\boldsymbol{\omega}}_b$ is the skew-symmetric matrix corresponding to $\boldsymbol{\omega_b}$. $\boldsymbol{F}_{ext}$ stands for all external forces: the external thrust force $\boldsymbol{F_t}$ induced by tail flapping, and the external hydrodynamic forces (lift force, drag force and side force) acting on the gliding robotic fish body, expressed in the body-fixed frame. Finally, $\boldsymbol{T}_{ext}$ is the total hydrodynamic moment caused by $\boldsymbol{F}_{ext}$.

3.2 Hydrodynamic Model

3.2.1 Hydrodynamic Equations

In order to model the hydrodynamics, we first introduce the velocity reference frame $Ox_v y_v z_v$. Ox_v axis is along the direction of the velocity, and Oz_v lies in the sagittal plane perpendicular to Ox_v. Rotation matrix $\boldsymbol{R}_{bv}$ represents the rotation operation from the velocity reference frame to the body-fixed frame:

$$
\boldsymbol{R}_{bv} = \begin{pmatrix} c\alpha c\beta & -c\alpha s\beta & -s\alpha \\ s\beta & c\beta & 0 \\ s\alpha c\beta & -s\alpha s\beta & c\alpha \end{pmatrix},
\tag{3.6}
$$

where the angle of attack $\alpha = \arctan(v_3/v_1)$ and the sideslip angle $\beta = \arcsin(v_2/\|\boldsymbol{v_b}\|)$

The hydrodynamic forces include the lift force L, the drag force D, and the side force F_S; the hydrodynamic moments include the roll moment M_1, the pitch moment M_2, and the yaw moment M_3. All of those forces and moments are defined in the velocity frame [86]. And if we further assume that the tail is flapping relatively slowly and smoothly, usually true for the yaw control motion during steady glide, the propelling force from the tail will be negligible compared to the buoyancy-induced propelling force, which means $\boldsymbol{F_t} = \boldsymbol{0}$. Then we will have the following relationship:

$$
\boldsymbol{F}_{ext} = \boldsymbol{R}_{bv} \begin{bmatrix} -D & F_S & -L \end{bmatrix}^T
\tag{3.7}
$$

$$
\boldsymbol{T}_{ext} = \boldsymbol{R}_{bv} \begin{bmatrix} M_1 & M_2 & M_3 \end{bmatrix}^T
\tag{3.8}
$$

The hydrodynamic forces and moments are dependent on the angle of attack α, the sideslip

angle β, the velocity magnitude V [87–90], and the tail angle δ:

$$D = 1/2\rho V^2 S(C_{D0} + C_D^\alpha \alpha^2 + C_D^\delta \delta^2) \tag{3.9}$$

$$F_S = 1/2\rho V^2 S(C_{F_S}^\beta \beta + C_{F_S}^\delta \delta) \tag{3.10}$$

$$L = 1/2\rho V^2 S(C_{L0} + C_L^\alpha \alpha) \tag{3.11}$$

$$M_1 = 1/2\rho V^2 S(C_{M_R}^\beta \beta + K_{q1}\omega_1) \tag{3.12}$$

$$M_2 = 1/2\rho V^2 S(C_{M0} + C_{M_P}^\alpha \alpha + K_{q2}\omega_2) \tag{3.13}$$

$$M_3 = 1/2\rho V^2 S(C_{M_Y}^\beta \beta + K_{q3}\omega_3 + C_{M_Y}^\delta \delta) \tag{3.14}$$

where ρ is the density of water and S is the characteristic area of the gliding robotic fish. The tail angle δ is defined as the angle between the longitudinal axis Ox_b and the center line of the tail projected into the $Ox_b y_b$ plane, with Oz_b axis as the positive direction. K_{q1}, K_{q2}, K_{q3} are rotation damping coefficients. All other constants with 'C' in their notations are hydrodynamic coefficients, whose values can be evaluated through CFD-based water tunnel simulation [82].

3.2.2 CFD-based Evaluation of Hydrodynamic Coefficients

In this dissertation we use CFD simulation to obtain the hydrodynamic coefficients for any given gliding robotic fish body geometry. Experimental methods like towing experiments can be further used to verify the CFD results [91], [92].

We first look into the influence of the angle of attack on the hydrodynamic forces and moments. We simulate the steady gliding motion when the sideslip angle and the tail angle (no actively-controlled tail for miniature underwater) are both zero in order to eliminate their influence. In the simulation the gliding robotic fish model is created in SolidWorks 2009. We use Gambit 2.3.16 to
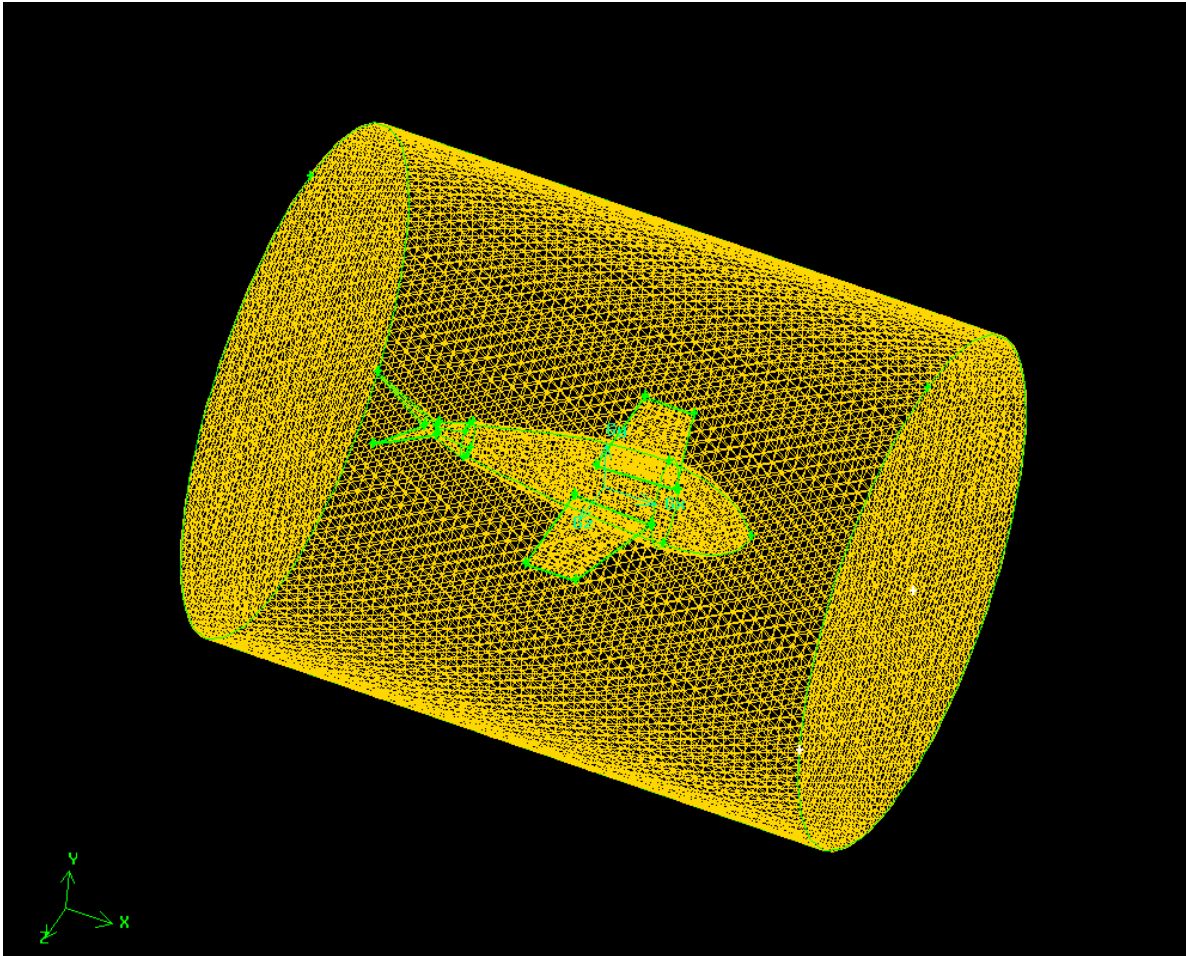
Figure 3.3: The meshing used for the water tunnel simulation for miniature underwater glider prototype (generated with Gambit).

2.10e+01
1.93e+01
1.75e+01
1.58e+01
1.41e+01
1.23e+01
1.06e+01
8.85e+00
7.11e+00
5.37e+00
3.63e+00
1.90e+00
1.58e-01
-1.58e+00
-3.32e+00
-5.05e+00
-6.79e+00
-8.53e+00
-1.03e+01
-1.20e+01
-1.37e+01

Contours of Static Pressure (pascal)

ANSYS
Noncommercial use o

Sep 05, 2011
ANSYS FLUENT 12.1 (3d, pbns, lam)
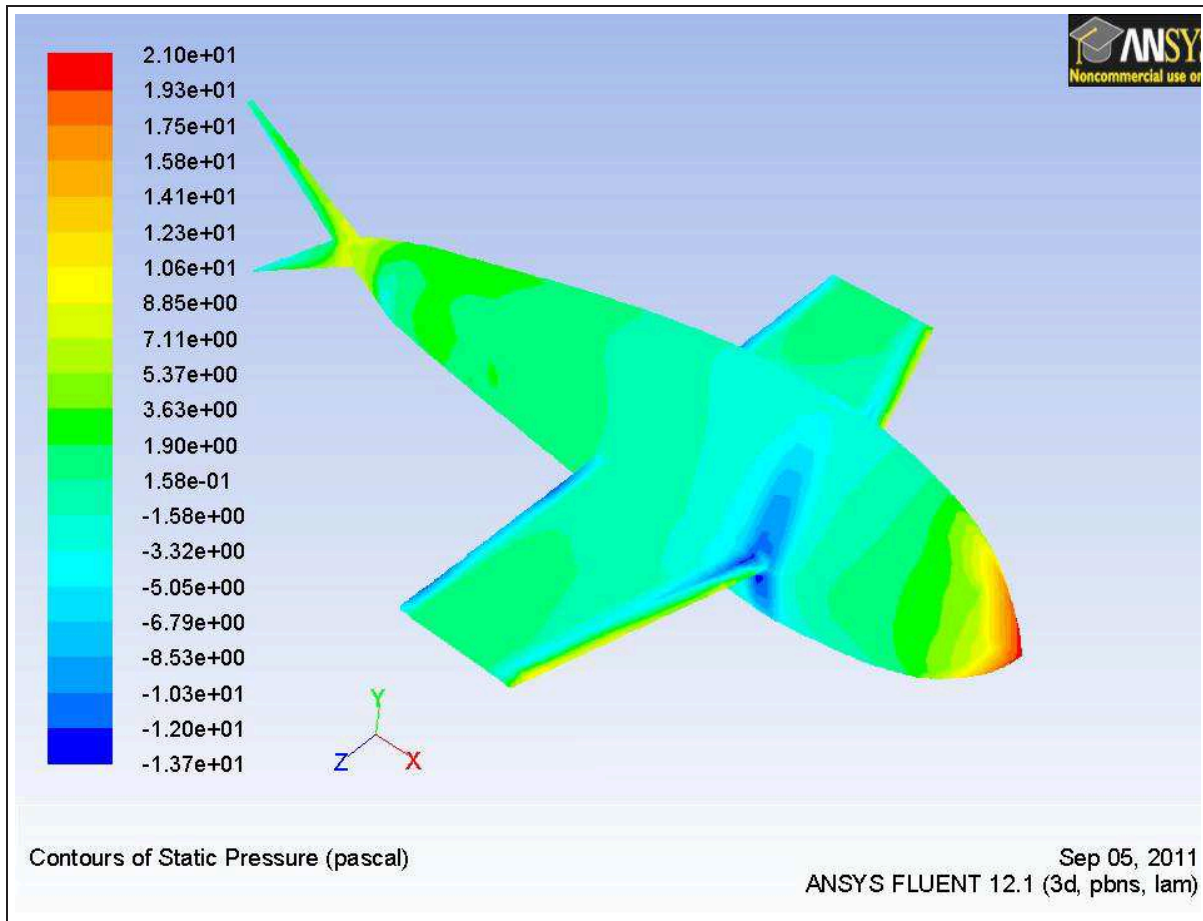
Figure 3.4: An example contour of the static pressure with the angle of attack and the sideslip angle set as zero for miniature underwater glider prototype.

create a mesh file (Fig. 3.3), which contains the shape and size information of the gliding robotic fish, and then use Fluent 6.2.16 to simulate the flow and pressure distribution around the robot body (Fig. 3.4), which is placed in a water channel. With a virtual water tunnel as the simulation workspace, the boundary conditions are set to different inlet velocities, and under such different boundary conditions, CFD simulations are run with different angles of attack. With given values for the characteristic area and length, we form a table for the convergent coefficients obtained from CFD simulation for lift force, drag force and pitch moment. From the convergent results, lift, drag and pitch moment functions are approximated with polynomials in α by curve fitting. When tail angle δ and sideslip angle β are fixed at zero, drag coefficient C_D, lift coefficient C_L, and pitch moment coefficient C_{M2} can be expressed as

$$C_D = C_{D_0} + C_D^\alpha \alpha^2,$$

$$C_L = C_{L_0} + C_L^\alpha \alpha,$$

$$C_{M_2} = C_{M_0} + C_{M_P}^\alpha \alpha,$$

Data fitting is conducted to compute the drag, lift, and pitch moment coefficients, as a function of the angle of attack α. For example, Fig. 3.5, Fig. 3.6 and Fig. 3.7 show the fitted function for the drag, lift and pitch moment coefficients, respectively, for the miniature underwater glider model. The constants in the lift, drag, and pitch moment functions are estimated to be:

$$C_{D_0} = 0.45275, \ C_D^\alpha = 17.5948,$$

$$C_{L_0} = 0.074606, \ C_L^\alpha = 19.5777,$$

$$C_{M_0} = 0.0075719, \ C_{M_P}^\alpha = 0.5665.$$

As a comparative trial, another set of wings is used with the same wingspan but doubled aspect
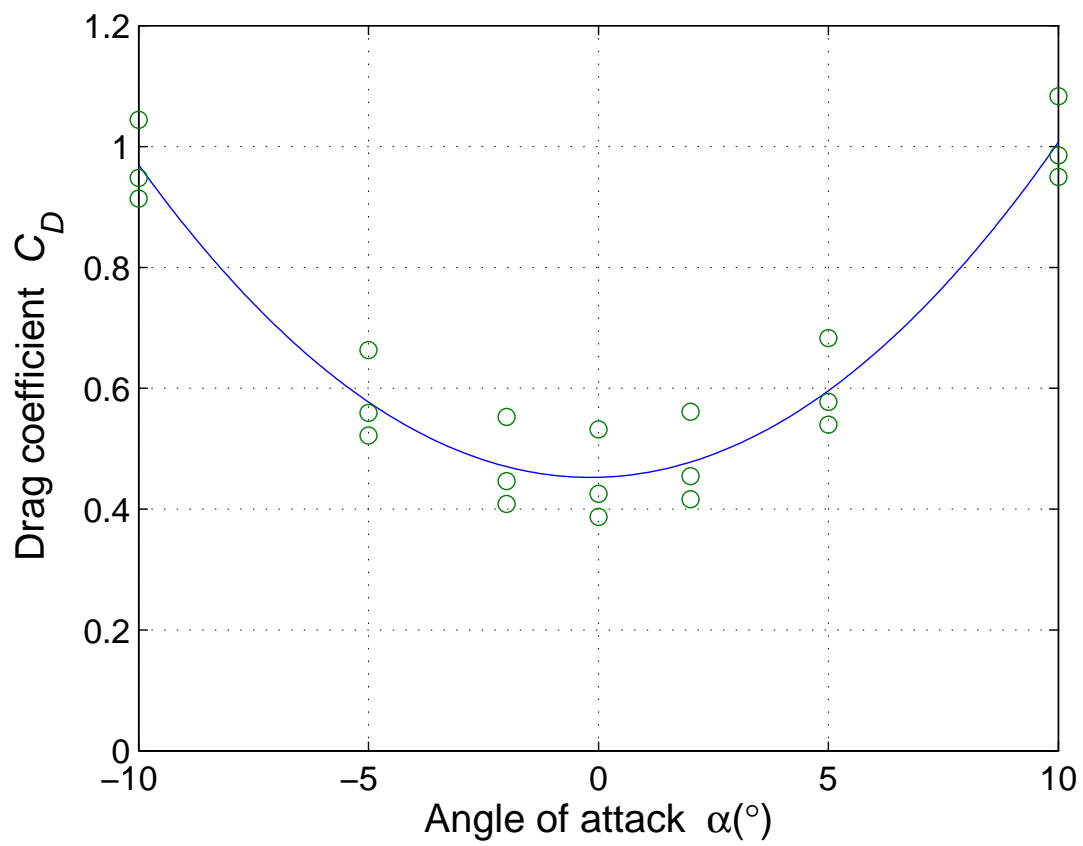
Figure 3.5: Data fitting for the drag force coefficient as a function of α.
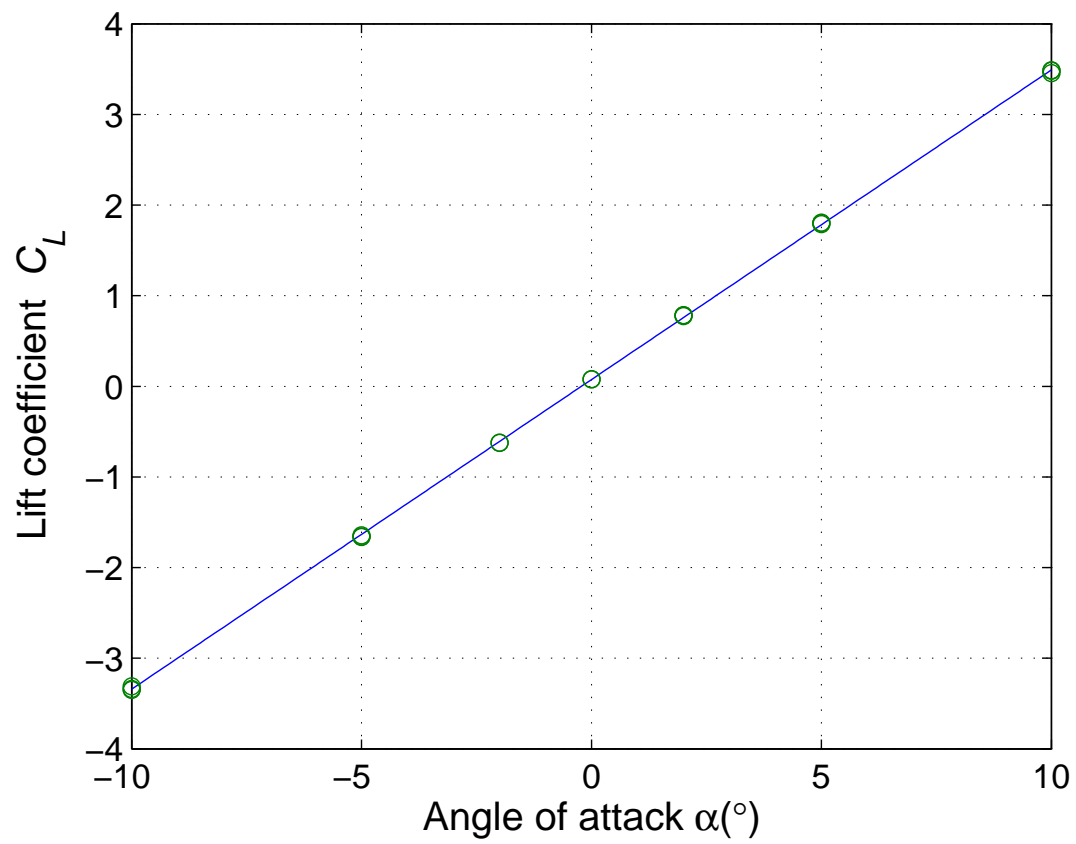
Figure 3.6: Data fitting for the lift force coefficient as a function of α.
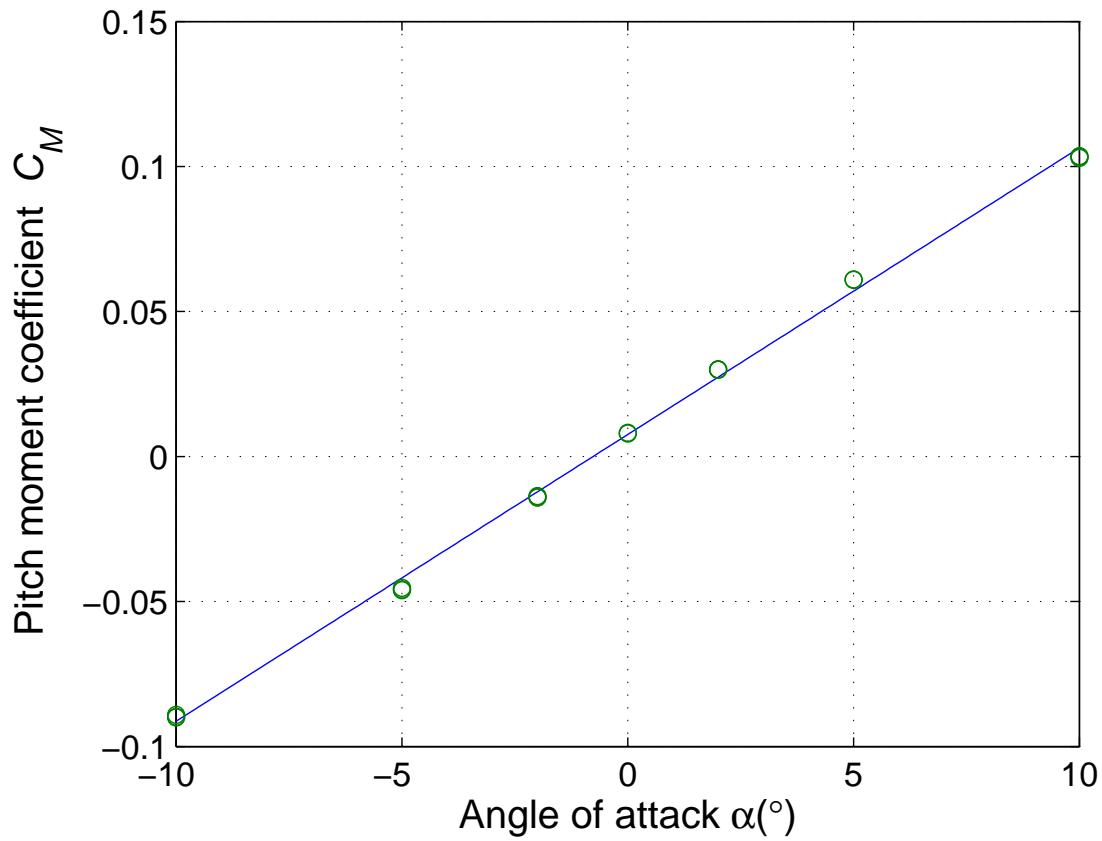
Figure 3.7: Data fitting for the pitch moment coefficient as a function of α.
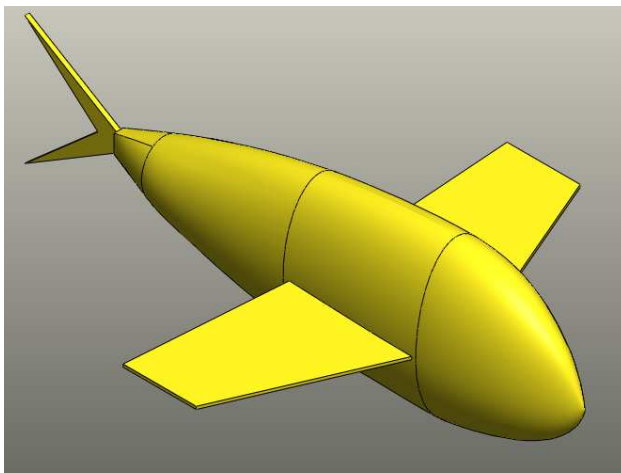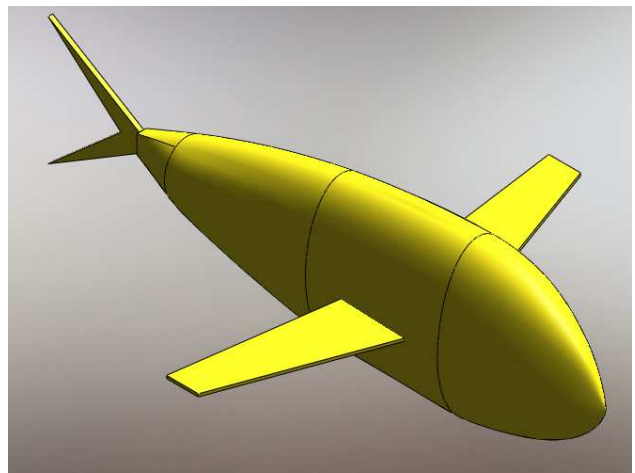
(a) larger wings   (b) smaller wings

Figure 3.8: Illustration of the glider body with two different wing designs. Area of wings in (b) is half of that in (a).
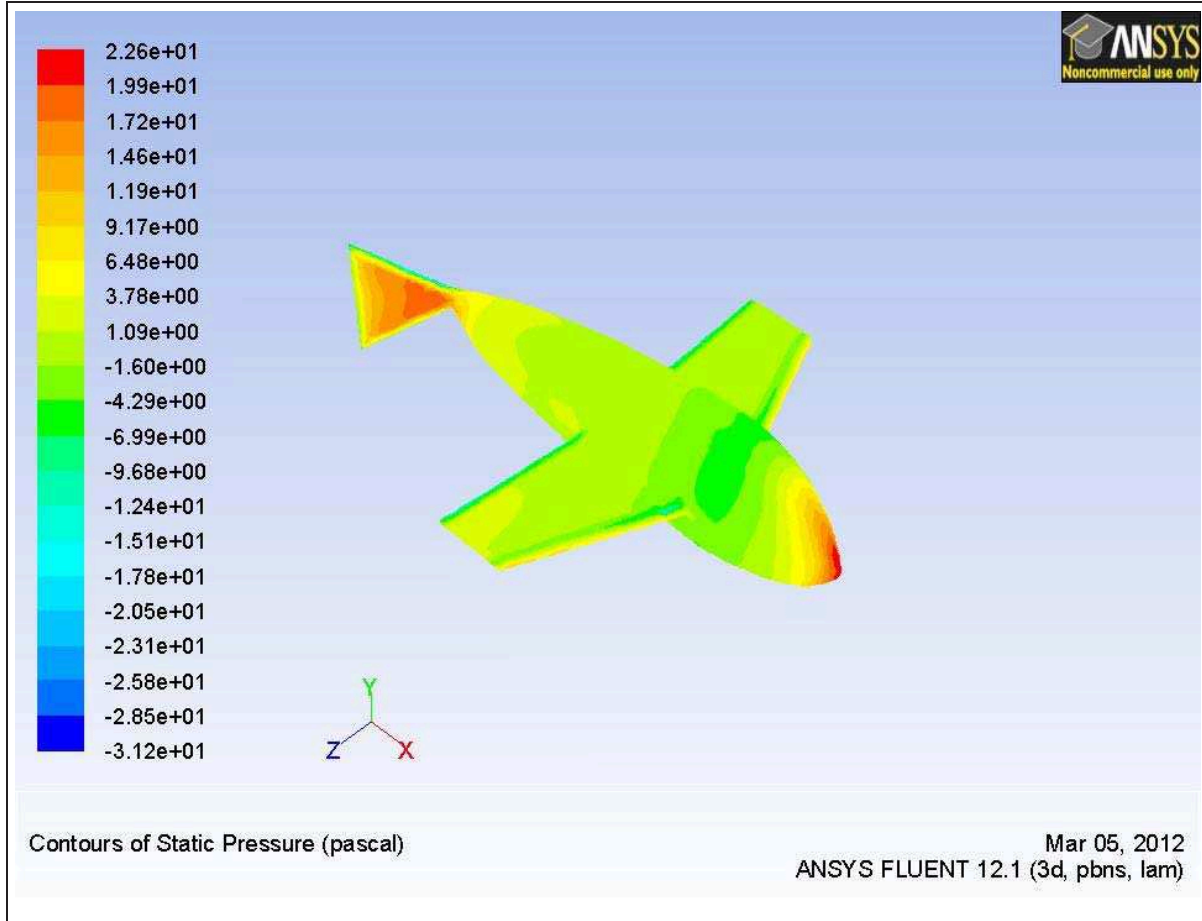
Figure 3.9: An example contour of the static pressure with the angle of attack and the sideslip angle set as zero for gliding robotic fish prototype.

ratio (i.e., chord length is half), while the body is left unchanged. Fig. 3.8 illustrates the two models created in SolidWorks that have different sets of wings. The hydrodynamic coefficients relevant to the angle of attack for the one with smaller wings are:

$$C_{D_0} = 0.44724, \ C_D^\alpha = 10.298,$$

$$C_{L_0} = 0.054273, \ C_L^\alpha = 11.5545,$$

$$C_{M_0} = 0.0062683, \ C_{M_P}^\alpha = 0.2903.$$

With similar CFD water-tunnel simulation, we can obtain the hydrodynamic coefficients re-

garding the sideslip angle when we set the angle of attack and the tail angle to zero. And the hydrodynamic coefficients involving the tail angle can be obtained similarly by setting the other two hydrodynamic angles to zero (3.9). Here we want to point out that in this dissertation we ignore the coupling effect of those three angles (the sideslip angle, the angle of attack and the tail angle) on the dynamics of the glider.

Chapter 4

Steady Gliding in the Sagittal Plane

4.1 Reduced Model in the Sagittal Plane

Steady-state gliding in the sagittal plane is one of most important working patterns for the gliding

robotic fish. When the robot motion is restricted to the sagittal plane, we have

$$
R = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix}, \quad b_i = \begin{pmatrix} x \\ 0 \\ z \end{pmatrix}, \quad v_b = \begin{pmatrix} v_1 \\ 0 \\ v_3 \end{pmatrix},
$$

$$
\omega_b = \begin{pmatrix} 0 \\ \omega_2 \\ 0 \end{pmatrix}, \quad r_p = \begin{pmatrix} r_{p1} \\ 0 \\ 0 \end{pmatrix}, \quad r_w = \begin{pmatrix} 0 \\ 0 \\ r_{w3} \end{pmatrix}, \quad \delta = 0.
$$

Here we assume the point mass m_w is just below the center of geometry O by r_{w3} as such bottom-

heavy design is desirable for stability concern and also achievable with manufacture. Plugging the

hydrodynamic forces and moment into the robot dynamics equations, we get the following model:

$$\dot{v}_1 \;=\; (m_1 + \bar{m})^{-1}\big(-(m_3 + \bar{m})\,v_3\omega_2 - m_0 g\sin\theta + L\sin\alpha - D\cos\alpha\big) \tag{4.1}$$

$$\dot{v}_3 \;=\; (m_3 + \bar{m})^{-1}\big((m_1 + \bar{m})\,v_1\omega_2 + m_0 g\cos\theta - L\cos\alpha - \sin\alpha\big) \tag{4.2}$$

$$\dot{\omega}_2 \;=\; J_2^{-1}\big(M_2 + (m_3 - m_1)v_1 v_3 - m_w g r_{w3}\sin\theta - \bar{m}g r_{p1}\cos\theta\big) \tag{4.3}$$

$$\dot{x} \;=\; v_1\cos\theta + v_3\sin\theta \tag{4.4}$$

$$\dot{z} \;=\; -v_1\sin\theta + v_3\cos\theta \tag{4.5}$$

$$\dot{\theta} \;=\; \omega_2 \tag{4.6}$$

During steady glide, the angular velocity is zero, while the velocity stays unchanged. The control r_{p1} and m_0 are constant, which means that the position of the movable mass is fixed with respect to the origin O and the pumping rate is zero. So the steady motion can be described by:

$$0 \;=\; -m_0 g\sin\theta + L\sin\alpha - D\cos\alpha \tag{4.7}$$

$$0 \;=\; m_0 g\cos\theta - L\cos\alpha - D\sin\alpha \tag{4.8}$$

$$0 \;=\; M_2 + (m_3 - m_1)v_1 v_3 - m_w g r_{w3}\sin\theta - \bar{m}g r_{p1}\cos\theta \tag{4.9}$$

The solution to the above equation gives us the steady gliding path.

4.2   Computation of Steady Gliding Path in the Sagittal Plane

With the hydrodynamic parameters obtained from CFD simulation, let us take a look at the solution of the steady gliding equations (4.7)–(4.9). These equations are highly nonlinear due to the terms involving trigonometric functions and inverse trigonometric functions in the state. When the angle

Table 4.1: Computed steady gliding path under different values of the center of gravity z_{CG}, the movable mass displacement r_{p1}, and the excess mass m_0, for the two models shown in Fig. 3.8.

(a) larger wings

z_{CG} (cm)	r_{p1} (cm)	m_0 (g)	$\left(V, \alpha, \theta_g\right)$ (m/s,°,°)
0.1	0.3	10	(0.1129 3.0470 -29.5522)
0.1	0.5	10	(0.1366 1.6543 -43.0404)
0.1	0.7	10	(0.1485 1.0936 -52.7389)
0.1	0.3	30	(0.1766 3.9483 -25.0827)
0.1	0.5	30	(0.2245 2.0106 -38.4395)
0.1	0.7	30	(0.2495 1.3300 -48.7594)
0.1	0.3	50	(0.2245 4.0967 -24.5276)
0.1	0.5	50	(0.2846 2.1371 -37.0375)
0.1	0.7	50	(0.3174 1.3980 -47.0516)
0.2	0.3	10	(0.0856 5.8988 -20.2069)
0.2	0.5	10	(0.1084 3.3827 -27.6331)
0.2	0.7	10	(0.1240 2.3211 -35.1835)

(b) smaller wings

z_{CG} (cm)	r_{p1} (cm)	m_0 (g)	$\left(V, \alpha, \theta_g\right)$ (m/s,°,°)
0.1	0.3	10	(0.1221 4.0658 -37.0187)
0.1	0.5	10	(0.1396 2.4940 -48.3662)
0.1	0.7	10	(0.1486 1.7575 -56.4820)
0.1	0.3	30	(0.2260 3.2732 -41.9002)
0.1	0.5	30	(0.2477 2.2108 -51.2303)
0.1	0.7	30	(0.2598 1.6385 -58.0061)
0.1	0.3	50	(0.3105 2.5525 -47.8110)
0.1	0.5	50	(0.3290 1.8747 -55.0414)
0.1	0.7	50	(0.3401 1.4598 -60.4123)
0.2	0.3	10	(0.0949 7.7979 -26.1314)
0.2	0.5	10	(0.1136 5.0110 -32.7957)
0.2	0.7	10	(0.1265 3.6361 -39.4833)

of attack is small enough, we can use the approximation $\sin\alpha \approx \alpha$ and $\cos\alpha \approx 1$, and derive an approximate analytical solution for the desired control $\boldsymbol{r_p}$ and m_0 in order to achieve some given steady states [74]. However, here we are interested in the calculation of the steady gliding states themselves under a fixed control. Unfortunately, there are no feasible analytical solutions for this problem. So with Matlab command *solve()*, we numerically solve equations (4.7)–(4.9) to get the velocity $\boldsymbol{v}$, pitch angle θ, and glide angle θ_g for a given movable mass displacement $\boldsymbol{r_p}$ and net buoyancy m_0, under different conditions for $\boldsymbol{r_w}$, the location of nonuniform stationary mass. There is only one feasible solution for each pair of $(\boldsymbol{r_p}, m_0)$. Other solutions are rejected based on their physical interpretations.

Table 4.1 shows scan results where the steady gliding path is presented with different sets of center of mass distribution, location of movable mass, and net buoyancy, for both wing designs illustrated in Section 3.2.2. The gliding angle $\theta_g = \theta - \alpha$ is the angle between Ox_v and Ax with gliding up as positive. z_{CG} stands for the center of gravity expressed in the z-axis coordinate of the body-fixed frame and there is a bijective function from r_{w3} to z_{CG}:

$$z_{CG} = \frac{m_w}{m} r_{w3}. \tag{4.10}$$

Here we ignore the influence of the excess mass m_0 on z_{CG}, which is really small compared to that of m_w. For example, m_0 is generally around 30 g while m_w is up to several kilograms. From the results, we can see that different wing designs lead to different static gliding profiles. For example, the larger wings result in shallower gliding paths (longer horizontal travel) but slower total speed compared to the smaller wings, given the same set of control inputs. Since the wings can be easily replaced in our design, we can potentially tailor the wing designs, while leaving the glider body and its inside intact, to accommodate the requirements of different applications. On

the other hand, the results in the table show that, for a fixed wing design, the speed is influenced by both the excess mass m_0 and the pair $(\boldsymbol{r_p}, z_{CG})$ while the pitch angle is affected mainly by the pair $(\boldsymbol{r_p}, z_{CG})$. Therefore, the center of mass plays an important role in determining the steady gliding attitude. In particular, if we compare the cases where the values of z_{CG} are different but the other parameters are the same, we find that smaller z_{CG} results in higher speed and larger glide angle. This observation has been used in our design – by making z_{CG} small, we can achieve desired glide angle with very small displacement of the movable mass.

4.3   Scaling Analysis

We study the larger-wing glider model (Fig. 3.8a) at different scales and introduce a new cost performance index, which reflects the horizontal travel distance per unit energy consumption. For one dive (descent and ascent), the horizontal travel distance D_d is approximated as

$$D_d = V_h t_d = 2\frac{V_h h}{V_v},$$ (4.11)

where V_h and V_v are the steady-state horizontal speed and vertical speed, respectively, t_d is the travel time for one dive, and h is the vertical travel depth. The energy consumption in one dive comes from two sources, the pump actuation and the movable mass actuation, while the energy consumed for the latter is negligible compared to that for pumping since the pump needs to overcome large pressure when the glider switches to ascent from descent. So the energy consumption per dive E_d can be approximated as

$$E_d = \rho g h_0 S_p l_p + \rho g (h_0 + h) S_p l_p.$$ (4.12)

Table 4.2: Computed steady gliding path for the scaled models of the larger wing prototype. In computation, $r_p = 5mm$ is used for the original scale model (1:1) while the value is scaled linearly with dimension for other models.

scale	mass (kg)	m_0 (kg)	V_h (m/s)	V_v (m/s)	$\frac{V_h}{V_v m_0}$ (kg^{-1})	glide ratio
0.25:1	1	0.0075	0.063	0.018	488.35	3.5
0.5:1	2	0.015	0.11	0.039	203.25	2.82
1:1	4	0.03	0.19	0.094	74.55	2.02
2:1	8	0.06	0.28	0.207	28.30	1.35
4:1	16	0.12	0.39	0.377	12.01	1.03
8:1	32	0.24	0.54	0.574	5.72	0.94

Here, ρ is the water density, h_0 is the equivalent water depth of the atmosphere pressure, S_p is the cross-section area of the pump tank inlet (and outlet) and l_p represents the length of the water column if the water pumped in each cycle is placed in a cylindrical container with cross-section area S_p. Noting the net buoyancy $m_0 = \frac{1}{2}\rho S_p l_p$, we further simplify the energy consumption per dive to $E_d = 2m_0 g(2h_0 + h)$. Then we have the horizontal travel distance per unit energy consumption

$$\frac{D_d}{E_d} = \frac{V_h}{V_v m_0} \frac{1}{1 + 2\frac{h_0}{h}}. \tag{4.13}$$

For a specific task, the depth is fixed and we have

$$\frac{D_d}{E_d} \propto \frac{V_h}{V_v m_0}, \tag{4.14}$$

which we define as the cost performance index τ.

We now conduct scaling analysis to examine how the cost performance metric evolves with the dimensions (and consequently the weight) of the glider. CFD simulation shows that the drag coefficient C_D and lift coefficient C_L stay almost the same at different scales we considered (from 0.25:1 to 8:1), while the pitch moment coefficient C_M scales linearly with the characteristic dimen-
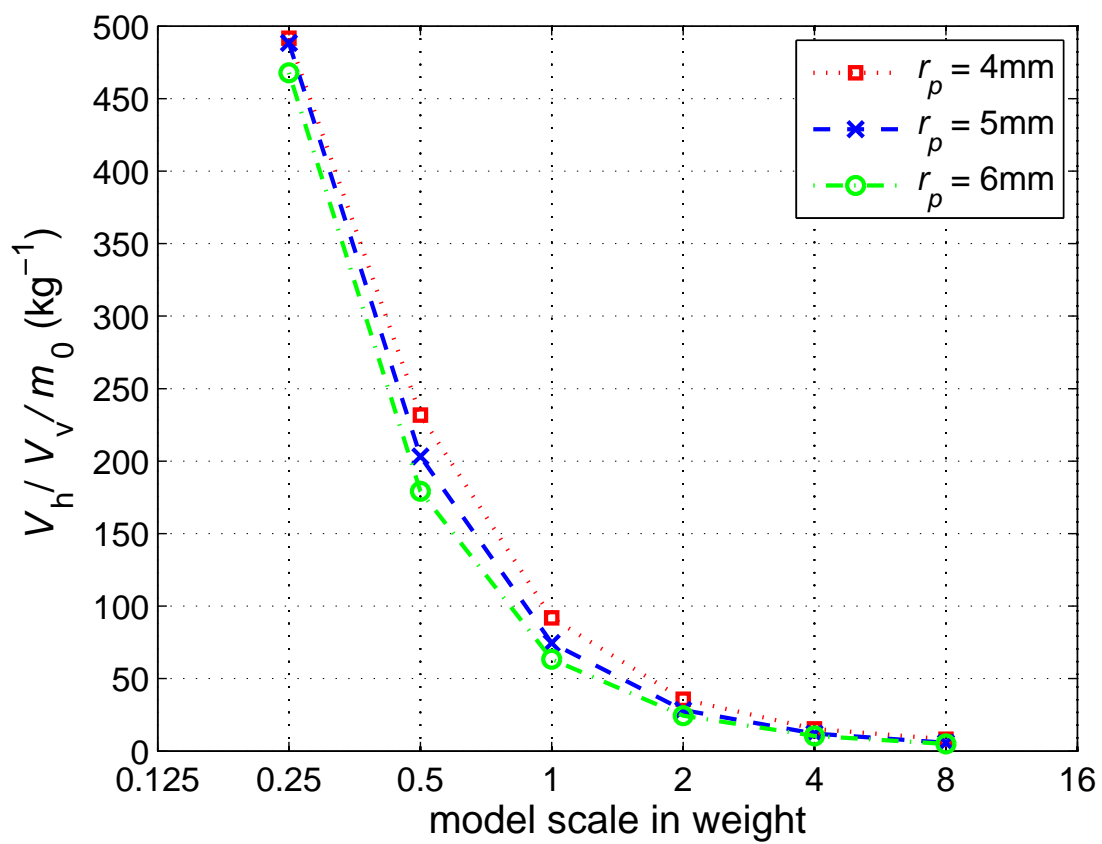
Figure 4.1: The glider cost performance index with respect to model scales.
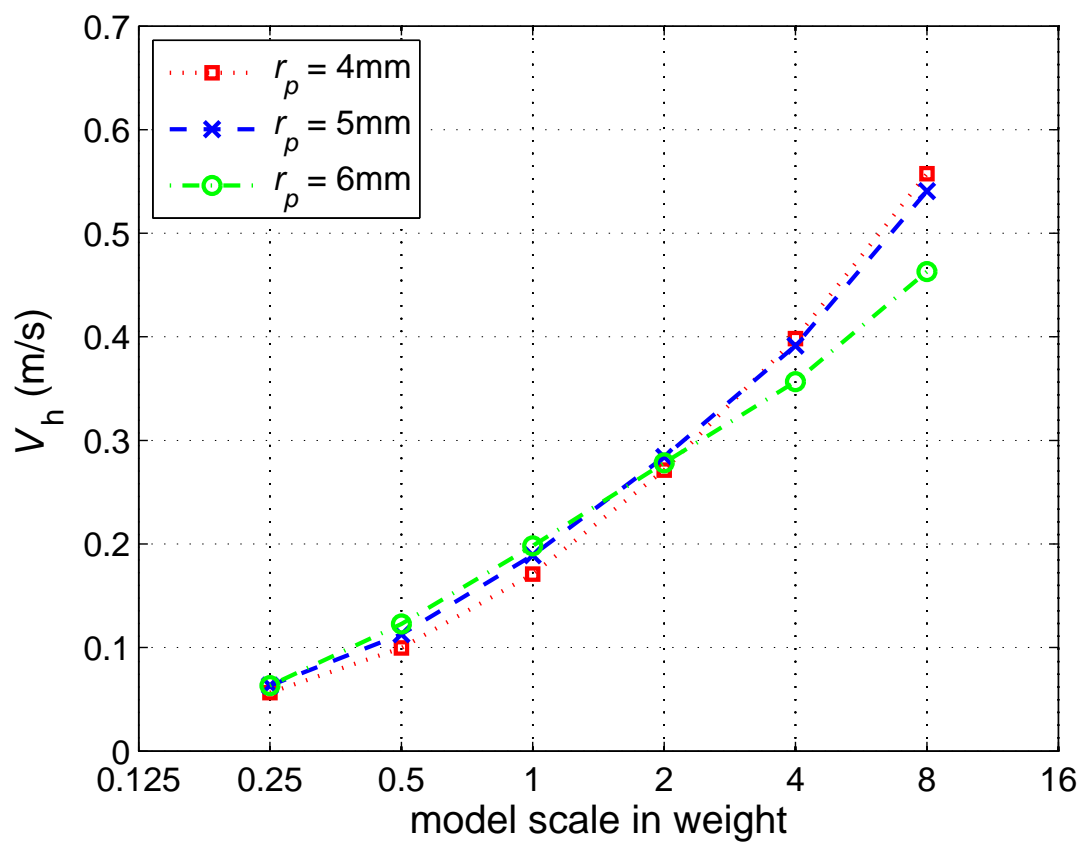
Figure 4.2: The horizontal velocity with respect to model scales.

sion l of the glider. All related masses of the glider will scale as l^3, including the movable mass $\bar{m}$ and the negative net buoyancy m_0. Taking the total length of the glider as l, the scale 1:1 would imply $l = 50$ cm. By plugging those new parameter values into equations (4.7)–(4.9), we can solve the glide path for the scaled model.

Table 4.2 shows the glide paths for glider models at different scales. Fig. 4.1 shows the relationship between the cost performance index $\frac{V_h}{V_v m_0}$ and the scale. The results show us that with a larger body the glider has a smaller glide ratio and a smaller cost performance index value, thus consuming more energy for a given horizontal travel distance. This is consistent with the fact that a larger glider needs to pump more water for a proper net buoyancy to provide the propelling force, which is also the main energy consumption source. However, a larger-scale glider is able to achieve faster horizontal speed as shown in Fig. 4.2. There is a trade-off between the achieved horizontal speed and the horizontal distance coverage per unit energy cost, when selecting the optimal scale for the glider. Other factors, like the dimension and the mass of the sensors and actuation devices, should be also taken into account in the design process.

4.4   Experimental Results and Model Validation

With the developed miniature glider prototype, underwater gliding experiments are conducted in order to validate the model. Most experiments are conducted in a large water tank that measures 15-foot long, 10-foot wide, and 4-foot deep.

For the steady gliding in the sagittal plane, we use the zero-angle tail. First, we set the initial net buoyancy (negatively buoyant) and the linear actuator position to desired values. Then the glider is released on the water surface with a stopwatch started, and the glider dives down until it reaches the programmed depth indicated by the pressure sensor. Then it pumps water out and

resets its attitude to glide up. Here we focus on the gliding-down section and record the whole gliding process with an underwater video camera fixed inside the water tank, pointing directly to the glide motion plane (Fig. 4.3). Then we conduct video post-processing to extract the steady gliding data, including the operating depth, horizontal travel distance and the time spent. We first carry out one series of experiments with net buoyancy fixed at -20 g, and vary the linear actuator positions. Then in another series of experiments, we vary the net buoyancy while holding linear actuator position at 0.5 cm forward. For each setting of net buoyancy and linear actuator position, we repeat the experiments 10 times and evaluate both the means and standard deviations for the measured variables. Most experiments are conducted with the larger set of wings, but we have also experimented with the smaller set of wings to further validate the model.

There are several things that must be given careful treatment. First, although the pressure sensor can provide accurate depth measurement when the glider is at rest, the measurement is subject to larger error when the glider is moving through the water. Therefore, we have chosen to measure the actual depth each time. Second, we need the value of the center of gravity of the glider z_{CG} to obtain model-based predictions. We calculate it by hanging the glider up at different points on the glider with strings and then taking the intersection of different hanging strings. This value is further fine-tuned with collected tank test data. Third, to deal with the image distortion, we have made a grid board for calibration. The board measures 2.5 m by 1.5 m, with grid cell size of 10 cm by 10 cm. We fix the grid board in the glide motion plane and take its images. Then with post-processing techniques, those grid images are incorporated into the gliding videos, to facilitate the extraction of glide paths. Fourth, there is a period of transients before the glider reaches steady gliding. To minimize the error introduced by the transients, we have chosen to use only the data from last three seconds of descent in each run to compute the steady glide parameters.

Fig. 4.4 and Fig. 4.5 show the comparison between model predictions and experiment results
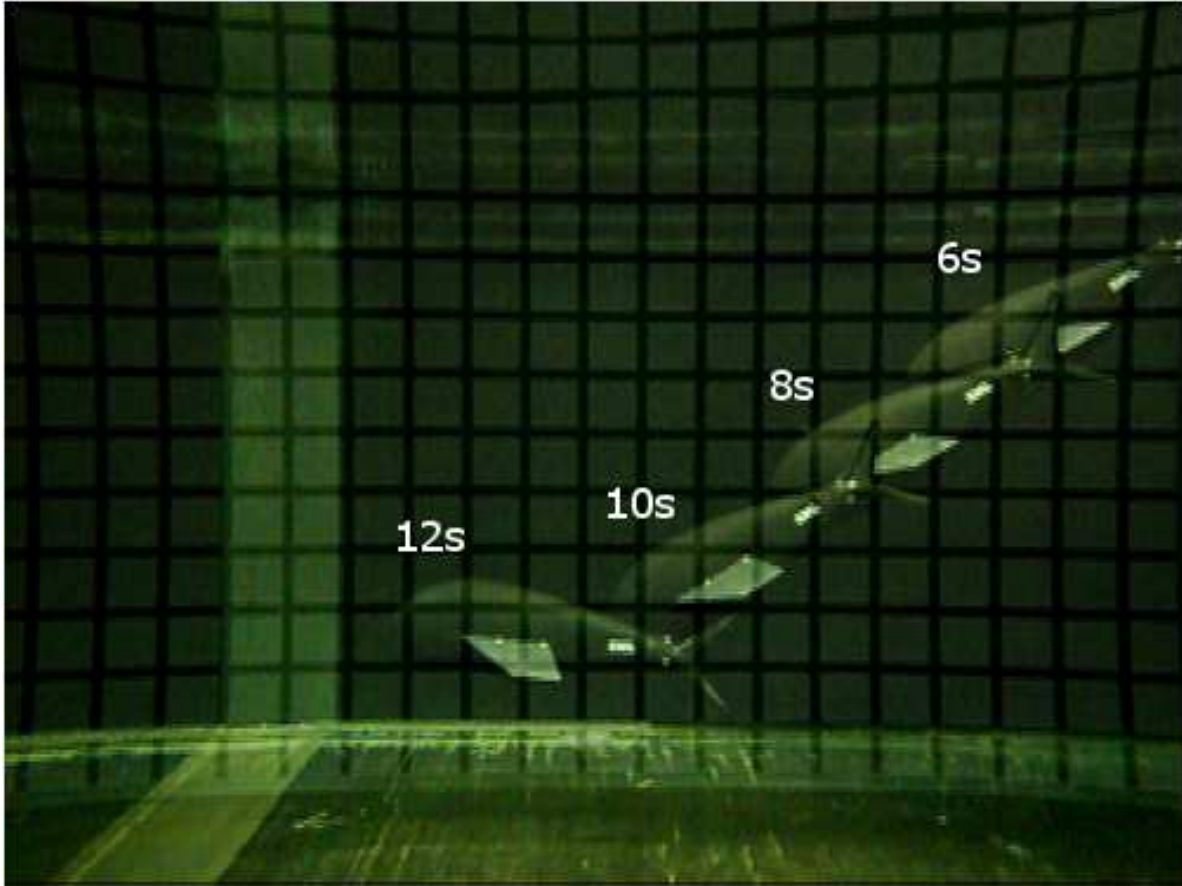
Figure 4.3: Superimposed snapshot of the gliding experiments in an indoor tank.
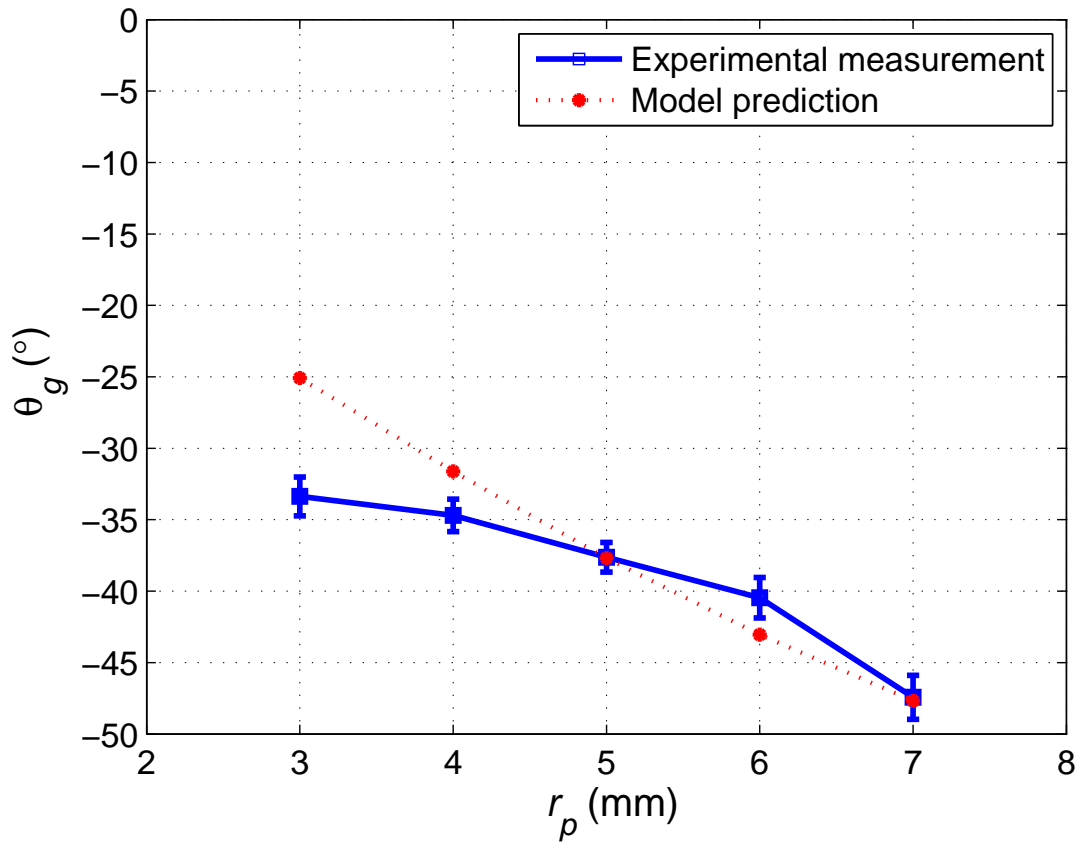
Figure 4.4: Glide angle with respect to the movable mass displacement, with fixed net buoyancy of -20 g, for the prototype with larger wings.
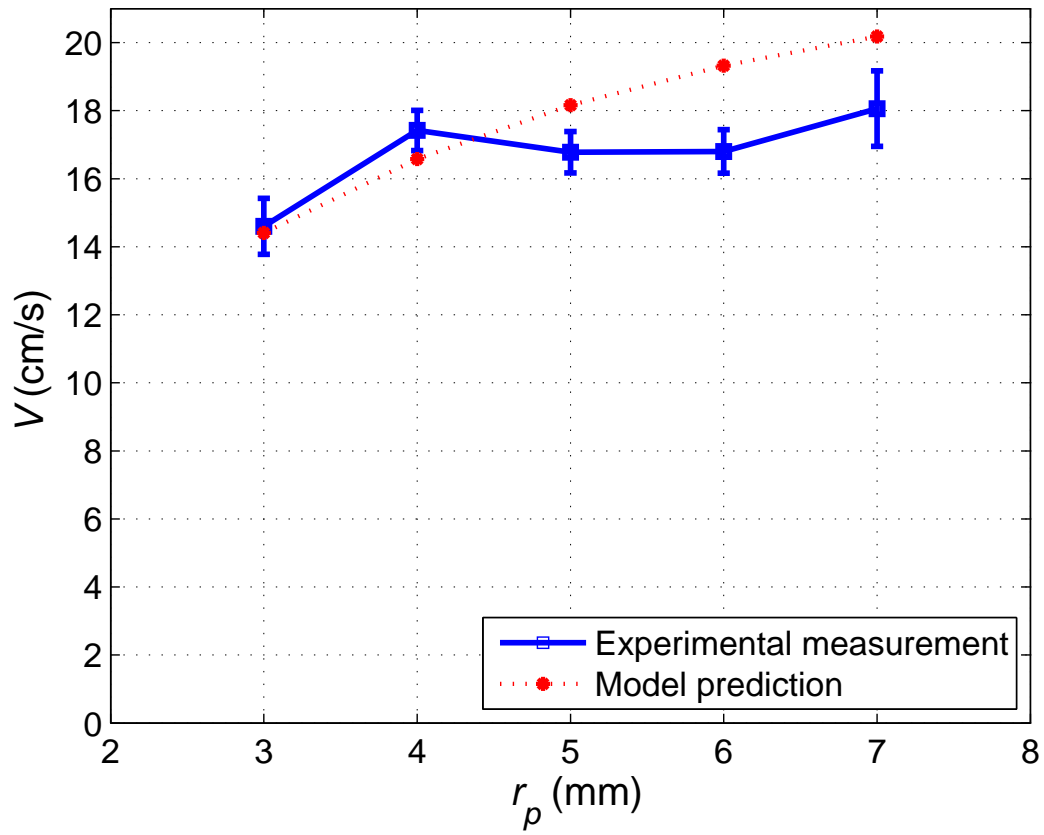
Figure 4.5: Total glide speed with respect to the movable mass displacement, with fixed net buoyancy of -20 g, for the prototype with larger wings.
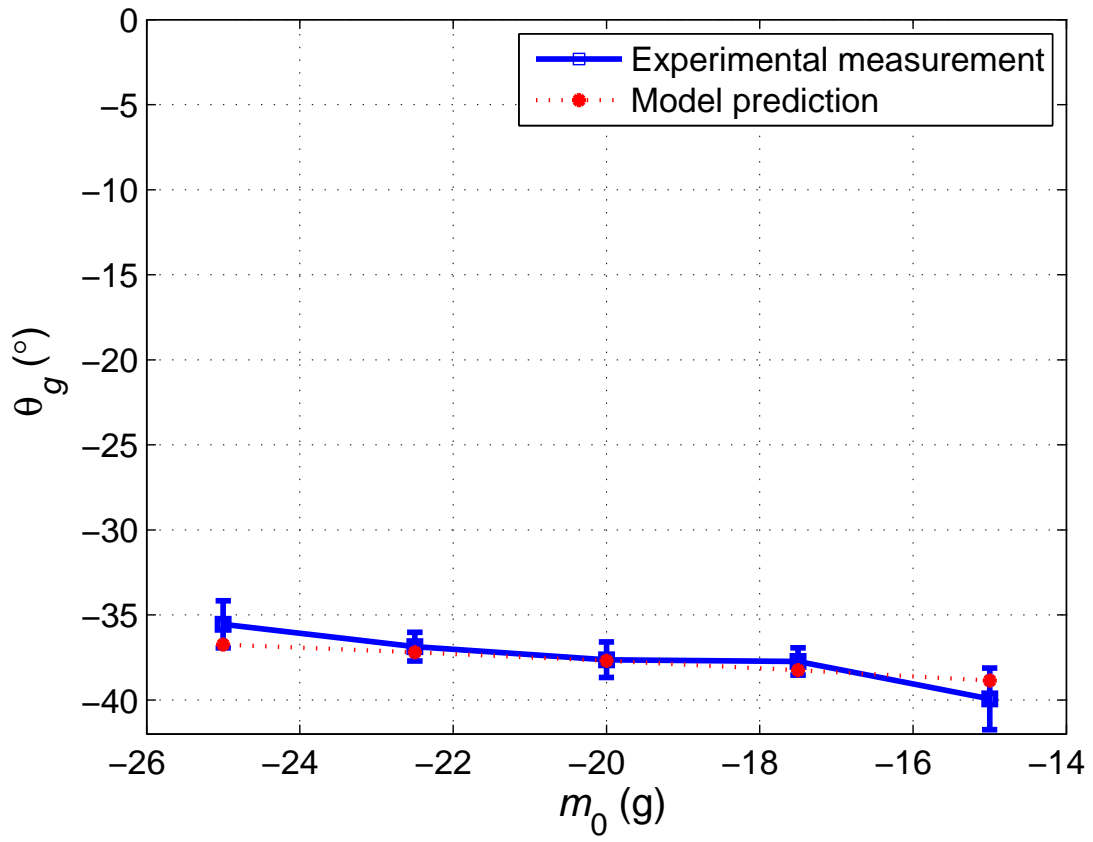
Figure 4.6: Glide angle with respect to the net buoyancy, with fixed movable mass displacement of 0.5 cm, for the prototype with larger wings.
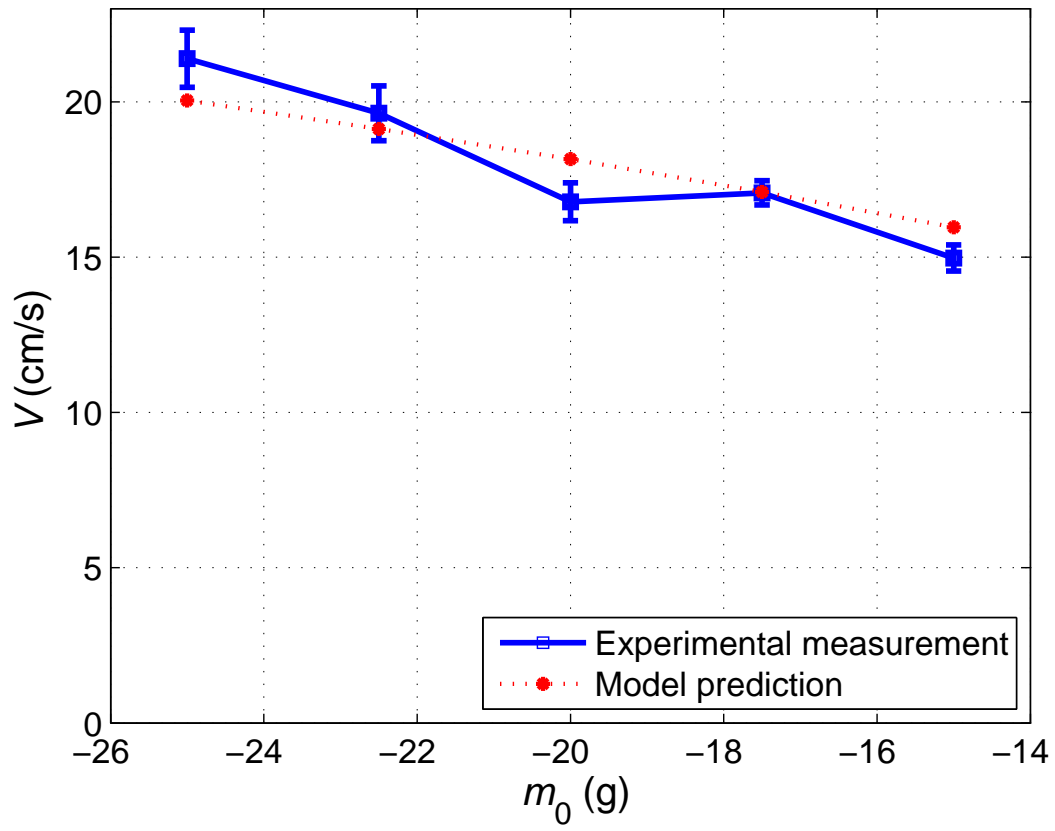
Figure 4.7: Total glide speed with respect to the net buoyancy, with fixed movable mass displacement of 0.5 cm, for the prototype with larger wings.

when we vary the movable mass position while holding the net buoyancy fixed, and Fig. 4.6 and Fig. 4.7 show the results when the net buoyancy is changing while the movable mass location is fixed. From the comparison results, we can see that the velocity and glide angle calculated from the model match the experimental data reasonably well. In particular, the model has predicted well the trends of how glide speed and angle vary with the center of the gravity and the net buoyancy.

We note that there are some non-negligible factors contributing to errors in the measurement. When the glider starts gliding from rest, it is accelerating rather than steadily gliding for a few seconds. We have already tried to remove the accelerating section from the data in the video processing process; however, it is difficult to completely eliminate that effect, especially considering the relatively shallowness of the test tank. This effect would be reduced with deeper gliding; however, conducting precise measurement in a deep water body itself presents challenges. In addition, flow disturbances in the tank influence the experiment results as well. So with these uncertainties, we consider the match between our experimental results and the model predictions in Figs. 4.4 – 4.7 satisfactory.

We have further compared the glide performance when the glider is equipped with the larger and smaller wings, respectively. Fig. 4.8 and Fig. 4.9 show the glide angle and the glide speed, respectively, as the movable mass displacement is varied while the net buoyancy is held fixed at -20 g. Both model predictions and experimental measurement are shown in the figures, and they match well for both sets of wings. From the results, we can see that with smaller wings, the glider tends to have a deeper gliding profile but higher speed, which matches our model predictions. These results further validate our derived model, and prove the effectiveness of our design method with CFD-based evaluation of hydrodynamic coefficients in Section 3.2.2. Meanwhile, the results also indicate that we can realize various gliding performance and meet different mission requirements by replacing the wings, which are designed to be easy to change.
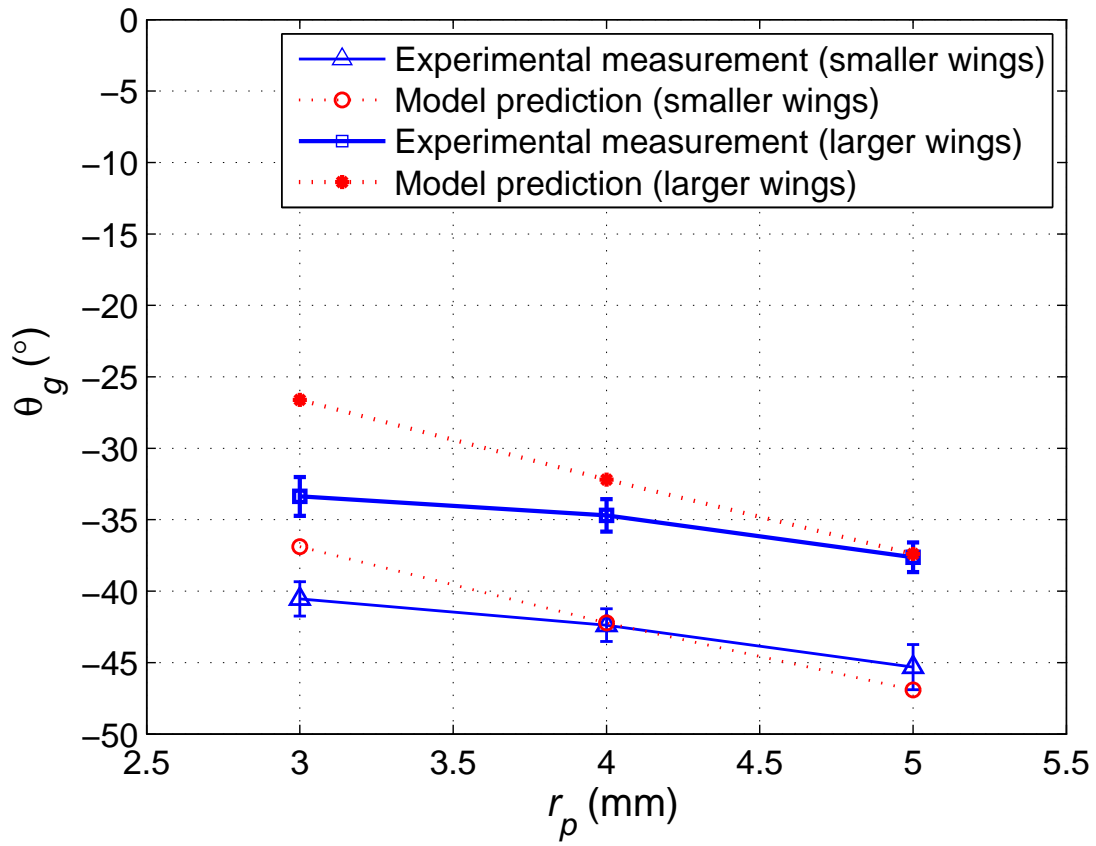
Figure 4.8: Glide angle of the miniature glider with respect to the movable mass displacement, with fixed net buoyancy of -20 g, when using two different sets of wings.
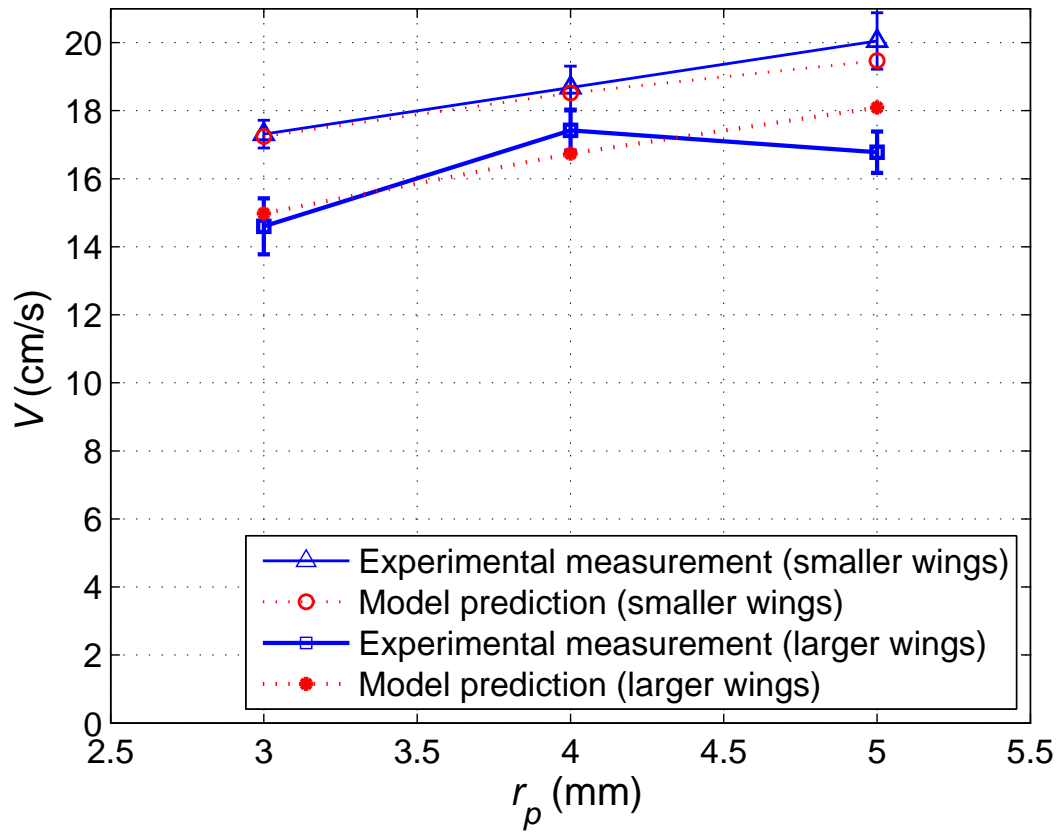
Figure 4.9: Total glide speed of the miniature glider with respect to the movable mass displacement, with fixed net buoyancy of -20 g, when using two different sets of wings.

Table 4.3: Comparison of our miniature glider with underwater gliders reported in the literature. Velocity refers to the terminal velocity. The bottom row shows the metrics of the glider reported in this dissertation (with the larger wings).

Name	Length	Weight	NetBuoyancy	Velocity
Slocum Electric Glider [47]	1.8 m	52 kg	520 g	0.4m/s
Spray Glider [47]	2 m	51 kg	900 g	0.45 m/s
Seaglider [47]	1.8 m	52 kg	840 g	0.45 m/s
ROGUE Lab Glider [73]	–	11.2 kg	360 g	–
ALBAC Glider [93]	1.4 m	45 kg	–	0.51 m/s
USM Glider [93]	1.3 m	–	–	–
ANT Littoral Glider [93]	2 m	120 kg	–	1.03 m/s
Miniature Glider (this work)	0.5 m	4.2 kg	20 g	0.17 m/s

The size of our miniature underwater glider is much smaller compared to traditional underwater gliders, since we expect the future gliding robotic fish to be lightweight, easy to carry, and operate in relatively shallow waters such as inland lakes and ponds. Table 4.3 shows some metrics of existing gliders and our miniature glider prototype. From Table 4.3, we can see that the ratio between net buoyancy and total weight of our miniature glider is similar to those of traditional underwater gliders, at the order of 1 %, while the velocity magnitude is less than 1/3 of those typical gliders. However, the cost performance index, the achieved horizontal travel distance per unit energy consumption, by our glider is over 18 times bigger than those of reported gliders, assuming they have similar gliding angles. All the above unique features of our miniature underwater glider are consistent with the scaling analysis in Section 4.3, which could be used to design the glider based on the specific applications.

Chapter 5

Steady Spiral in Three-Dimensional Space

5.1 Steady-State Spiraling Equations

If control inputs are fixed with nonzero tail angle, we can treat the influence of the tail on the hydrodynamic forces and moments as the effects of increased hydrodynamic angles (α, β), and we know that the gliding robotic fish will perform three-dimensional steady spiraling motion ([94], [81]), where the yaw angle ψ changes at constant rate while the roll angle ϕ and pitch angle θ are constants. Then $\boldsymbol{R}^T \boldsymbol{k}$ is constant since

$$\boldsymbol{R}^T \boldsymbol{k} = \boldsymbol{R}^T \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -\sin\theta \\ \sin\phi\cos\theta \\ \cos\phi\cos\theta \end{pmatrix} \tag{5.1}$$

Taking time derivative of $\boldsymbol{R}^T \boldsymbol{k}$, we have

$$\boldsymbol{\omega_b} \times (\boldsymbol{R}^T \boldsymbol{k}) = 0, \tag{5.2}$$

so the angular velocity has only one degree of freedom with ω_{3i} in Oz axis in the inertial frame. Then

$$\boldsymbol{\omega_b} = \omega_{3i}(\boldsymbol{R}^T \boldsymbol{k}) \tag{5.3}$$

The translational velocity in the body-fixed frame

$$\boldsymbol{v_b} = \boldsymbol{R}_{bv} \begin{bmatrix} V & 0 & 0 \end{bmatrix}^T \tag{5.4}$$

There are two important parameters in the spiraling motion: the turning radius R and the vertical speed V_{vertical}. By projecting the total velocity into the horizontal plane and vertical direction, we have

$$V_{\text{vertical}} = \boldsymbol{R}_{bv} \left(\begin{array}{ccc} V & 0 & 0 \end{array} \right)^T \left(\boldsymbol{R}^T \boldsymbol{k} \right) \tag{5.5}$$

$$R = \sqrt{V^2 - V_v^2} / \omega_{3i} \tag{5.6}$$

The steady-state spiraling equations are obtained by setting time derivatives to zero for the robot's dynamics:

$$0 = \boldsymbol{M} \boldsymbol{v_b} \times \boldsymbol{\omega_b} + m_0 g \boldsymbol{R}^T \boldsymbol{k} + \boldsymbol{F}_{ext} \tag{5.7}$$

$$0 = \boldsymbol{J} \boldsymbol{\omega_b} \times \boldsymbol{\omega_b} + \boldsymbol{M} \boldsymbol{v_b} \times \boldsymbol{v_b} + \boldsymbol{T}_{ext} + m_w g \boldsymbol{r_w} \times \left(\boldsymbol{R}^T \boldsymbol{k} \right) + \bar{m} g \boldsymbol{r_p} \times \left(\boldsymbol{R}^T \boldsymbol{k} \right) \tag{5.8}$$

From equations (3.1), (3.6), (5.3), (5.4) and the above steady-state spiraling equations, we know there are six independent states for describing the steady spiral motion: $\begin{bmatrix} \theta & \phi & \omega_{3i} & V & \alpha & \beta \end{bmatrix}$ with $\begin{bmatrix} m_0 & r_{p1} & \delta \end{bmatrix}$ as the three control inputs. Expanding equations (5.7) and (5.8), then transforming the original states to the above six independent states, we can obtain the nonlinear steady-

state spiraling equations as in (5.9) - (5.14).

$$0 = m_2 s\beta V c\phi c\theta \omega_{3i} - m_3 s\alpha c\beta V s\phi c\theta \omega_{3i} - m_0 gs\theta - 1/2\rho V^2 S(C_{D0} + C_D^\alpha \alpha^2 + C_D^\delta \delta^2)c\alpha c\beta$$

$$-1/2\rho V^2 S(C_{SF}^\beta \beta + C_{SF}^\delta \delta)c\alpha s\beta + 1/2\rho V^2 S(C_{L0} + C_L^\alpha \alpha)s\alpha \tag{5.9}$$

$$0 = -m_3 s\alpha c\beta V s\theta \omega_{3i} - m_1 c\alpha c\beta V c\phi c\theta \omega_{3i} - 1/2\rho V^2 S(C_{D0} + C_D^\alpha \alpha^2 + C_D^\delta \delta^2)s\beta$$

$$+1/2\rho V^2 S(C_{SF}^\beta \beta + C_{SF}^\delta \delta)c\beta + m_0 gs\phi c\theta \tag{5.10}$$

$$0 = m_1 c\alpha c\beta V s\phi c\theta \omega_{3i} + m_2 s\beta V s\theta \omega_{3i} + m_0 gc\phi c\theta - 1/2\rho V^2 S(C_{D0} + C_D^\alpha \alpha^2 + C_D^\delta \delta^2)s\alpha c\beta$$

$$-1/2\rho V^2 S(C_{SF}^\beta \beta + C_{SF}^\delta \delta)s\alpha s\beta - 1/2\rho V^2 S(C_{L0} + C_L^\alpha \alpha)c\alpha \tag{5.11}$$

$$0 = (J_2 - J_3)s\phi c\theta c\phi c\theta \omega_{3i}^2 + (m_2 - m_3)s\beta s\alpha c\beta V^2 + 1/2\rho V^2 S(C_{MR}^\beta \beta - K_{q1}s\theta \omega_{3i})c\alpha c\beta$$

$$-1/2\rho V^2 S(C_{M0} + C_{MP}^\alpha \alpha + K_{q2}s\phi c\theta \omega_{3i})c\alpha s\beta - m_w gr_w s\phi c\theta$$

$$-1/2\rho V^2 S(C_{MY}^\beta \beta + K_{q3}c\phi c\theta \omega_{3i} + C_{MY}^\delta \delta)s\alpha \tag{5.12}$$

$$0 = (J_1 - J_3)s\theta c\phi c\theta \omega_{3i}^2 + (m_3 - m_1)c\alpha c\beta s\alpha c\beta V^2 - m_w gr_w s\theta - \bar{m}gr_{p1}c\phi c\theta$$

$$+1/2\rho V^2 S(C_{MR}^\beta \beta - K_{q1}s\theta \omega_{3i})s\beta + 1/2\rho V^2 S(C_{M0} + C_{MP}^\alpha \alpha + K_{q2}s\phi c\theta \omega_{3i})c\beta \tag{5.13}$$

$$0 = (J_2 - J_1)s\theta s\phi c\theta \omega_{3i}^2 + (m_1 - m_2)c\alpha c\beta s\beta V^2 + 1/2\rho V^2 S(C_{MR}^\beta \beta - K_{q1}s\theta \omega_{3i})s\alpha c\beta$$

$$-1/2\rho V^2 S(C_{M0} + C_{MP}^\alpha \alpha + K_{q2}s\phi c\theta \omega_{3i})s\alpha s\beta + \bar{m}gr_{p1}s\phi c\theta$$

$$+1/2\rho V^2 S(C_{MY}^\beta \beta + K_{q3}c\phi c\theta \omega_{3i} + C_{MY}^\delta \delta)c\alpha \tag{5.14}$$

Here, we assume the mass matrix and inertia matrix have the following form:

$$M = \begin{pmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{pmatrix} \quad J = \begin{pmatrix} J_1 & 0 & 0 \\ 0 & J_2 & 0 \\ 0 & 0 & J_3 \end{pmatrix}$$

5.2 Computation of Spiral Path and Evaluation of Stability

5.2.1 Newton's method for Solving the Steady-State Spiraling Equations

The steady-state spiraling equations are highly nonlinear due to the terms involving trigonometric functions and inverse trigonometric functions. Given the angle of attack α, the sideslip angle β, and the velocity magnitude V, a recursive algorithm based on fixed-point iteration could potentially be applied to solve the equations for the other system states and control inputs [80]. However, we are more interested in the converse problem of how to calculate steady-state solutions given fixed control inputs, which are more useful for path planning and control purposes. Unfortunately, this problem does not admit analytical solutions and the convergence condition for the corresponding fixed-point problem is not satisfied. In the following we apply Newton's method to solve the problem.

Let $x = [\; \theta \quad \phi \quad \omega_{3i} \quad V \quad \alpha \quad \beta \;]^T$ be the six states that we want to solve for steady-state spiral gliding equations. And let $u = [\; m_0 \quad r_{p1} \quad \delta \;]^T$ be the three control inputs. For convenience of presentation, we write the governing equations in a compact form

$$0 = f(x, u) = [f_i(x, u)]_{i=1, \cdots, 6} \tag{5.15}$$

For example, f_1 is the right hand side of equation (5.9).

The iterative algorithm for Newton's method reads [95]

$$\hat{x}_{i+1} = \hat{x}_i - J^{-1}(\hat{x}_i, u) f(\hat{x}_i, u) \tag{5.16}$$

Here $\hat{x}_i$ is the ith-step iteration for the steady states, and $J(x, u)$ is the Jacobian matrix of $f(x, u)$

$$J(x, u) = \frac{\partial f}{\partial x} = \left[\frac{\partial f_i}{\partial x_j}\right]_{6 \times 6} \tag{5.17}$$

The first row elements of the Jacobian matrix are given in equations (5.18) - (5.23) while the others are omitted for succinct presentation, which can be calculated similarly.

$$\partial f_1 / \partial x_1 = -m_2 s\beta V c\phi s\theta \omega_{3i} + m_3 s\alpha c\beta V s\phi s\theta \omega_{3i} - m_0 gc\theta \tag{5.18}$$

$$\partial f_1 / \partial x_2 = -m_2 s\beta V s\phi c\theta \omega_{3i} - m_3 s\alpha c\beta V c\phi c\theta \omega_{3i} \tag{5.19}$$

$$\partial f_1 / \partial x_3 = m_2 s\beta V c\phi c\theta - m_3 s\alpha c\beta V s\phi c\theta \tag{5.20}$$

$$\partial f_1 / \partial x_4 = m_2 s\beta c\phi c\theta \omega_{3i} - m_3 s\alpha c\beta s\phi c\theta \omega_{3i} - m_0 gs\theta - \rho VS(C_{D0} + C_D^\alpha \alpha^2 + C_D^\delta \delta^2) c\alpha c\beta$$
$$- \rho VS(C_{SF}^\beta \beta + C_{SF}^\delta \delta) c\alpha s\beta + \rho VS(C_{L0} + C_L^\alpha \alpha) s\alpha \tag{5.21}$$

$$\partial f_1 / \partial x_5 = -m_3 c\alpha c\beta V s\phi c\theta \omega_{3i} - \rho V^2 SC_D^\alpha \alpha c\alpha c\beta + 1/2\rho V^2 S(C_{D0} + C_D^\alpha \alpha^2 + C_D^\delta \delta^2) s\alpha c\beta$$
$$+ 1/2\rho V^2 S(C_{SF}^\beta \beta + C_{SF}^\delta \delta) s\alpha s\beta + 1/2\rho V^2 SC_L^\alpha s\alpha$$
$$+ 1/2\rho V^2 S(C_{L0} + C_L^\alpha \alpha) c\alpha \tag{5.22}$$

$$\partial f_1 / \partial x_6 = m_2 c\beta V c\phi c\theta \omega_{3i} + m_3 s\alpha s\beta V s\phi c\theta \omega_{3i} + 1/2\rho V^2 S(C_{D0} + C_D^\alpha \alpha^2 + C_D^\delta \delta^2) c\alpha s\beta$$
$$- 1/2\rho V^2 SC_{SF}^\beta c\alpha s\beta - 1/2\rho V^2 S(C_{SF}^\beta \beta + C_{SF}^\delta \delta) c\alpha c\beta \tag{5.23}$$

Based on the parameters of the miniature underwater glider prototype as listed in Table 5.1, Newton's iterative formula is used to solve the steady-state spiraling equations. Characteristic parameters for steady spiraling motion, including the turning radius and ascending/descending speed, are obtained with different inputs as shown in Table 5.2. To apply Newton's method, the

59

Table 5.1: Parameters of the lab-developed underwater robot used in the steady-state spiraling equations.

Parameter	Value	Parameter	Value
m_1	3.88 kg	m_2	9.9 kg
m_3	5.32 kg	$\bar{m}$	0.8 kg
C_{D0}	0.45	C_D^α	17.59 rad^{-2}
C_{FS}^β	-2 rad^{-1}	C_{FS}^δ	1.5 rad^{-1}
C_{L0}	0.075	C_L^α	19.58 rad^{-1}
J_1	0.8 kg·m^2	J_2	0.05 kg·m^2
J_3	0.08 kg·m^2	C_{M0}	0.0076 m
C_{MR}^β	-0.3 m/rad	C_{MP}^α	0.57 m/rad
C_{MY}^β	5 m/rad	C_{MY}^δ	-0.2 m/rad
K_{q1}	-0.1 m·s/rad	K_{q2}	-0.5 m·s/rad
K_{q3}	-0.1 m·s/rad	S	0.012 m^2

initial values of states for iteration are chosen to be $\theta = -10°$, $\phi = -10°$, $\omega_{3i} = 0.1$ rad/s, $V = 0.3$ m/s, $\alpha = 0°$, $\beta = 0°$. From the calculated results, we can see that a small turning radius requires a large tail angle, a large displacement of movable mass, and a small net buoyancy, while a low descending or ascending speed demands a small tail angle, a small displacement of movable mass, and a medium net buoyancy.

5.2.2   Region of Convergence for Newton's Method

For Newton's method, the choice of the initial condition is important to the convergence of the algorithm. Here, we numerically explore the region of convergence. For a fixed set of control inputs, e.g., $r_{p1} = 5$ mm, $m_0 = 30$ g and $\delta = 30°$, we carry out the convergence test by running the algorithm starting from different initial values of the states, and record whether a given initial condition leads to convergence. Fig. 5.1a shows the convergence test results for different initial conditions of the roll angle ϕ, pitch angle θ and spiraling speed V while the initial values of the

Table 5.2: Computed spiraling steady states through Newton's method.

m_0 (g)	r_{p1} (cm)	$\delta(^\circ)$	$(\theta,\phi,\omega_{3i},V,\alpha,\beta)$ ($^\circ,^\circ$,rad/s,m/s,$^\circ,^\circ$)	(V_{vertical},R) (m/s,m)
25	0.3	45	(-44.5, -31.0, 0.425, 0.264, -0.914, 4.10)	(0.182, 0.450)
25	0.4	45	(-46.8, -36.6, 0.448, 0.267, -1.32, 4.52)	(0.190, 0.417)
25	0.5	45	(-48.3, -40.6, 0.464, 0.268, -1.61, 4.87)	(0.195, 0.396)
25	0.6	45	(-49.3, -43.8, 0.476, 0.267, -1.84, 5.18)	(0.197, 0.380)
25	0.7	45	(-50.2, -46.5, 0.486, 0.267, -2.04, 5.48)	(0.211, 0.338)
10	0.5	45	(-70.8, -49.3, 0.589, 0.184, -3.64, 7.36)	(0.169, 0.121)
15	0.5	45	(-63.5, -52.7, 0.571, 0.218, -3.30, 6.98)	(0.189, 0.190)
20	0.5	45	(-55.5, -47.8, 0.517, 0.247, -2.46, 5.85)	(0.197, 0.287)
30	0.5	45	(-42.1, -34.3, 0.423, 0.281, -0.901, 4.24)	(0.185, 0.500)
35	0.5	45	(-36.9, -29.3, 0.392, 0.289, -0.306, 3.85)	(0.172, 0.591)
40	0.5	45	(-32.3, -25.3, 0.368, 0.293, 0.224, 3.60)	(0.157, 0.670)
25	0.5	30	(-37.6, -11.9, 0.235, 0.242, 0.854, 2.19)	(0.151, 0.806)
25	0.5	35	(-43.4, -20.7, 0.311, 0.258, 0.0698, 2.87)	(0.178, 0.602)
25	0.5	40	(-46.8, -31.2, 0.389, 0.266, -0.761, 3.77)	(0.192, 0.474)
25	0.5	50	(-49.2, -48.8, 0.537, 0.264, -2.54, 6.19)	(0.192, 0.337)
25	0.5	55	(-51.1, -56.4, 0.615, 0.257, -3.62, 7.86)	(0.190, 0.283)
25	0.5	60	(-55.0, -63.8, 0.705, 0.247, -4.95, 10.0)	(0.189, 0.225)

other three states are set as $\alpha = 0^\circ, \beta = 0^\circ, \omega_{3i} = 0.1$ rad/s; Fig. 5.1b shows the results for different initial conditons of the angle of attack α, sideslip angle β and the angular speed ω_{3i} with the initial values of the other three states set as $\phi = -10^\circ, \theta = -10^\circ, V = 0.3$ m/s. From the results, we see that a small roll angle, a small pitch angle and a large velocity in the reasonable range will lead to convergence; and the signs of the angle of attack and sideslip angle are very important to the convergence of the solution. These observations offer insight into how to properly choose the initial conditions when running the Newton's method to obtain the steady spiraling path; for example, one may want to select zero degree for the initial values of the angle of attack and sideslip angle when having no idea about the signs of those two variables.
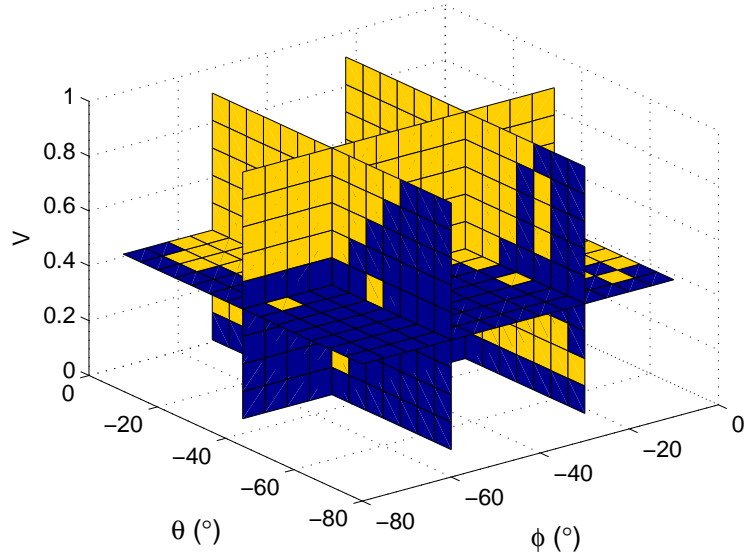
(a) Convergence with respect to the roll angle, pitch angle and spiraling speed
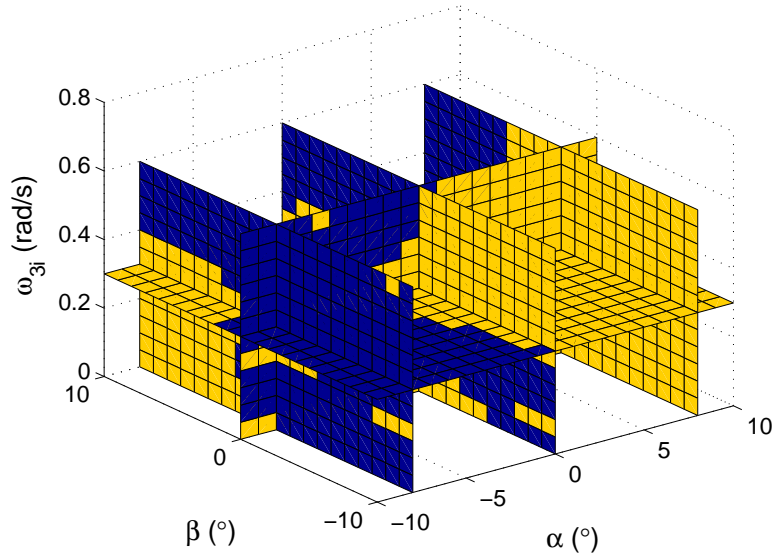
(b) convergence with respect to the angle of attack, sideslip angle and angular speed

Figure 5.1: Convergence test results for Newton's method with respect to initial conditions. Color yellow (light) means that convergence to the steady-state spiraling equilibrium is achievable with the corresponding initial values; color blue (dark) means that there is no convergent solution or the convergent solution is not at the steady-state spiraling equilibrium. In the test, the used set of control inputs is $r_{p1} = 5$ mm, $m_0 = 30$ g, $\delta = 45°$; and the corresponding equilibrium state values are $\theta = -42.1281°$, $\phi = -34.2830°$, $\omega_{3i} = 0.4229$ rad/s, $V = 0.2809$ m/s, $\alpha = -0.9014°$, $\beta = 4.2414°$.

5.2.3 Stability Analysis of Spiraling Motion

It is of interest to understand the stability of the spiral motion under a given set of control inputs. Global stability analysis, however, is very difficult if not impossible due to the highly nonlinear dynamics of the system. In this subsection we perform local stability analysis for the steady spiraling motion obtained from (5.9)–(5.14). A solution to these equations can be considered a relative equilibrium of the system since it is independent of the coordinates of the robot in the inertial frame. We denote with $J_d(x_d, u)$ the Jacobian matrix for the dynamics (3.4) and (3.5), where in a compact form the system state vector is represented as $x_d = [\ v_b\quad \omega_b\ ]^T$ and the system dynamics as $\dot{x}_d = f_d(x_d, u)$. As the (relative) equilibrium point is computed using a different set of system states $x = [\ \theta\quad \phi\quad \omega_{3i}\quad V\quad \alpha\quad \beta\ ]^T$, and the Jacobian matrix $J(x, u)$ for the steady-state equations has been evaluated with those states (Section 5.2.1), we can evaluate J_d through the chain rule

$$J_d(x_d, u) = \begin{pmatrix} M^{-1} & 0 \\ 0 & J^{-1} \end{pmatrix} J(x, u) \left(\frac{\mathrm{d}x_d}{\mathrm{d}x} \right)^{-1} \tag{5.24}$$

where $\dfrac{\mathrm{d}x_d}{\mathrm{d}x} = h_{\mathrm{spiral}}(x)_{6\times 6} =$

$$\begin{pmatrix} 0 & 0 & 0 & c\alpha c\beta & -Vs\alpha c\beta & -Vc\alpha s\beta \\ 0 & 0 & 0 & s\beta & 0 & Vc\beta \\ 0 & 0 & 0 & s\alpha c\beta & Vc\alpha c\beta & -Vs\alpha s\beta \\ -c\theta\omega_{3i} & 0 & -s\theta & 0 & 0 & 0 \\ -s\phi s\theta\omega_{3i} & c\phi c\theta\omega_{3i} & s\phi c\theta & 0 & 0 & 0 \\ -c\phi s\theta\omega_{3i} & -s\phi c\theta\omega_{3i} & c\phi c\theta & 0 & 0 & 0 \end{pmatrix}$$

So the value of linearization matrix J_d at the equilibrium point can be obtained by just plugging the steady state values x_e computed in Section 5.2.1 into the above equation, and by checking the

Hurwitz property of the linearization matrix J_d, we can understand the local stability property of the steady-state spiraling motion.

We test the listed steady-state spiral motions in Table 5.2 for local stability. For example, for the steady spiral corresponding to the control input set $m_0 = 25$ g, $r_{p1} = 0.3$ cm, and $\delta = 45°$, the eigenvalues of Jacobian matrix J_d are $-0.91 \pm 5.02i, -6.69, -2.09, -0.42, -0.090$, which shows exponential stability. We find that all spirals in Table 5.2 have a Hurwitz Jacobian matrix and thus the equilibrium of each spiral is locally asymptotically stable.

5.2.4   Basins of Attraction for the Spiraling Dynamics

The analysis in the previous subsection suggests that the relative equilibria associated with steady spiraling are locally stable. It is of interest to gain some insight into the sizes of basins of attraction for those equilibria. In this subsection, we run the dynamics simulation starting from different initial conditions for a given fixed control input, and then record whether each initial state configuration will lead to convergence to steady spiraling, and if yes, what is the approximate time it takes to converge. Since one cannot visualize a state space of more than three dimensions, we have chosen to visualize the basin of attraction in three-dimensional subspaces of the original state space.

Fig. 5.2 shows the simulation results of convergence time based on the parameters of our experimental prototype with respect to three states ϕ, θ, v_1. To obtain the results shown in this figure, the control inputs are given as $r_{p1} = 5$ mm, $m_0 = 30$ g and $\delta = 45°$. Following this simulation method, we can get the basins of attraction with convergence time for any other set of control inputs. Similarly, we can obtain the convergence test results, shown in Fig. 5.3, when we vary the initial conditions for the angular velocities in the body-fixed frame. From the results, it seems that the basin of attractions for the spiraling dynamics is very large, which means that, starting
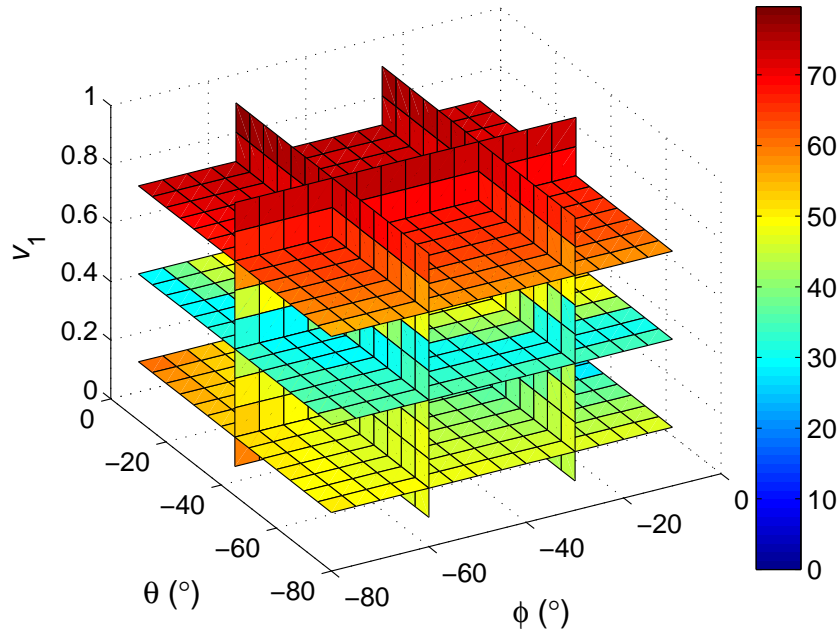
(a) Display in orthogonal slice planes
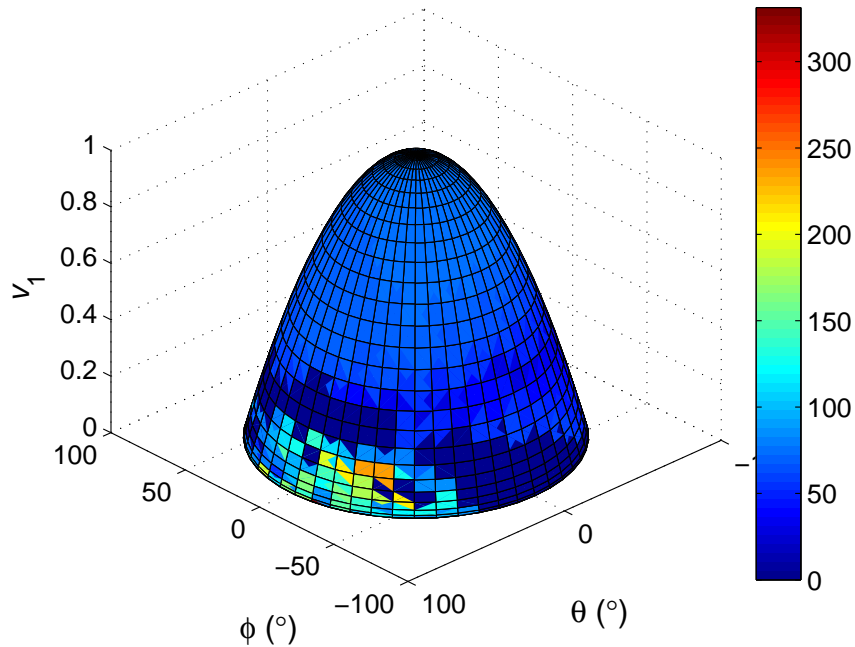
(b) Display in a half sphere surface

Figure 5.2: Convergence time in seconds for spiraling dynamics with respect to different initial values of states in roll angle ϕ, pitch angle θ, and translational velocity v_1 along the Ox_b direction, for the control inputs of $r_{p1} = 5$ mm, $m_0 = 30$ g and $\delta = 45°$. The corresponding equilibrium state values are $\theta = -42.1281°$, $\phi = -34.2830°$, and $v_1 = 0.2801$ m/s.
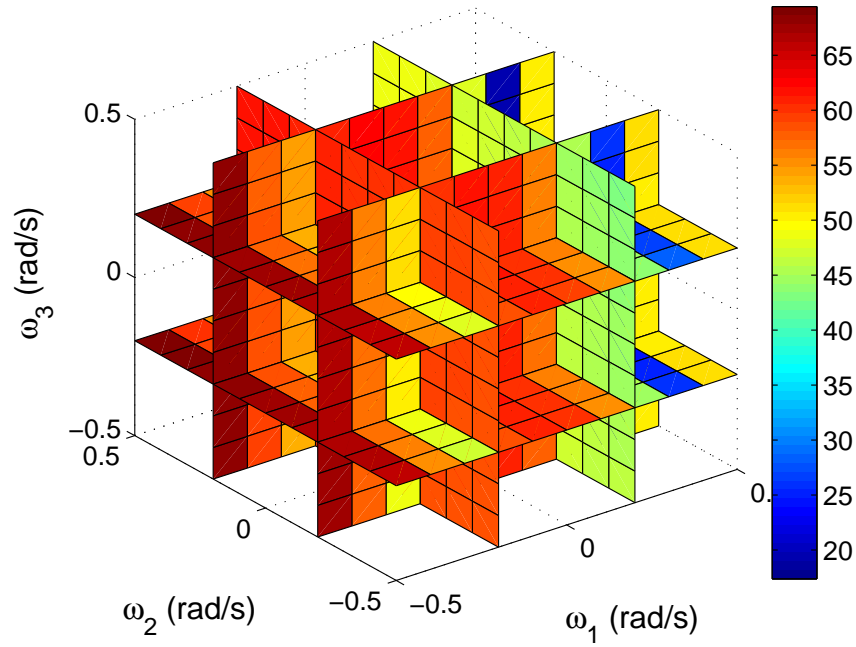
Figure 5.3: Convergence time in seconds for spiraling dynamics with respect to different initial values of states in angular velocities, for the control inputs of $r_{p1} = 5$ mm, $m_0 = 30$ g and $\delta = 45°$, displayed in orthogonal slice planes. The corresponding equilibrium state values are $\omega_1 = 0.2837$ rad/s, $\omega_2 = -0.1767$ rad/s, and $\omega_3 = 0.2592$ rad/s.
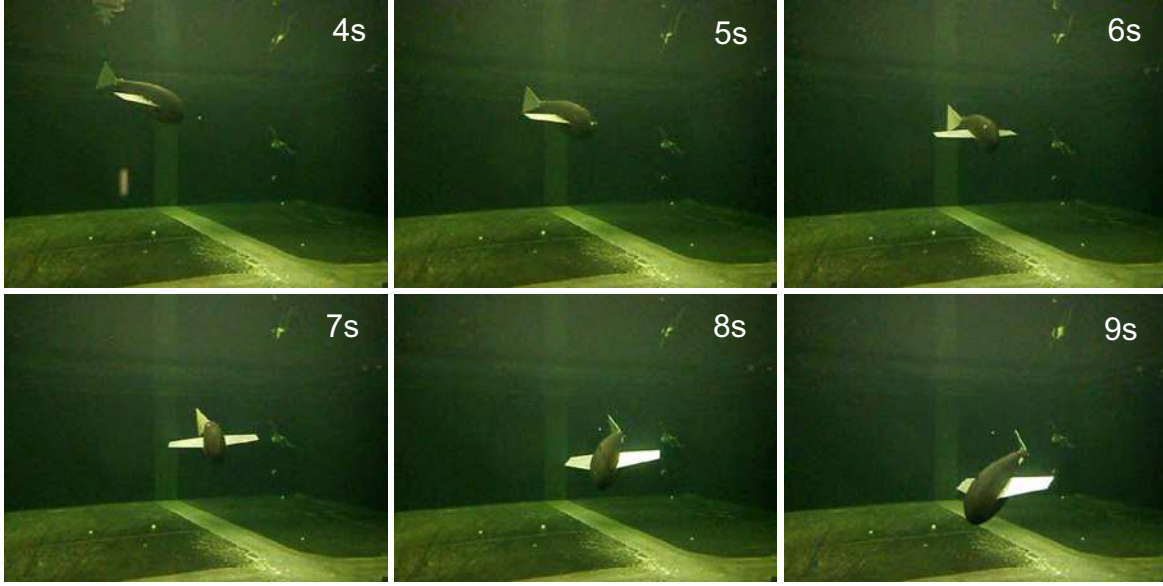
Figure 5.4: Snap shots of the robot spiraling in the experiment tank.

from almost every state configuration in a reasonable state value range, the glider is able to achieve the steady spiraling motion eventually. However, we also notice that the convergence time varies significantly with the starting condition. When the pitch angle and roll angle are negative, and the speed is neither too high nor too low, the convergence time is relatively short. This provides us with some ideas about when to switch to a desired gliding profile and how long we expect for the transient period. We also notice that among all three angular velocity states, only the initial condition of ω_1 takes a noticeable influence on the convergence time of the glider dynamics. This is consistent with the slow dynamics of the rotation motion in Ox_b direction due to the enhanced inertia from the large wings.

5.3 Experimental Results

With the miniature underwater glider prototype featuring a swappable tail fin, experiments are conducted in order to confirm the spiraling motion and validate the derived mathematical model.

67

The experiments are performed in a large water tank that measures 15-foot long, 10-foot wide, and 4-foot deep, as shown in Fig. 5.4. We set the net buoyancy (negatively buoyant), the linear actuator position and the tail angle to fixed values. Then the glider is released on the water surface and enters into the spiraling mode. Cameras are set to record the videos in both top view and side view. The turning radii of the spirals are extracted after video processing. The comparison between model predictions and experiment results on turning radius for different tail angles and different excess masses are shown in Fig. 5.5 and Fig. 5.6, respectively. From the results, we can see that the turning radius of the spiral is smaller with a larger deflected tail angle and a smaller net buoyancy. The error bar in the figures shows the average value and standard deviation of the turning radius out of ten repeated experiments. The model prediction shares the same trend with the experimental results, and generally speaking, the match between those two are good.

There are some factors contributing to the measurement errors. First, the scales of camera images are different at different distances. Here, an average scale is used in the information acquisition during video processing. A grid board is used for calibration, captured with the camera at the average distance. Second, there are some initial transient processes, which is difficult to be completely separate from the steady spiraling period. Experimental environment with deeper water will effectively reduce the influence of initial transient; however, the complexity of experiments setup will be increased as a result. The environmental disturbances such as currents will also affect the experimental results. So with these uncertainties, we consider the match between our experimental results and the model predictions satisfactory.

68
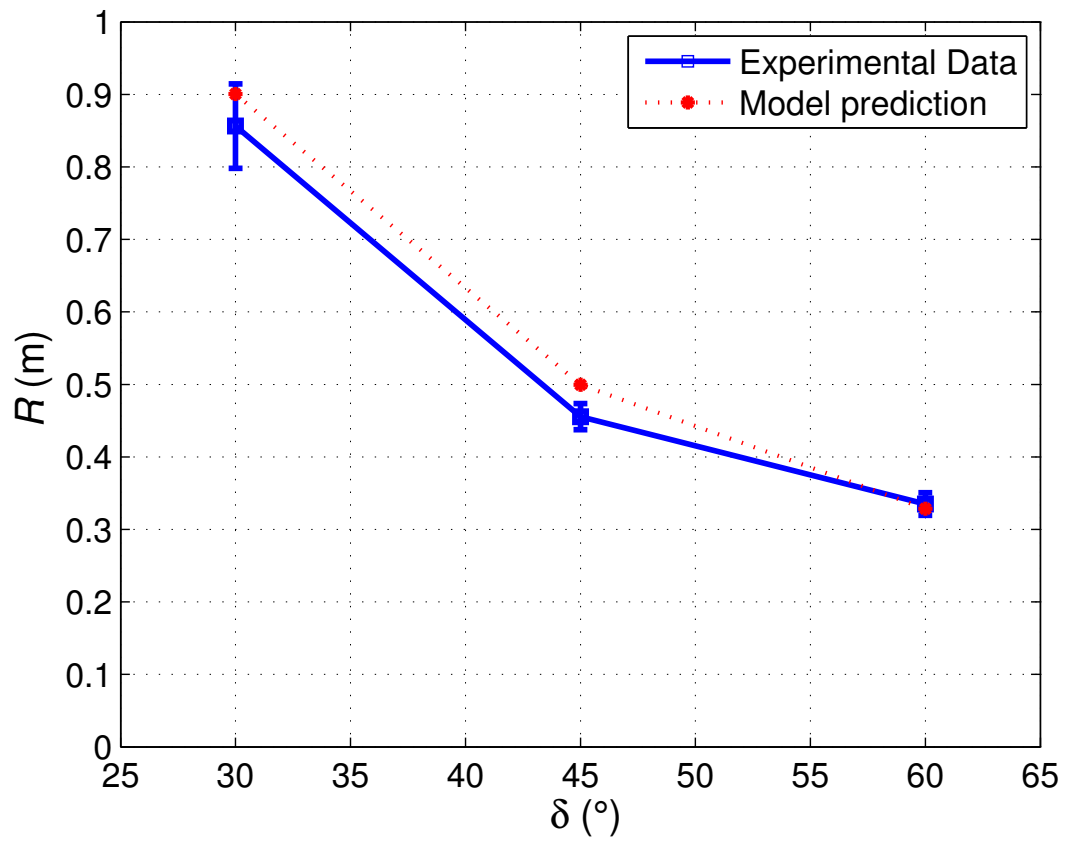
Figure 5.5: Spiraling radius with respect to the tail angle, with fixed movable mass displacement of 0.5 cm and fixed excess mass of 30 g.
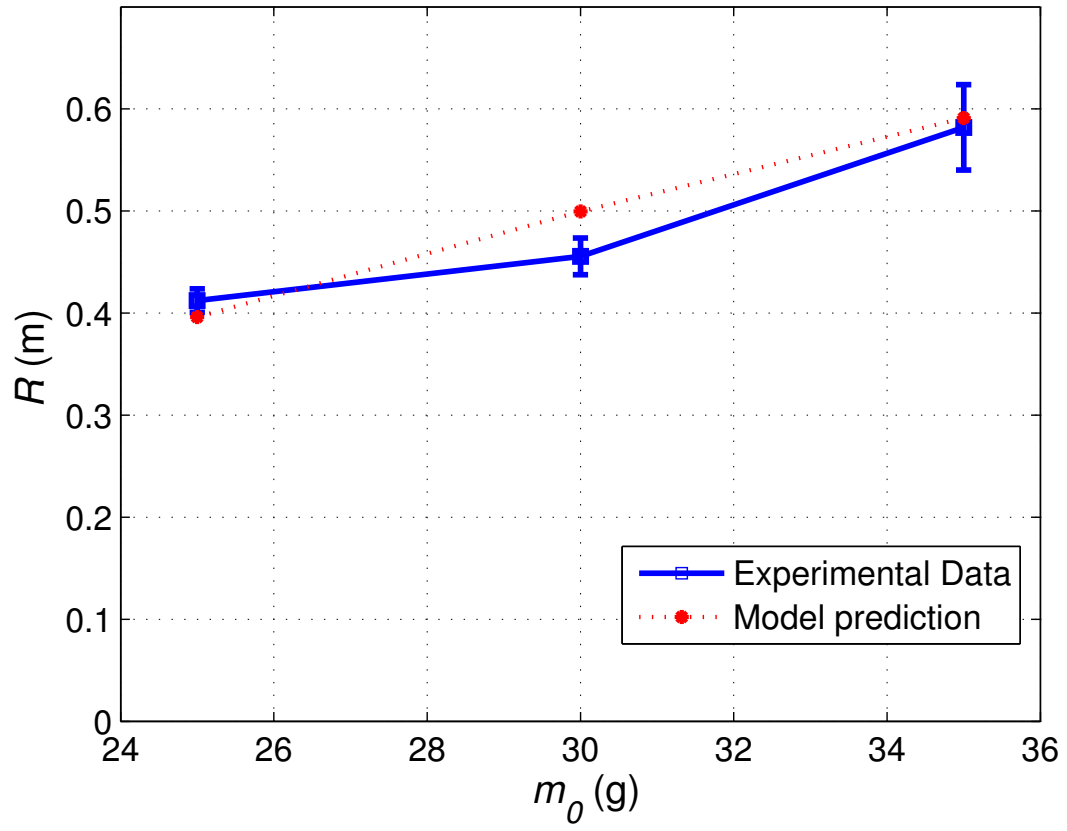
Figure 5.6: Spiraling radius with respect to the excess mass , with fixed movable mass displacement of 0.5 cm and fixed tail angle of 45°.

Chapter 6

Passivity-based Stabilization with a Whale-type Tail

In this chapter, the sagittal-plane stabilization problem of a gliding robotic fish with a whale-type tail is discussed. The dynamics of an underwater glider in the sagittal plane is first reviewed and separated into the slow dynamics and fast dynamics based on singular perturbation analysis. In Section 6.2, a passivity-based nonlinear controller for the approximated reduced model is proposed, and stability analysis for the full system is conducted via linearization. Simulation results are presented to show the effectiveness of the designed controller. A nonlinear observer is then proposed in Section 6.3 for the implementation of the control strategy. In Section 6.4, both open-loop and closed-loop experimental results are presented using a gliding robotic fish to illustrate the effectiveness of both the controller and the observer.

6.1 Model of a Gliding Robotic Fish with a Whale-type Tail

In this section, the effect of a whale-type tail of a gliding robotic fish is considered as a control surface. A dynamic model for underwater gliders in the sagittal plane is first reviewed [76, 78], which is an invariant plane for such robots. Then with singular perturbation analysis, the model is separated into two subsystems, a fast-mode system and a slow-mode system, for controller design in Section 6.2.
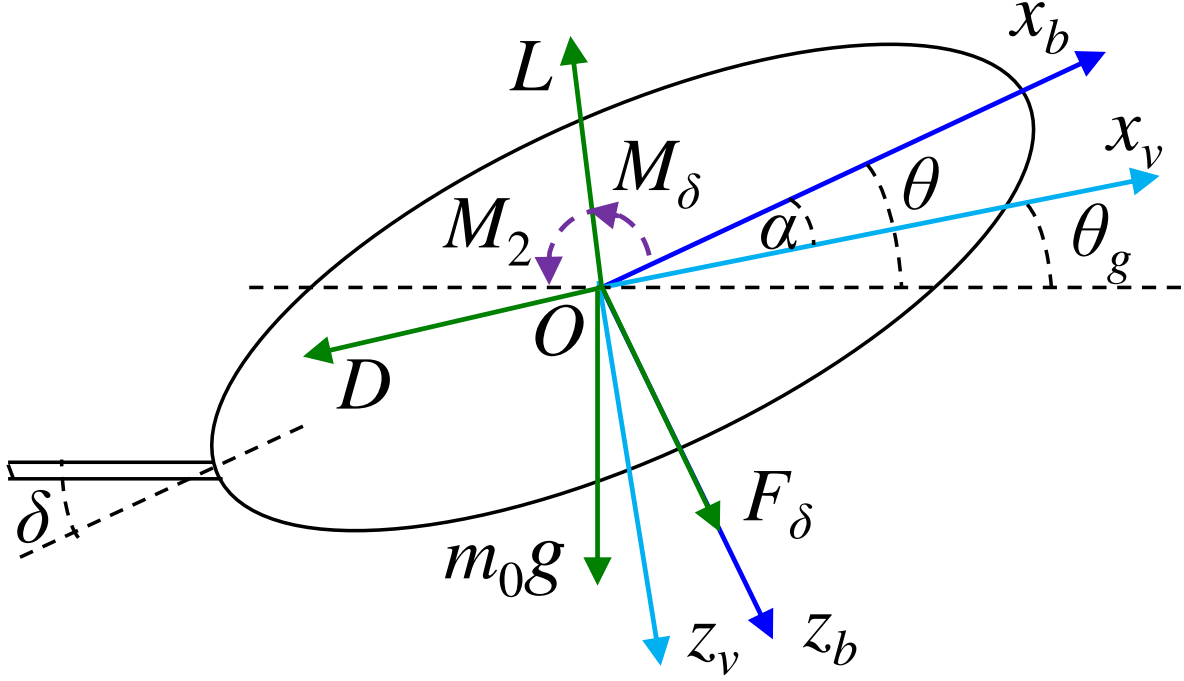
Figure 6.1: The schematic of a gliding robotic fish with forces and moments defined in the corresponding coordinate frames (side view).

6.1.1 Dynamic Model in the Sagittal Plane

This dissertation is focused on the motion in the robot's invariant plane. As shown in Fig. 6.1, a gliding robotic fish is modeled as a rigid-body system, with an ellipsoidal shape and fixed wings as typically reported in the underwater glider literature. The relevant coordinate reference frames are defined as follows: the body-fixed reference frame, denoted as $Ox_b y_b z_b$, has its origin O at the geometric center. The Ox_b axis is along the body's longitudinal axis pointing to the head; the Oz_b axis is perpendicular to Ox_b axis in the sagittal plane of the robot pointing downwards, and Oy_b axis will be automatically formed by the right-hand orthonormal principle. In the velocity reference frame $Ox_v y_v z_v$, Ox_v axis is along the direction of velocity whose magnitude is V, and Oz_v lies in the sagittal plane perpendicular to Ox_v. In the inertial frame $Axyz$ (not shown in Fig. 6.1), Az axis points in the gravitational acceleration direction, and Ax is defined in the same direction as the intersection line of the horizontal plane and the sagittal plane, while the origin A is a fixed point

in space. There are three angles defined following the standard convention in marine applications, based on the aforementioned reference frames. The pitch angle θ is the angle between Ox_b and Ax with nose up as positive; the gliding angle θ_g is the angle between Ox_v and Ax with gliding up as positive; the angle of attack α is the angle from Ox_b axis to Ox_v axis with rotation axis Oy_b.

We define the sum of the mass of the gliding robotic fish, m_g, and the added mass in Ox_b direction as m_1, and similarly, the sum of m_g and the added mass in Oz_b direction as m_3. The robot displaces a volume of water of mass m_w. Let $m_0 = m_g - m_w$ represent the excess mass (negative buoyancy).

The forces acting on the robot body include gravitational force, buoyancy force, hydrodynamic forces (lift and drag) and control force. Due to the symmetric shape of the robot, the center of buoyancy will be through the origin O. The assumptions in [76] are taken that the movable mass is fixed at the origin O (during steady gliding), with the stationary mass distributed uniformly, and the added masses are equally valued ($m_1 = m_3 = m$). Then the center of gravity will coincide with the center of buoyancy at the origin. The force pair, gravitational force and buoyancy force, act like one force of excess mass m_0 at the origin O in Az direction. The hydrodynamic lift force L is along negative Oz_v axis, while the drag force D is along negative Ox_v direction. The control force F_δ is in Oz_b direction, exerted by the control surface (e.g., a whale fluke-type tail) traveling through the fluid medium, which is essentially another hydrodynamic force. The control surface angle δ is defined as the angle between the control surface plane and the Ox_by_b plane. The hydrodynamic

forces are dependent on the angle of attack and the velocity as follows:

$$L = (K_{L0} + K_L \alpha) V^2 \tag{6.1}$$

$$D = \left(K_{D0} + K_D \alpha^2 \right) V^2 \tag{6.2}$$

$$F_\delta = K_{F_\delta} V^2 u_\delta \tag{6.3}$$

where K_{L0}, K_L are lift coefficients, and K_{D0}, K_D are drag coefficients. u_δ is the effective angle of attack that the control surface contributes to the gliding robotic fish. There is a linear relationship between u_δ and the control surface angle δ, $u_\delta = K_{u_\delta} \delta$, where K_{u_δ} is a scale constant. K_{F_δ} is the coupling factor that describes the additional force that the control surface induces.

There are two moments about the Oy_b axis, which rotate the robot to a specific attitude. One is the hydrodynamic pitch moment M_2, and the other is the control moment M_δ. They are modeled as

$$M_2 = \left(K_{M0} + K_M \alpha + K_{q2} \omega_2 \right) V^2 \tag{6.4}$$

$$M_\delta = -K_M u_\delta V^2 \tag{6.5}$$

where K_{M0} and K_M are pitch moment coefficients, K_q is the pitching damping coefficient and ω_2 is the angular velocity for the pitch.

By applying Newton's second law and the moment of momentum equation, the gliding robotic

fish dynamics are obtained as

$$\dot{V} = -\frac{1}{m_1}\left(m_0 g \sin\theta_g + D - F_\delta \sin\alpha\right) \tag{6.6}$$

$$\dot{\theta}_g = \frac{1}{m_1 V}\left(-m_0 g \cos\theta_g + L + F_\delta \cos\alpha\right) \tag{6.7}$$

$$\dot{\alpha} = \omega_2 - \frac{1}{m_1 V}\left(-m_0 g \cos\theta_g + L + F_\delta \cos\alpha\right) \tag{6.8}$$

$$\dot{\omega}_2 = \frac{1}{J_2}\left(K_{M0} + K_M\alpha + K_{q2}\omega_2 - K_M u_\delta\right)V^2 \tag{6.9}$$

where J_2 is the total inertia about Oy_b axis, consisting of stationary mass inertia and added inertia in water, and g represents the gravitational acceleration.

For the open-loop system (i.e., $u_\delta = 0$), the steady gliding profile can be obtained from (6.6)-(6.9). The state variables at the equilibrium have the following relationships

$$\theta_{ge} = \arctan\frac{-K_{De}}{K_{Le}}, \qquad \alpha_e = -\frac{K_{M0}}{K_M}, \qquad \omega_{2e} = 0, \qquad V_e = \left(\frac{|m_0|g}{\sqrt{K_{De}^2 + K_{Le}^2}}\right)^{\frac{1}{2}}$$

where $K_{De} = K_{D0} + K_D\alpha_e^2, K_{Le} = K_{L0} + K_L\alpha_e$.

6.1.2   System Reduction via Singular Perturbation

Bhatta and Leonard [76] have shown with singular perturbation analysis that for the open-loop system, the dynamic model can be reduced to a second-order system with good approximation,

and the corresponding non-dimensional full state model is:

$$\frac{d\bar{V}}{dt_n} = -\frac{1}{K_{De}V_e^2}\left(m_0 g \sin\left(\bar{\theta}_g + \theta_{ge}\right) + D - F_\delta \sin\left(\bar{\alpha} + \alpha_e\right)\right) \tag{6.10}$$

$$\frac{d\bar{\theta}_g}{dt_n} = \frac{1}{K_{De}V_e^2\left(1+\bar{V}\right)}\left(-m_0 g \cos\left(\bar{\theta}_g + \theta_{ge}\right) + L + F_\delta \cos\left(\bar{\alpha} + \alpha_e\right)\right) \tag{6.11}$$

$$\varepsilon_1\frac{d\bar{\alpha}}{dt_n} = \bar{\omega}_2 - \varepsilon_1\frac{1}{K_{De}V_e^2\left(1+\bar{V}\right)}\left(-m_0 g \cos\left(\bar{\theta}_g + \theta_{ge}\right) + L + F_\delta \cos\left(\bar{\alpha} + \alpha_e\right)\right) \tag{6.12}$$

$$\varepsilon_2\frac{d\bar{\omega}_2}{dt_n} = -\left(\bar{\alpha} + \bar{\omega}_2 - u_\delta\right)\left(1+\bar{V}\right)^2 \tag{6.13}$$

where the new state variables are defined as

$$\bar{V} = \frac{V - V_e}{V_e}, \qquad \bar{\theta}_g = \theta_g - \theta_{ge}, \qquad \bar{\alpha} = \alpha - \alpha_e, \qquad \bar{\omega}_2 = \frac{K_q}{K_M}\omega_2$$

the non-dimensional time t_n and some related constants are defined as

$$\tau_s = \frac{m_3}{K_{De}V_e}, \qquad \varepsilon_2 = -\frac{J_2}{K_q V_e^2}\frac{1}{\tau_s}, \qquad t_n = t/\tau_s, \qquad \varepsilon_1 = \frac{K_q}{K_M}\frac{1}{\tau_s}$$

For the new state model, the hydrodynamic forces and moment can be described as

$$D = \left(K_{D0} + K_D\left(\bar{\alpha} + \alpha_e\right)^2\right)V_e^2\left(1+\bar{V}\right)^2 \tag{6.14}$$

$$L = \left(K_{L0} + K_L\left(\bar{\alpha} + \alpha_e\right)\right)V_e^2\left(1+\bar{V}\right)^2 \tag{6.15}$$

$$M_2 = \left(K_{M0} + K_M\alpha + K_q\bar{\omega}_2\right)V_e^2\left(1+\bar{V}\right)^2 \tag{6.16}$$

$$F_\delta = K_{F_\delta}u_\delta V_e^2\left(1+\bar{V}\right)^2 \tag{6.17}$$

76

The system can be further written in a compact form

$$\frac{d\xi}{dt_n} = f(\xi, \eta, u_\delta) \tag{6.18}$$

$$\mu \frac{d\eta}{dt_n} = Ag(\xi, \eta, \varepsilon, u_\delta) \tag{6.19}$$

where,

$$\xi = \begin{bmatrix} \bar{V} \\ \bar{\theta}_g \end{bmatrix}, \quad \eta = \begin{bmatrix} \bar{\alpha} \\ \bar{\omega}_2 \end{bmatrix}, \quad f = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, \quad g = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix},$$

$$A = \begin{bmatrix} \frac{\mu}{\varepsilon_1} & 0 \\ 0 & \frac{\mu}{\varepsilon_2} \end{bmatrix}, \quad \varepsilon = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \end{bmatrix}, \quad \mu = \max(\varepsilon_1, \varepsilon_2),$$

and f and g are defined accordingly based on (6.10) - (6.13).

From singular perturbation analysis, by setting $\mu = 0$, we arrive at $\bar{\omega}_2 = 0$ and $\bar{\alpha} = u_\delta$. Plugging those two fast-mode states into the other two state equations, the reduced model for the full system is obtained. Now we further set $\bar{\alpha} = 0$ in the reduced model for design convenience, since $\bar{\alpha}$ is relatively small in value. Then the approximation of the reduced model can be expressed as

$$\frac{d\xi}{dt_n} = f(\xi, 0, u_\delta) \tag{6.20}$$

and this second-order system will be used in the controller design. The effectiveness of designing the controller based on the approximated reduced system for the original full system, will be demonstrated in next section.

6.2 Passivity-based Controller Design

6.2.1 Passivity-based Controller for the Approximated Reduced Model

The open-loop reduced model (6.20) with $u_\delta = 0$ has an exponentially stable equilibrium point at the origin, which can be proven by Lyapunov analysis with the following positive definite Lyapunov function [76]

$$\Phi = \frac{2}{3} - (1 + \bar{V}) \cos \bar{\theta}_g + \frac{1}{3}(1 + \bar{V})^3 \tag{6.21}$$

and $\frac{\partial \Phi}{\partial \xi} f(\xi, 0, 0) \leq -b_1 \|\xi\|$ with $b_1 > 0$.

Now the objective is to design a feedback controller to stabilize the origin of the approximated reduced model, which also provides a faster convergence speed. The approximated reduced system is linear in control

$$\frac{d\xi}{dt_n} = f(\xi, 0, 0) + g_r(\xi) u_\delta \tag{6.22}$$

where

$$g_r(\xi) = \begin{bmatrix} K_{F_\delta}(1 + \bar{V})^2 \sin \alpha_e / K_{D_e} \\ K_{F_\delta}(1 + \bar{V}) \cos \alpha_e / K_{D_e} \end{bmatrix} \tag{6.23}$$

For passivity-based controller design, an output y_r needs to be defined for the approximated reduced system, to make the system passive [96]. The output is chosen as

$$y_r = \frac{\partial \Phi}{\partial \xi} g_r(\xi) \tag{6.24}$$

where

$$\frac{\partial \Phi}{\partial \xi} = \begin{bmatrix} \frac{\partial \Phi}{\partial \bar{V}} & \frac{\partial \Phi}{\partial \bar{\theta}_g} \end{bmatrix}$$

$$= \begin{bmatrix} -\cos \bar{\theta}_g + (1+\bar{V})^2 & (1+\bar{V})\sin \bar{\theta}_g \end{bmatrix} \qquad (6.25)$$

We check the following expression for the approximated reduced model,

$$\frac{d\Phi}{dt_n} = \frac{\partial \Phi}{\partial \xi}(f(\xi,0,0) + g_r(\xi)u_\delta)$$

It is known that

$$\frac{\partial \Phi}{\partial \xi} f(\xi,0,0) \le 0$$

So

$$\frac{d\Phi}{dt_n} \le u_\delta y_r$$

Then by the definition of a passive system, the following system

$$\begin{cases} \dfrac{d\xi}{dt_n} = f(\xi,0,u_\delta) \\ \\ y_r = \dfrac{\partial \Phi}{\partial \xi} g_r(\xi) \end{cases} \qquad (6.26)$$

is passive. Let control u_δ for system (6.22) be

$$u_\delta = -\phi(y_r) \qquad (6.27)$$

for some function ϕ, where $y_r u_\delta = -y_r \phi(y_r) \le 0$.

Now we take Φ in (6.21) as the Lyapunov function for the closed-loop system (6.26). Then

$$\frac{d\Phi}{dt_n} = \frac{\partial \Phi}{\partial \xi}\left(f(\xi,0,0) + g_r(\xi)u_\delta\right)$$

$$= \frac{\partial \Phi}{\partial \xi}f(\xi,0,0) + \frac{\partial \Phi}{\partial \xi}g_r(\xi)u_\delta$$

$$\leq -b_1\|\xi\| + y_r u_\delta$$

For the robot controller design, there is limitation on the magnitude of the control variable u_δ, so in this dissertation, we take

$$\phi(y_r) = \frac{1}{K_c}\arctan(y_r) \tag{6.28}$$

where K_c is the control parameter that is used to limit the control output magnitude. We then have

$$\frac{d\Phi}{dt_n} \leq -b_1\|\xi\| - \frac{1}{K_c}y_r\arctan(y_r) \tag{6.29}$$

which proves the asymptotic stability of the origin. Furthermore, the additional negative term $-\frac{1}{K_c}y_r\arctan(y_r)$ in the derivative of Lyapunov function provides an extra stabilization advantage. With that term, the Lyapunov function will converge to zero more quickly, which results in a faster convergence speed. That would help the robot to return to its steady gliding path with less time.

6.2.2 Stability Analysis for the Full Closed-loop System

From Eqs. (6.23) - (6.25), the designed controller u_δ in (6.27) is dependent on the reduced model states as

$$u_\delta = \frac{1}{K_c} \arctan(\frac{1}{K_{De}}(K_{F_\delta} \cos\alpha_e \sin\bar{\theta}_g(1+\bar{V})^2 + K_{F_\delta} \sin\alpha_e(1+\bar{V})^2((1+\bar{V})^2 - \cos\bar{\theta}_g)))$$

(6.30)

If applying the above controller to the full-order system (6.10) - (6.13), we will have the closed-loop system, expressed in the compact form as

$$\frac{dz}{dt_n} = h(z, u_\delta(\xi))$$

(6.31)

where the full system state vector $z = \begin{bmatrix} \xi & \eta \end{bmatrix}^T$.

It is challenging to establish the global stability of the origin in (6.31), so it is focused on the local stability in this dissertation. The linearization matrix is defined as

$$A = \frac{\partial h}{\partial z}\bigg|_0 = [a_{ij}]_{4\times 4}$$

(6.32)

The calculated elements in the Jacobian matrix A are

$$a_{11} = -2 - \frac{2K_{F_\delta}{}^2 \sin^2 \alpha_e}{K_c K_{De}{}^2},$$

$$a_{12} = -\frac{m_0 g \cos \theta_{ge}}{K_{De} V_e^2} - \frac{K_{F_\delta}{}^2 \sin \alpha_e \cos \alpha_e}{K_c K_{De}{}^2},$$

$$a_{13} = -\frac{2K_D \alpha_e}{K_{De}},$$

$$a_{21} = \frac{m_0 g \cos \theta_{ge}}{K_{De} V_e^2} + \frac{K_L \alpha_e}{K_{De}} - \frac{2K_{F_\delta}{}^2 \sin \alpha_e \cos \alpha_e}{K_c K_{De}{}^2},$$

$$a_{22} = \frac{m_0 g \sin \theta_{ge}}{K_{De} V_e^2} - \frac{K_{F_\delta}{}^2 \cos^2 \alpha_e}{K_c K_{De}{}^2},$$

$$a_{23} = K_L / K_{De},$$

$$a_{14} = 0, \quad a_{24} = 0, \quad a_{31} = a_{21}, \quad a_{32} = a_{22}, \quad a_{33} = a_{23},$$

$$a_{41} = -\frac{1}{\varepsilon_2} \frac{2K_{F_\delta} \sin \alpha_e}{K_c K_{De}},$$

$$a_{42} = -\frac{1}{\varepsilon_2} \frac{K_{F_\delta} \cos \alpha_e}{K_c K_{De}},$$

$$a_{34} = 1/\varepsilon_1, \quad a_{43} = -1/\varepsilon_2, \quad a_{44} = -1/\varepsilon_2.$$

By examining the A matrix for the Hurwitz property, we would know whether the closed-loop system with the passivity-based controller is asymptotically stable at its equilibrium. However, it is technically difficult to check this 4-by-4 matrix directly unless we use a numerical approach. But due to the time-scale separation property of system (6.31), there is an alternative way to check the stability, by checking two 2-by-2 matrices on their Hurwitz property [97].

First, we break the matrix A into four blocks using four 2-by-2 matrices $A_{11}, A_{12}, A_{21}, A_{22}$.

Here,

$$
A = \begin{bmatrix} A_{11} & A_{12} \\ \frac{1}{\mu}A_{21} & \frac{1}{\mu}A_{22} \end{bmatrix}
$$

Following [97], instead of checking matrix A, we only need to check 2-by-2 matrices A_{22} and $A_{11} - A_{12}A_{22}^{-1}A_{21}$ on whether they are Hurwitz, for proving the stability of the full-order system when μ is sufficiently small. Plugging the calculated elements of linearization matrix A into those four matrices, we have

$$
A_{22} = \begin{bmatrix} \mu\dfrac{K_L}{K_{De}} & \dfrac{\mu}{\varepsilon_1} \\ -\dfrac{\mu}{\varepsilon_2} & -\dfrac{\mu}{\varepsilon_2} \end{bmatrix}, \qquad A_{11} - A_{12}A_{22}^{-1}A_{21} = \begin{bmatrix} b_1 & b_2 \\ b_3 & b_4 \end{bmatrix}
$$

where

$$
b_1 = -2 - \frac{2K_{F_\delta}{}^2 \sin^2 \alpha_e}{K_c K_{De}^2} + \frac{4\mu^2 K_{F_\delta} \alpha_e \sin \alpha_e K_D}{\varepsilon_1 \varepsilon_2 K_c K_{De}^2}
$$

$$
b_2 = -\frac{m_0 g \cos \theta_{ge}}{K_{De} V_e^2} - \frac{K_{F_\delta}{}^2 \sin \alpha_e \cos \alpha_e}{K_c K_{De}^2} + \frac{2\mu^2 K_{F_\delta} K_D \alpha_e \cos \alpha_e}{\varepsilon_1 \varepsilon_2 K_c K_{De}^2}
$$

$$
b_3 = \frac{m_0 g \cos \theta_{ge}}{K_{De} V_e^2} + \frac{K_L \alpha_e}{K_{De}} - \frac{2K_{F_\delta}{}^2 \sin \alpha_e \cos \alpha_e}{K_c K_{De}^2} - \frac{2\mu^2 K_{F_\delta} K_L \sin \alpha_e}{\varepsilon_1 \varepsilon_2 K_c K_{De}^2}
$$

$$
b_4 = \frac{m_0 g \sin \theta_{ge}}{K_{De} V_e^2} - \frac{K_{F_\delta}{}^2 \cos^2 \alpha_e}{K_c K_{De}^2} - \frac{\mu K_{F_\delta} K_L \cos \alpha_e}{\varepsilon_1 K_c K_{De}^2}
$$

For matrix A_{22}, the characteristic equation is

$$
\lambda^2 + k_1 \lambda + k_0 = 0
$$

where

$$k_1 = \frac{\mu}{\varepsilon_2} - \mu \frac{K_L}{K_{De}}, \qquad k_0 = \frac{\mu^2}{\varepsilon_1 \varepsilon_2} - \mu \frac{K_L \mu}{K_{De} \varepsilon_2}$$

Because $\mu \sim O(\varepsilon_i)$, i.e., μ is in the same order of the time scale ε_i, and they are both sufficiently small, it can be shown easily that the coefficients k_1 and k_0 are both positive, which means A_{22} is Hurwitz.

Now for matrix $A_{11} - A_{12} A_{22}^{-1} A_{21}$, let the coefficients of the characteristic equation be l_1 and l_0, defined similar to k_1 and k_0. It is easy to get

$$l_1 = -(b_1 + b_4), \qquad l_0 = b_1 b_4 - b_2 b_3 \tag{6.33}$$

Here we exploit information about the gliding robotic fish system and set a range for those parameters. From the gliding robotic fish application perspective, we have

$$\sin \alpha_e \approx \alpha_e, \quad \cos(\alpha_e) \approx 1, \quad 0 \leq K_{F_\delta} \sim O(10), \quad V_e \leq 1, \quad K_{De} \sim O(10), \quad K_{u_\delta} \sim O(1),$$

$$K_{Le} \sim O(10), \quad K_L \sim O(100), \quad K_D \sim O(100), \quad 0.25 \leq |\theta_{ge}| \leq 0.75$$

where metric units are applied to all above parameter values. Besides, we bound the control parameter K_c in an open set $(1, 10)$ to restrain the magnitude of the control u_δ from $9°$ to $90°$, which is consistent with the application constraints. Then we find that for each element in the 2-by-2 matrix $A_{11} - A_{12} A_{22}^{-1} A_{21}$, there is a dominant term, which determines the sign of that element, with all other minor terms only influencing the value. We express the approximation using the dominant

terms for the matrix elements

$$b_1 \approx -2, \qquad b_2 \approx -\frac{m_0 g \cos \theta_{ge}}{K_{De} V_e^2}, \qquad b_3 \approx \frac{m_0 g \cos \theta_{ge}}{K_{De} V_e^2}, \qquad b_4 \approx \frac{m_0 g \sin \theta_{ge}}{K_{De} V_e^2}$$

Since m_0 and θ_{ge} are always opposite in sign, $b_4 < 0$. We also notice that b_2 and b_3 are opposite in sign. So from equation (6.33), we have $l_1 > 0$ and $l_0 > 0$. With all characteristic equation coefficients positive, the matrix $A_{11} - A_{12} A_{22}^{-1} A_{21}$ is Hurwitz.

From the above analysis, we have shown that the original 4-by-4 linearization matrix A is Hurwitz for sufficiently small μ, which proves that the passivity-based controller derived from the approximated reduced model also stabilizes the full-order system. Furthermore, by the fact that the controller is designed based on the approximated reduce system through passivity analysis, we conjecture that this controller will be similarly effective for the full system as it does for the approximated reduced system. In particular, we anticipate that the controller will provide a faster convergence speed than the open-loop controller $u_\delta = 0$, due to the additional negative term it introduced into (6.29).

6.2.3   Simulation Results

We apply the passivity-based controller to the full dynamics model, and use Matlab Simulink to simulate the controller performance. The robot parameters we used are: $m = 10$ kg, $J_2 = 0.08$ kgm^2, $K_{L0} = 0$ kg/m, $K_L = 303.6$ kg/m, $K_{D0} = 3.15$ kg/m, $K_D = 282.8$ kg/m, $K_q = -0.8$ kgs, $K_{M0} = 0.39$ kg, $K_M = -14.7$ kg, $\delta = 29.5$, $m_0 = 0.05$ kg. The equilibrium point is $V_e = 0.24$ m/s, $\theta_{ge} = -22.5°$, $\alpha_e = 1.52°$ and $\omega_{2e} = 0$ rad/s. Suppose that we have a current disturbance that makes the robot deviate off its steady gliding path. From that point we want the robot to return to its equilibrium gliding profile. The initial states are given as $V_0 = 0.2$ m/s, $\theta_{g0} = -35°$, $\alpha_0 = 1°$ and
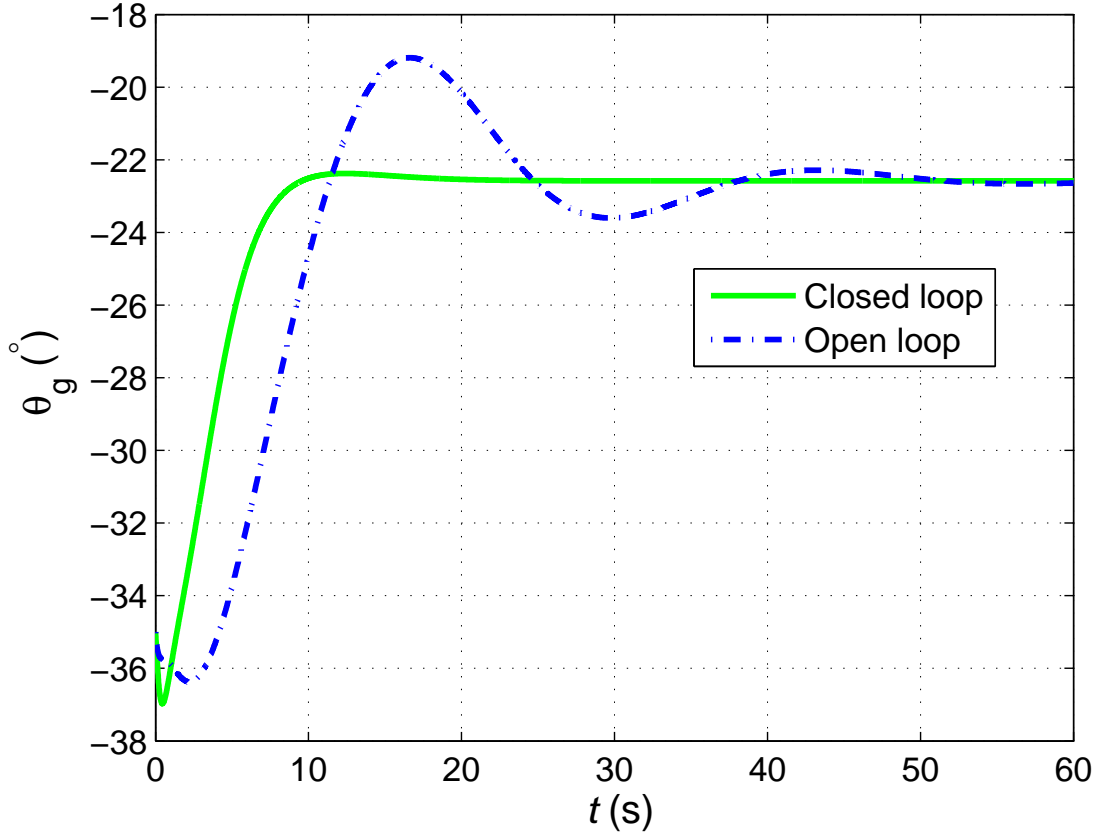
Figure 6.2: Simulation results on the trajectories of the gliding angle θ_g for the open-loop $u_\delta = 0$ and closed-loop ($K_c = 2$) cases, respectively.

$\omega_{20} = 0$ rad/s. In simulation, we also consider the dynamics of the actuator for moving the control surface, approximated by a first-order system with a time constant of 10 ms. The simulation time is 60 seconds.

Fig. 6.2 shows that the passivity-based controller designed for the reduced model works for the original full-order system, not only stabilizing the steady gliding equilibrium but also speeding up the convergence process as we expected from the analysis. Figs. 6.3-6.4 show the influences of the control parameter K_c on the control output and the glide angle transients. It can be seen that with a smaller K_c, the system converges faster but requires larger initial control effort. With the arctangent function in (6.28), the tunable parameter K_c makes it easy to balance between the
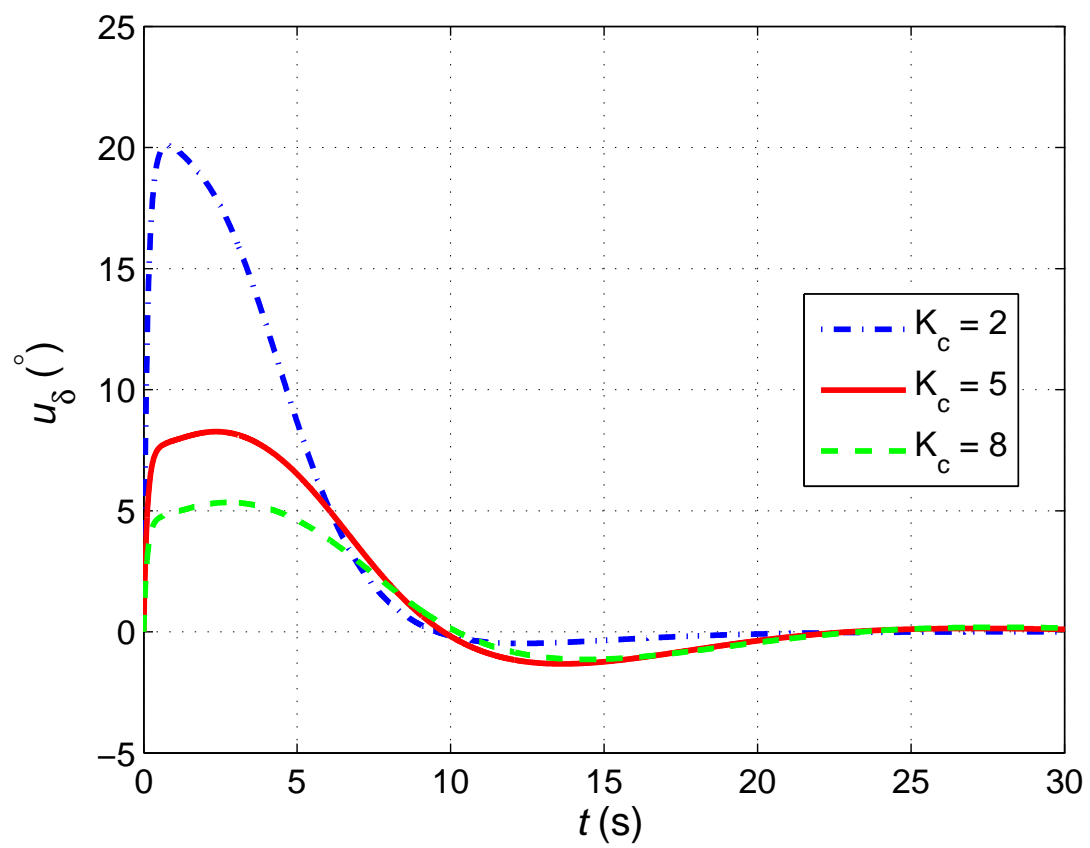
Figure 6.3: The trajectory of control u_δ for the closed-loop simulation with different values for K_c.
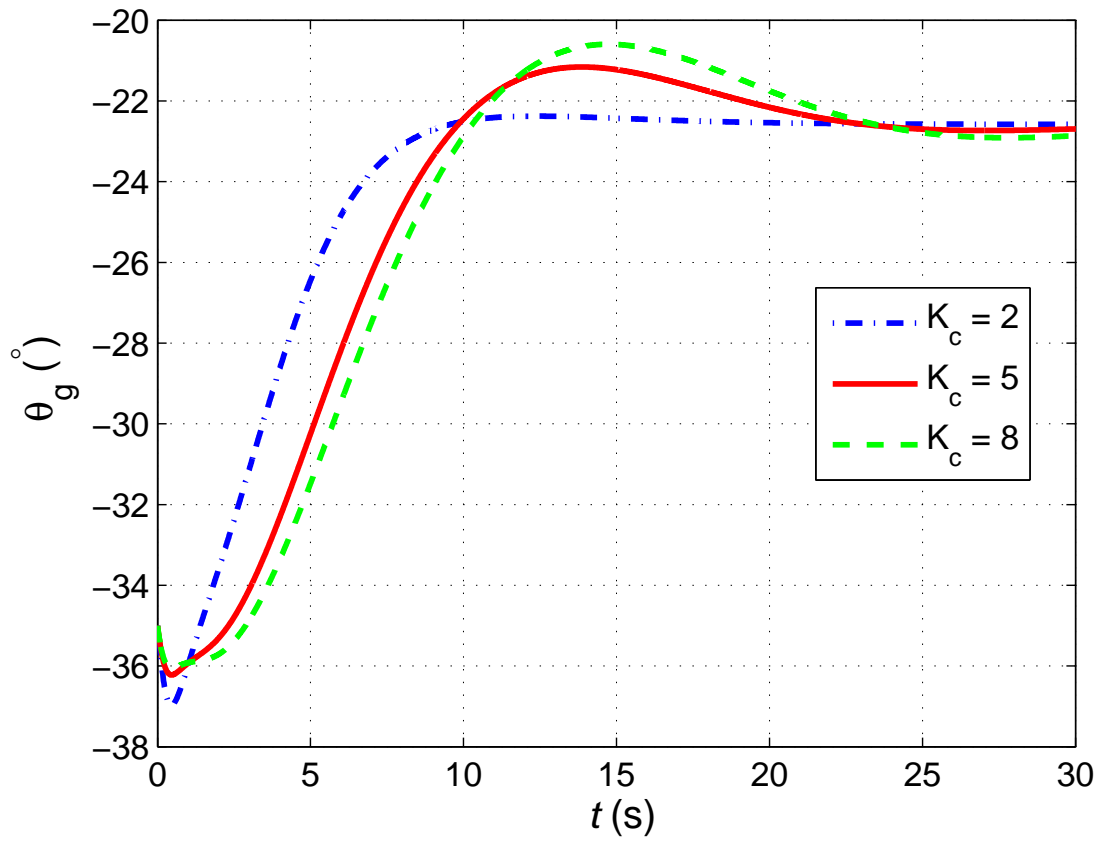
Figure 6.4: The trajectory of gliding angle θ_g for the closed-loop simulation with different values for K_c.

6.3 Observer Design

In this section, we propose a nonlinear model-based observer to estimate the velocity V and the gliding angle θ_g (see (6.30)), using only the pitch angle which can be measured from onboard sensors. The local stability of the observer is analyzed with a constructed Lyapunov function [98].

A nonlinear observer is proposed in order to estimate system states with a relatively large convergence region. The observer gain structure is selected to be linear, to enable efficient computation and onboard implementation in experiments. The gain is obtained by solving the Riccati equation as in the Extended Kalman Filter, in order to take the robustness to the measurement noise into the design consideration. The nonlinear observer can be expressed as

$$\dot{\hat{x}} = f(\hat{x}, u_\delta) + H_o(\theta - \hat{\theta}) \tag{6.34}$$

$$\hat{\theta} = h(\hat{x}) \tag{6.35}$$

Here, $\hat{x} = [\ \hat{V}\quad \hat{\theta}_g\quad \hat{\alpha}\quad \hat{\omega}_2\ ]^T$ is the estimated system state, $f(\hat{x}, u_\delta)$ is the observer dynamics, as described in Eqs. (6.6)–(6.9), $h(\hat{x}) = \hat{\theta}_g + \hat{\alpha}$ is the output function, and H_o is the observer gain.

Let Q be a 4-by-4 symmetric positive definite matrix, which denotes the process noise covariance for the state dynamics. Let R be a constant, representing the measurement noise variance. Let P be the solution of the Riccati equation

$$A_o P + P A_o^T + Q - P C_o^T R^{-1} C_o P = 0 \tag{6.36}$$

where A_o and C_o are the linearization matrix of the system,

$$
A_o =
\begin{bmatrix}
-\dfrac{1}{m_1}\dfrac{\partial D}{\partial V} & -\dfrac{m_0 g \cos\theta_g}{m_1} & -\dfrac{1}{m_1}\dfrac{\partial D}{\partial \alpha} & 0 \\[2ex]
\dfrac{1}{m_1 V}\dfrac{\partial L}{\partial V} & \dfrac{m_0 g \sin\theta_g}{m_1 V} & \dfrac{1}{m_1 V}\dfrac{\partial L}{\partial \alpha} & 0 \\[2ex]
-\dfrac{1}{m_1 V}\dfrac{\partial L}{\partial V} & -\dfrac{m_0 g \sin\theta_g}{m_1 V} & -\dfrac{1}{m_1 V}\dfrac{\partial L}{\partial \alpha} & 1 \\[2ex]
\dfrac{2V(K_{M0}+K_M\alpha)}{J_2} & 0 & \dfrac{K_M V^2}{J_2} & \dfrac{K_{q2}V^2}{J_2}
\end{bmatrix}_e
$$

$$
C_o =
\begin{bmatrix}
0 & 1 & 1 & 0
\end{bmatrix}
$$

Here, $[\cdot]_e$ means that matrix elements are evaluated at the equilibrium point.

$$
\frac{\partial L}{\partial \alpha} = K_L V^2, \qquad \frac{\partial L}{\partial V} = 2V(K_{L0}+K_L\alpha), \qquad \frac{\partial D}{\partial \alpha} = 2K_D V^2\alpha, \qquad \frac{\partial D}{\partial V} = 2V(K_{D0}+K_D\alpha^2).
$$

The existence of a positive definite matrix P is guaranteed by the observability of (A_o, C_o). If P exists, the observer gain H_o can be designed as

$$
H_o = PC_o^T R^{-1} \tag{6.37}
$$

By tuning the value of R, we can adjust the robustness of the observer to different levels of measurement noise.

The stability of the designed nonlinear observer is analyzed below. We first define the dynamics

system states as $x = [\ V\quad \theta_g\quad \alpha\quad \omega_2\ ]^T$. Then the system dynamics can be expressed as

$$\dot{x} = f(x, u_\delta) \tag{6.38}$$

$$\theta = h(x) \tag{6.39}$$

Define the state estimation error as $\tilde{x} = x - \hat{x}$, and then from Eqs. (6.34) (6.35) (6.38) (6.39) we have the estimation error dynamics

$$\dot{\tilde{x}} = f(x, u_\delta) - f(\hat{x}, u_\delta) - H_o(h(x) - h(\hat{x})) \tag{6.40}$$

Taylor series of functions f and h are taken at the equilibrium point, and we have

$$\dot{\tilde{x}} = (A_o - H_o C_o)\tilde{x} + \eta(\tilde{x}, t) \tag{6.41}$$

Here $\eta(\tilde{x}, t)$ represents the sum of Taylor series terms that contain second-order and higher-order $\tilde{x}$. Furthermore, $\eta(0, t) = 0$. There exist positive constants c_0 and k_1 such that when estimation error $|\tilde{x}| < c_0$ and x is bounded, we have

$$\|\eta(\tilde{x}, t)\| \le k_1 \|\tilde{x}\|^2 \tag{6.42}$$

We define a Lyapunov function $V(\tilde{x})$ for the system (6.41)

$$V(\tilde{x}) = \tilde{x}^T P^{-1} \tilde{x} \tag{6.43}$$

We take the time derivative of $V(\tilde{\boldsymbol{x}})$

$$\dot{V}(\tilde{\boldsymbol{x}}) = \tilde{\boldsymbol{x}}^T P^{-1}\dot{\tilde{\boldsymbol{x}}} + \dot{\tilde{\boldsymbol{x}}}^T P^{-1}\tilde{\boldsymbol{x}} \tag{6.44}$$

From Eqs. (6.36), (6.37), (6.41), (6.44), we have

$$\dot{V}(\tilde{\boldsymbol{x}}) = -\tilde{\boldsymbol{x}}^T (C_0^T R^{-1} C_0 + P^{-1} Q P^{-1})\tilde{\boldsymbol{x}} + 2\tilde{\boldsymbol{x}}^T P^{-1}\eta(\tilde{\boldsymbol{x}}, t) \tag{6.45}$$

Here $(C_0^T R^{-1} C_0 + P^{-1} Q P^{-1}) \geq \sigma_{\min} \mathbf{1}_{4\times 4}$ is positive definite, and $\sigma_{\min}$ is a finite scaler. From Eq. (6.42), the time derivative of Lyapunov function is bounded

$$\dot{V}(\tilde{\boldsymbol{x}}) \leq -c_1\|\tilde{\boldsymbol{x}}\|^2 + c_2\|\tilde{\boldsymbol{x}}\|^3 \tag{6.46}$$

Here c_1 and c_2 are positive scalar coefficients that depend on the noise signals and the system matrices. The coefficients can be selected as

$$c_1 = \sigma_{\min}, \quad c_2 = 2k_1\|P^{-1}\| \tag{6.47}$$

It can be easily shown from Eq. (6.46) that the state estimation error $\tilde{\boldsymbol{x}}$ dynamics is locally exponentially stable. Furthermore, the local exponential stability of the full closed-loop system can be verified by the separation principle [96] with the proven local exponential stability property of the passivity-based controller and the nonlinear observer.

We also consider the influence of the system noise and measurement noise on the observer

stability. The noise-corrupted system dynamics can be expressed as

$$\dot{x} = f(x, u_\delta) + v_1 \tag{6.48}$$

$$\theta = h(x) + v_2 \tag{6.49}$$

where v_1 and v_2 represent the process noise and measurement noise, respectively. The state estimation error dynamics for such a system is

$$\dot{\tilde{x}} = f(x, u_\delta) - f(\hat{x}, u_\delta) - H_o(h(x) - h(\hat{x})) + v_1 - H_o v_2 \tag{6.50}$$

Taylor series of functions f and h are taken at the equilibrium point, and we have

$$\dot{\tilde{x}} = (A_o - H_o C_o)\tilde{x} + \eta(\tilde{x}, t) + \xi(t) \tag{6.51}$$

Here $\xi(t) = v_1 - H_o v_2$. Assuming that x is bounded and the noise signals v_1 and v_2 are also bounded, there exist constant k_1 and k_2 such that

$$\|\eta(\tilde{x}, t)\| \le k_1 \|\tilde{x}\|^2, \quad \|\xi(t)\| \le k_2 \tag{6.52}$$

We take the same Lyapunov function $V(\tilde{x})$ for the system (6.51)

$$V(\tilde{x}) = \tilde{x}^T P^{-1} \tilde{x} \tag{6.53}$$

From Eqs. (6.36), (6.37), (6.44), (6.51), the time derivative of $V(\tilde{x})$ is

$$\dot{V}(\tilde{x}) = -\tilde{x}^T (C_0^T R^{-1} C_0 + P^{-1} Q P^{-1}) \tilde{x} + 2\tilde{x}^T P^{-1} (\eta(\tilde{x},t) + \xi(t)) \tag{6.54}$$

Because of the boundedness properties of the noise signal, the time derivative of Lyapunov function is bounded

$$\dot{V}(\tilde{x}) \leq -c_1 \|\tilde{x}\|^2 + c_2 \|\tilde{x}\|^3 + c_3 \|\tilde{x}\| \tag{6.55}$$

Here c_1, c_2 and c_3 are positive scalar coefficients that depend on the noise signals and the system matrices. The coefficients can be selected as

$$c_1 = \sigma_{\min}, \quad c_2 = 2k_1 \|P^{-1}\|, \quad c_3 = 2k_2 \|P^{-1}\| \tag{6.56}$$

It can be shown from Eq. (6.55) that the state estimation error will be bounded

$$|\tilde{x}| \leq \frac{c_1}{2c_2} \quad \text{when } 4c_2 c_3 - c_1^2 \leq 0 \text{ and } \frac{c_1}{2c_2} < c_0 \tag{6.57}$$

The condition inequalities in Eq. (6.57) can be checked for a given system with known system matrices and the noise signal characteristics. We speculate that the stabilization output of the full closed-loop system will also be bounded around the nominal value, based on the boundedness of the state estimation error and the exponential stability of the designed passivity-based controller.

Simulation is carried out to examine the performance of the designed observer. The closed-loop system with the passivity-based controller for stabilization is simulated with full state feedback. The observer runs in parallel with the closed-loop dynamics with the measured pitch angle as the
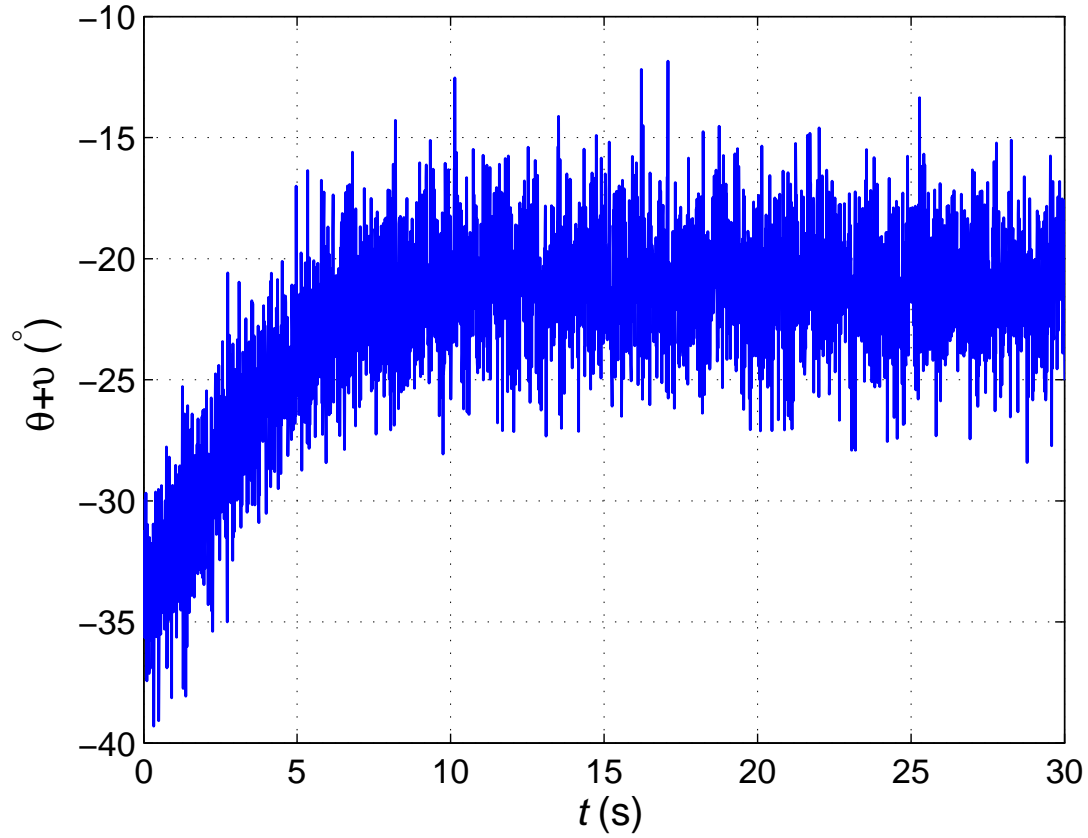
Figure 6.5: Simulated trajectory of the measured pitch angle which is corrupted with noise. The measurement noise is modeled as Gaussian random process with $R = 0.1$.

observer input. In simulation, measurement noise $\upsilon(t)$, added to the system output $\theta(t)$, is modeled as a Gaussian random process with zero-mean, $E\{\upsilon\} = 0$, and variance R, $E\{\upsilon(t)\upsilon(\tau)\} = R\delta(t-\tau)$. Fig. 6.5 shows the noise-corrupted system output, the measured pitch angle with the variance of measurement noise $R = 0.1$. Fig. 6.6 shows both the real and estimated gliding angle trajectories. Gliding angle θ_g is used here to evaluate the effectiveness of state estimation because it is the signature variable in the sagittal-plane glide motion. Control parameter $K_c = 2$ is used. The simulation results show that the proposed observer is able to estimate the system states with good robustness to the measurement noise.
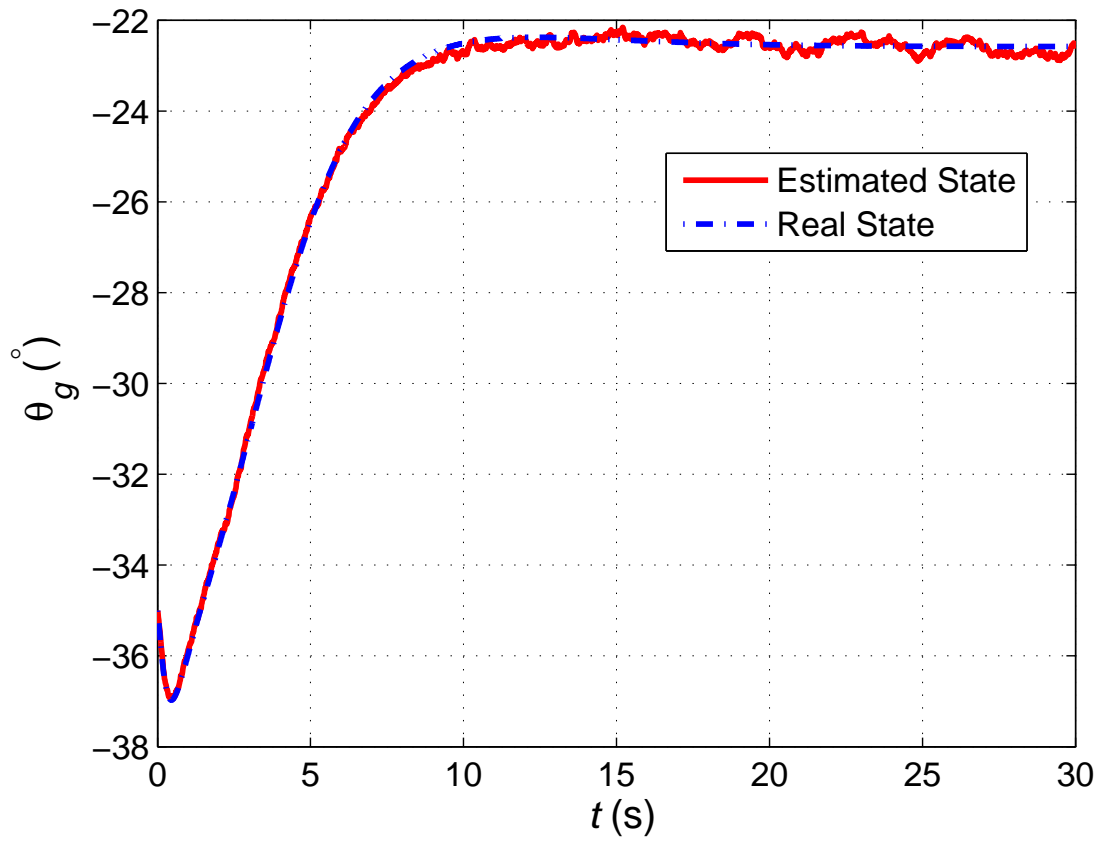
Figure 6.6: Simulation results: the trajectories of the gliding angle θ_g of the real state and nonlinear observer estimation with measurement noise.

6.4 Experimental Results

In order to test the effectiveness of the proposed passivity-based controller (Section 6.2.1) and the nonlinear observer (Section 6.3), we conduct both open-loop and closed-loop experiments using a gliding robotic fish prototype "Grace". The tail fin system in "Grace" is driven by a servo motor (Hitec Servo HS-7980TH) through a chain transmission. In experiments, the tail is adjusted up or down like a whale fluke to modulate the glide motion. A microcontroller (dsPIC 30F6014A) in the robot runs the glide control program and provides storage memory for the measurement data. "Grace" is also equipped with the inertial measurement units including gyros (ST LPY503AL), accelerometers and a digital compass (ST LSM303DLMTR), which are used to measure the robot's attitude.

6.4.1 Filter Design

The IMU sensor output is corrupted with a high-frequency noise. The system output, namely, the pitch angle, is computed from the accelerometer sensor output. In this dissertation, in order to smooth the measured pitch angle, a second-order Butterworth digital filter is adopted. The discrete-time transfer function of the filter can be expressed as

$$H(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 + b_1 z^{-1} + b_2 z^{-2}} \tag{6.58}$$

where a_0, a_1, a_2, b_1, b_2 are filter coefficients.

Let f_r denote the ratio between the sampling frequency f_s and the cutoff frequency f_c

$$f_r = \frac{f_s}{f_c} \tag{6.59}$$

For the Butterworth filter, the above coefficients can be determined as the follows

$$a_0 = a_2 = \frac{\omega_c'^2}{c}$$

$$a_1 = 2a_0$$

$$b_1 = \frac{2(\omega_c'^2 - 1)}{c}$$

$$b_2 = \frac{1 - 2\cos(\pi/4)\omega_c' + \omega_c'^2}{c}$$

where $\omega_c' = \tan(\pi/f_r)$, and $c = 1 + 2\cos(\pi/4)\omega_c' + \omega_c'^2$.

In this dissertation, the sampling frequency is 50 Hz, and the cutoff frequency is designed to be 10 Hz. The low-pass Butterworth filter for the pitch angle can be described using the following recursive difference equation,

$$\theta(n) = a_0\theta_s(n) + a_1\theta_s(n-1) + a_2\theta_s(n-2) - b_1\theta(n-1) - b_2\theta(n-2) \qquad (6.60)$$

where $\theta(n)$ and $\theta_s(n)$ are the Butterworth filter's output and input (pitch angle) at the nth step, respectively. In this dissertation, $a_0 = a_2 = 0.20644$, $a_1 = 0.41289$, $b_1 = -0.3702$, $b_2 = 0.19597$.

6.4.2 Open-loop Experiments

Open-loop experiments are first conducted in a large indoor tank, which measures 5 m long, 3.3 m wide, and 1.3 m deep, using the gliding robotic fish prototype named "Grace" (Fig. 6.7). The robot is first released from the water surface with a fixed tail angle δ, and then water is pumped into the robot's body until the net buoyancy m_0 reaches 50 g. Two seconds after the gliding down motion is initiated, the observer is initialized in the microcontroller. During the following period, the robot records the pitch angle readings from both onboard sensors and the observer. The
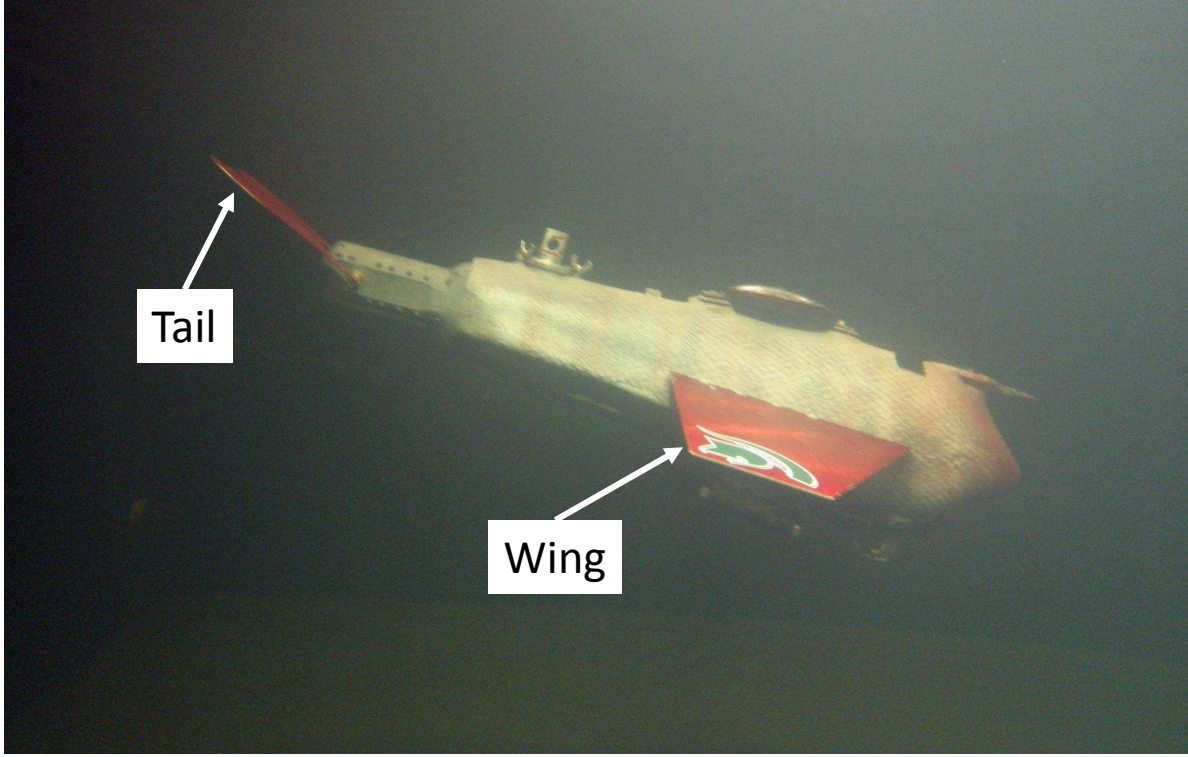
Figure 6.7: A snapshot of the open-loop experiment with fixed tail angle using gliding robotic fish "Grace" in the lab tank.

readings are further compared together with the pure computer-based Matlab simulation result, which is obtained by running the same observer dynamics continuously in Matlab based on the sensor output history recorded onboard. Fig. 6.8 shows the experimental results on the pitch angle θ for different fixed tail angles $\delta = 15^\circ, 30^\circ, 45^\circ$, where the measured pitch angle is compared with the values derived from the state estimate with Eq. (6.35). First, we can observe that the computer-based Matlab simulation of the nonlinear observer produces estimate trajectories almost identical to those from the onboard observer, confirming that the microcontroller is able to execute the observer with little loss in accuracy. Second, the match between the observer estimation and sensor output further validates the design of the nonlinear observer.
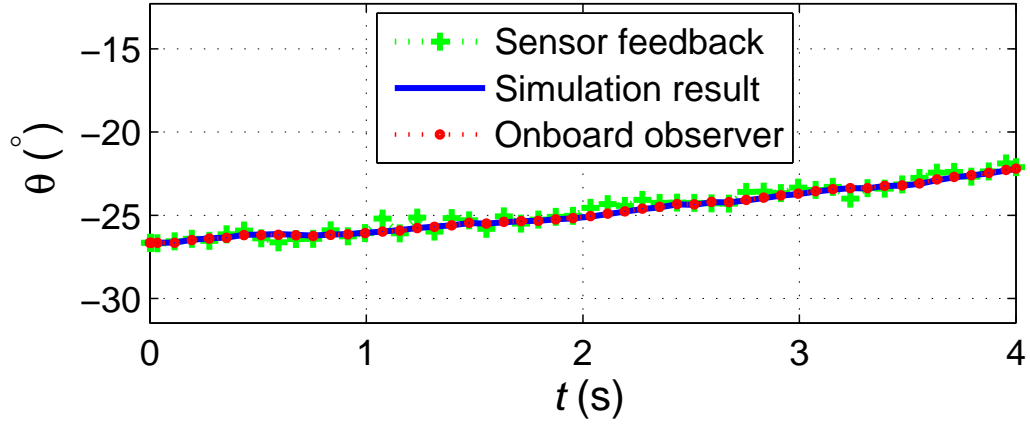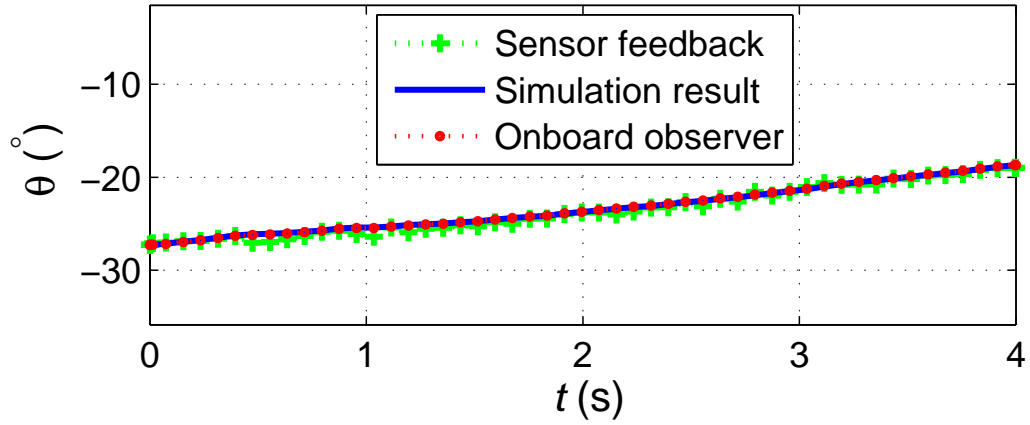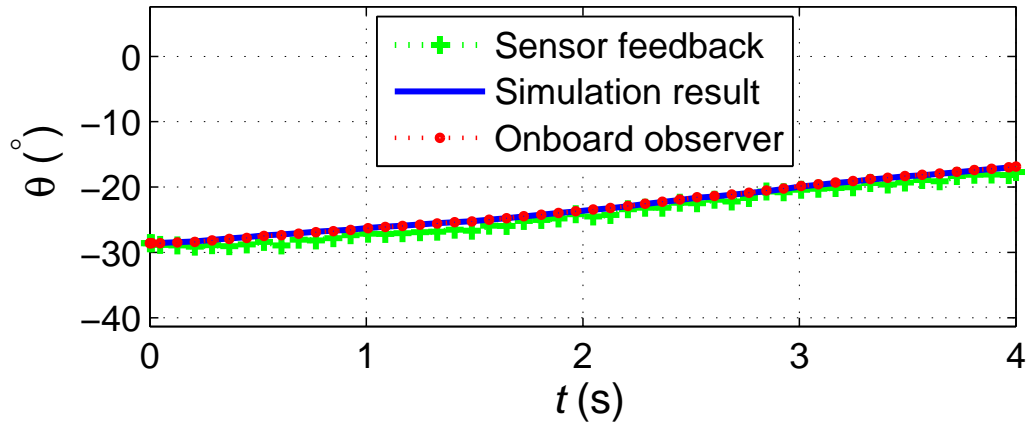
Figure 6.8: The trajectories of pitch angle of the on-board sensor reading, observer estimation and computer-based simulation result, in the open-loop experiments using gliding robotic fish "Grace". (a) $\delta = 15°$; (b) $\delta = 30°$; (c) $\delta = 45°$.

6.4.3 Closed-loop Experiments

In this subsection, experimental results on sagittal-plane stabilization using the passivity-based controller are presented to verify the effectiveness of both the proposed nonlinear controller and the nonlinear observer. Experiments are conducted in the Neutral Buoyancy Research Facility (NBRF) at the University of Maryland, College Park, which is a water tank measuring 15 m across and 7.5 m deep.

In the experiments, the robot is released from the water surface with a net buoyancy $m_0 = 50$ g. The tail angle δ is initially set to $60°$. Then the robot enters gliding down motion. When the robot reaches a preset depth of 1 m, the tail flaps to $\delta = 0°$ to provide an initial perturbation for the stabilization process. Then the designed passivity-based controller (6.30) kicks in to stabilize the system, with the state estimation using the nonlinear observer (Fig. 6.9). A Qualysis underwater motion capture system, which consists of 12 underwater cameras and motion tracking software, is used to record the whole stabilization process and analyze the motion afterwards (Fig. 6.10).

Fig. 6.11 shows the experimental results on three types of pitch angle trajectories including onboard sensor measurements, onboard observer estimation, and motion capture system output, together with the onboard gliding angle estimation, when no feedback control exists, with tail angle trajectory designed as in Fig. 6.12. Figs. 6.13–6.16 show the experimental results for passivity-based stabilization for two different values of the controller gain. From the experimental results, we observe a good match among the pitch angle estimation, the on-board sensor reading, and the motion capture system output. The results further verify the proposed nonlinear observer design. Besides, the gliding angle converges to the equilibrium point around $-23.5°$. The results confirm that the passivity-based controller effectively stabilizes the sagittal-plane glide motion and speeds up the convergence process, comparing with the glide angle trajectory in the open-loop
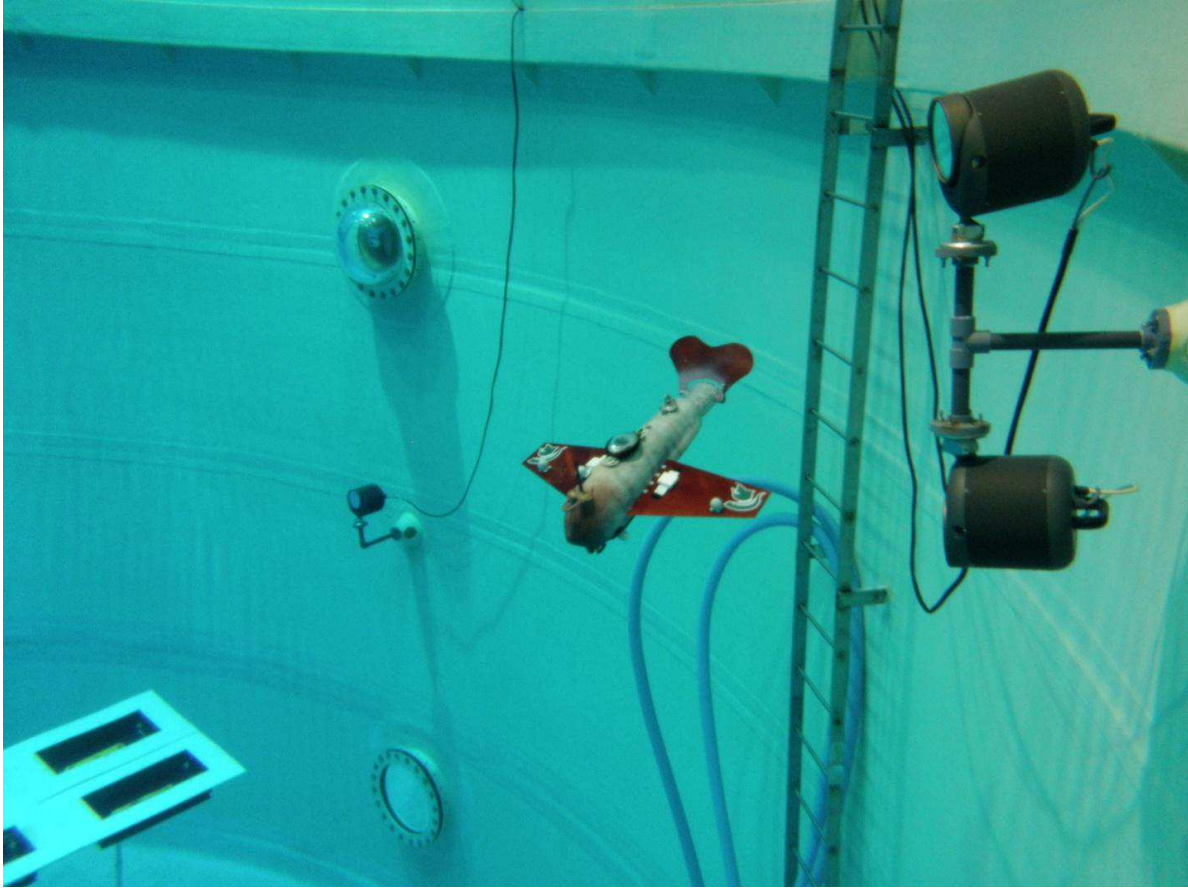
Figure 6.9: A snapshot of stabilization experiment using gliding robotic fish "Grace" in NBRF, University of Maryland.

Figure 6.10: A snapshot of stabilization experiment in the view of the 12 underwater cameras of the Qualysis motion capture system in NBRF, University of Maryland.
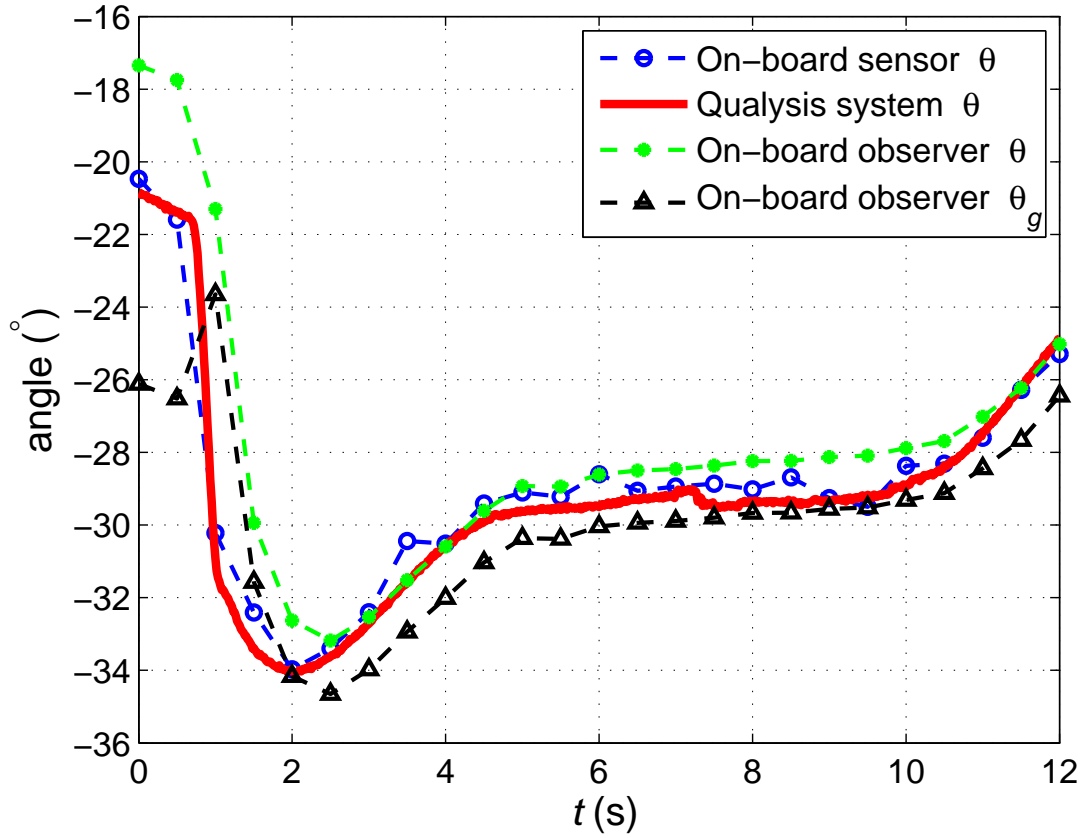
Figure 6.11: The trajectories of pitch angle and gliding angle in the experiment of gliding stabilization without feedback control.

case (Fig. 6.11). It also shows that with a smaller controller gain K_c, the arising time of the pitch angle is shorter (1 second in Fig. 6.15 vs 2 seconds in Fig. 6.13) and the control output magnitude is larger ($25°$ in Fig. 6.16 vs $15°$ in Fig. 6.14), which is consistent with and complementary to the simulation findings in Section 6.2.3. This provides insight into the control parameter design in order to balance between convergence time and control efforts.

In experiments, we also implemented proportional controller and PI controller for glide stabilization for the purpose of comparison. Experimental results using a proportional controller and a PI controller are shown in Figs. 6.17–6.18 and Figs. 6.19–6.20, respectively. We varied the controller parameters, the proportional gain K_P and the integral gain K_I in order to obtain a better
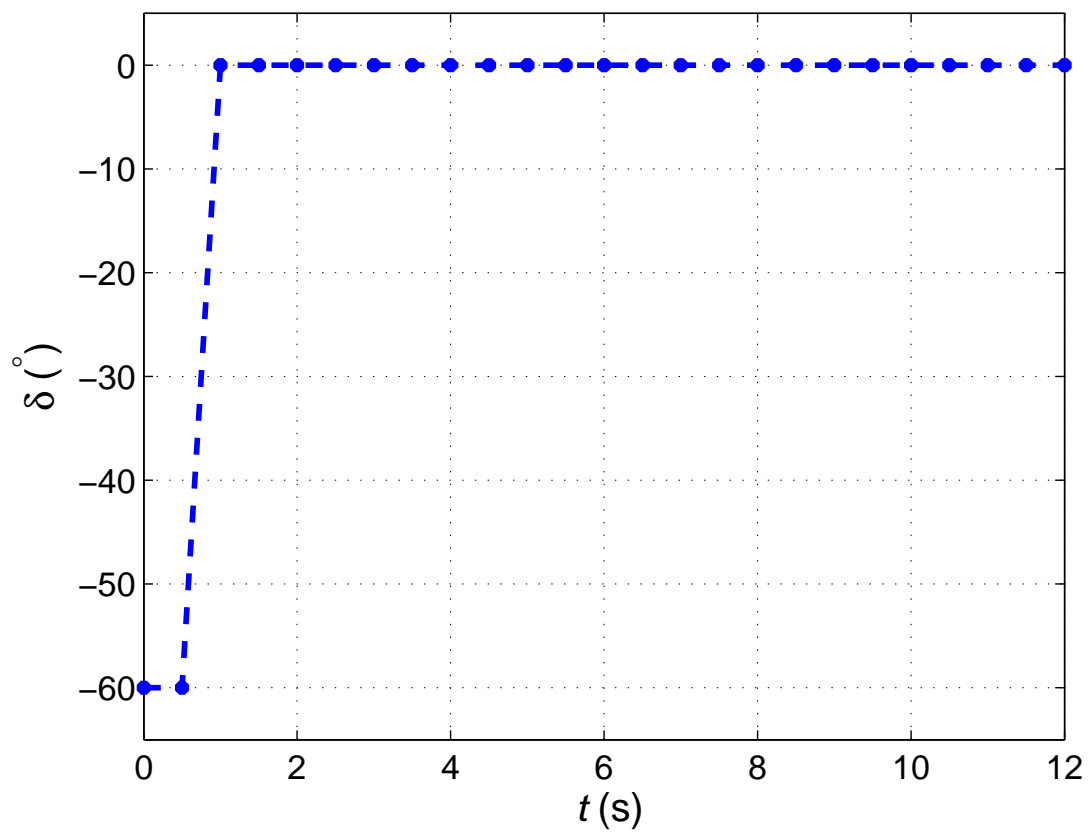
Figure 6.12: The trajectory of predefined tail angle in the experiment of gliding stabilization without feedback control.
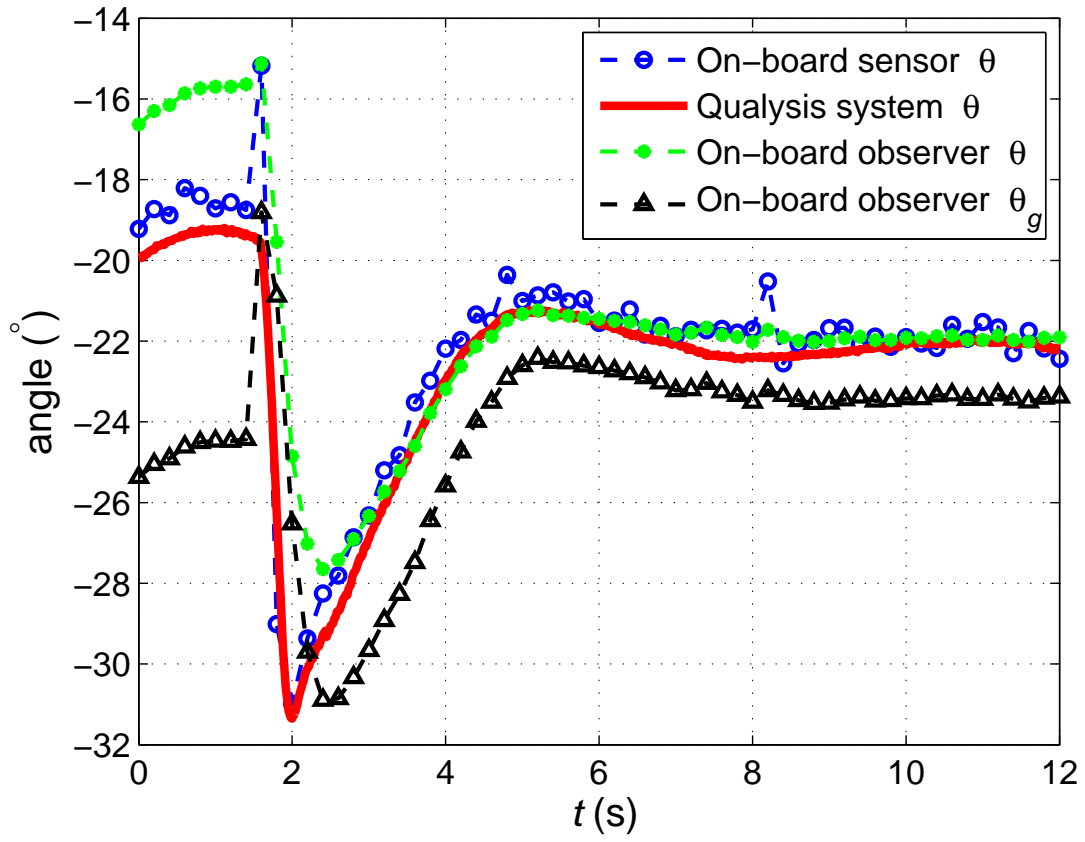
Figure 6.13: The trajectories of pitch angle and gliding angle in the passivity-based stabilization experiment with $K_c = 3$.
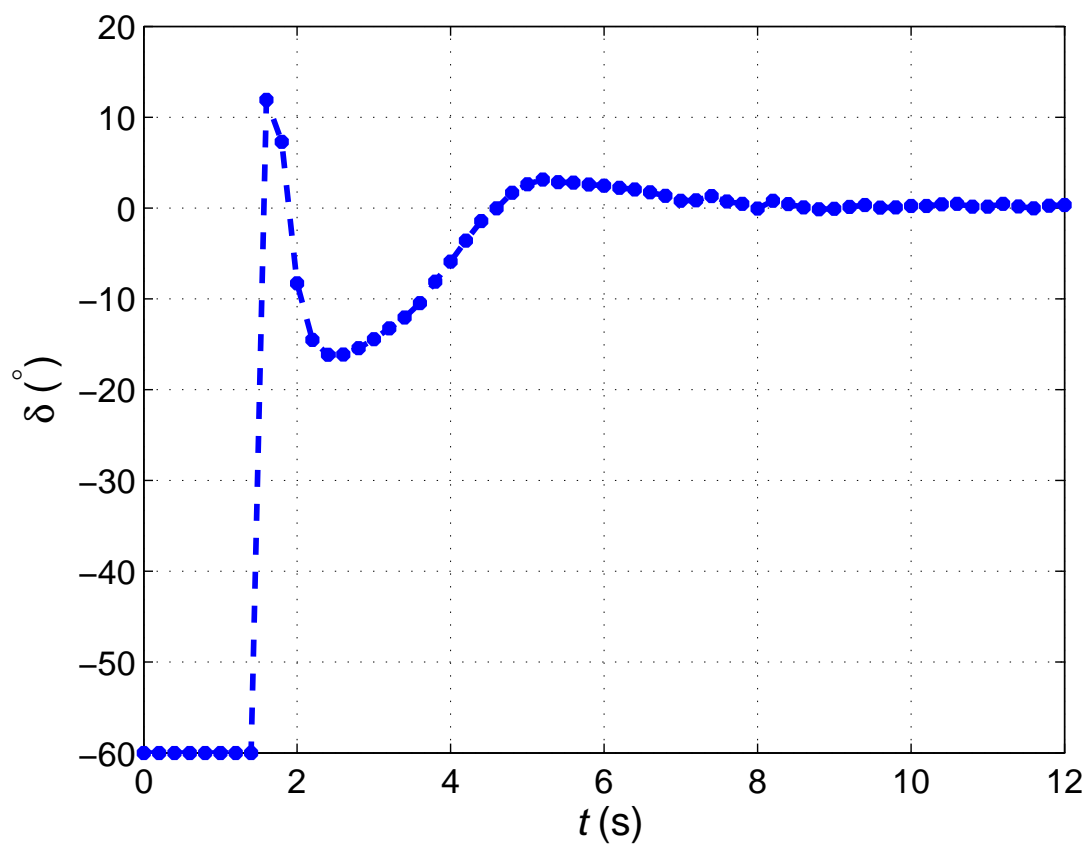
Figure 6.14: The trajectory of tail angle in the passivity-based stabilization experiment with $K_c = 3$.
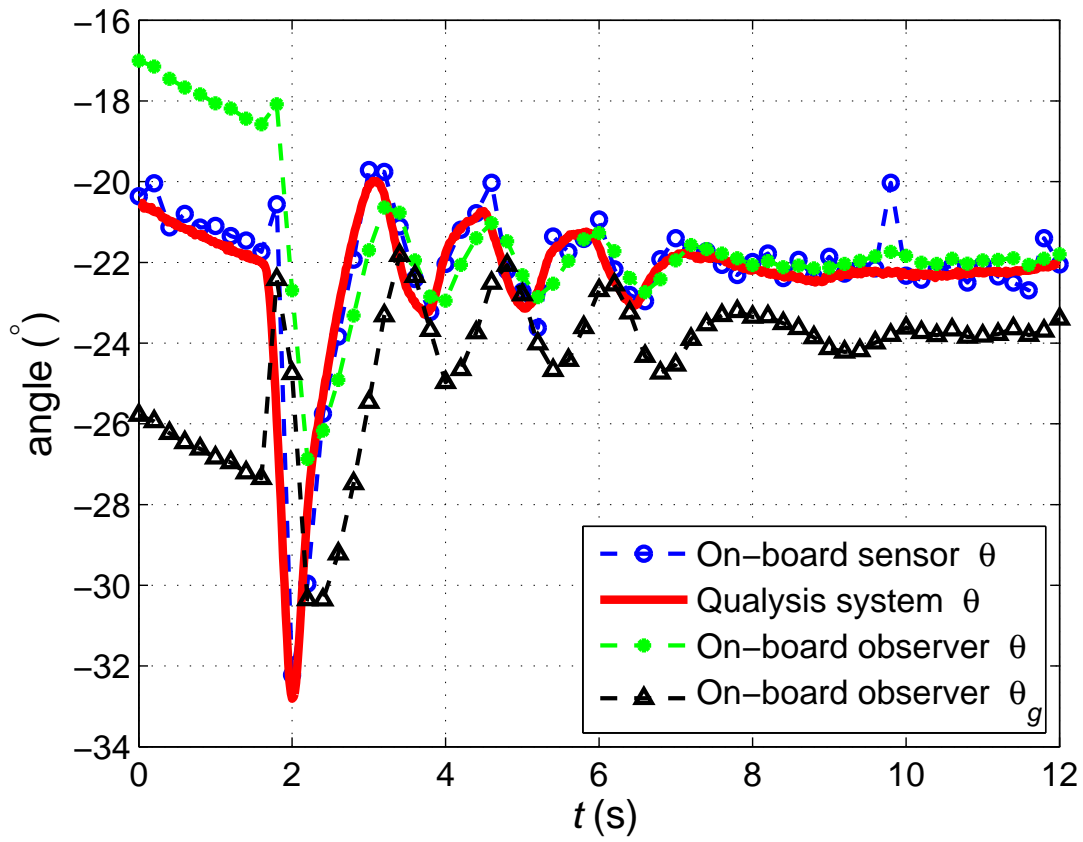
Figure 6.15: The trajectories of pitch angle and gliding angle in the passivity-based stabilization experiment with $K_c = 1$.
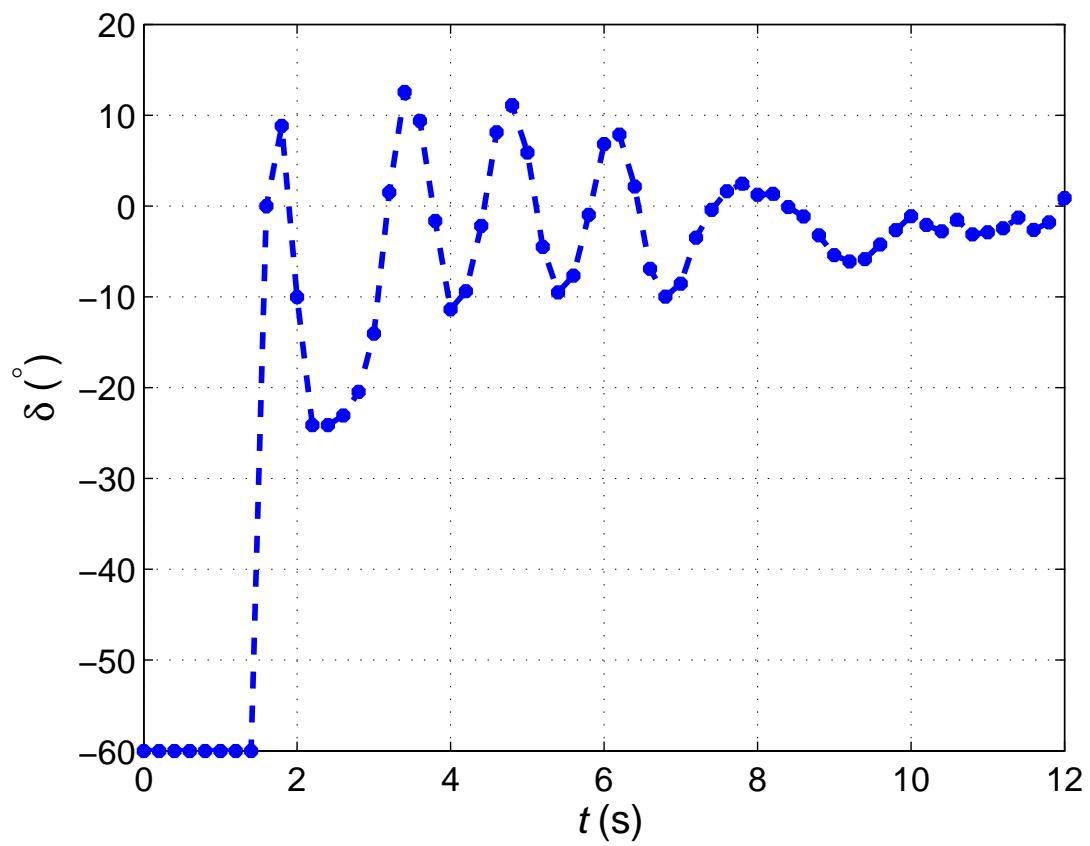
Figure 6.16: The trajectory of tail angle in the passivity-based stabilization experiment with $K_c = 1$.

performance. We find that with a larger K_P and K_I, the arising time of the gliding angle trajectory is reduced, however, with the cost of larger overshoot and oscillation amplitude, which shows similar tradeoff in tuning the passivity-based controller gain. Furthermore, in the comparison of the experimental results between using P/PI control and passivity-based control, the differences in arising time, percent overshoot, and oscillation time show that passivity-based controller has an overall better performance than the P/PI controller, especially in terms of the balance between convergence speed and control effort.
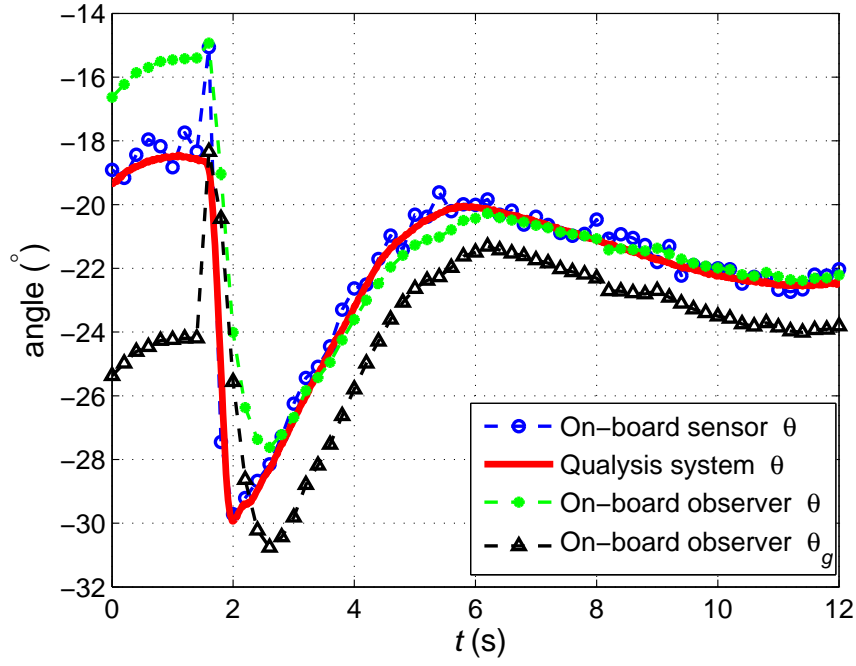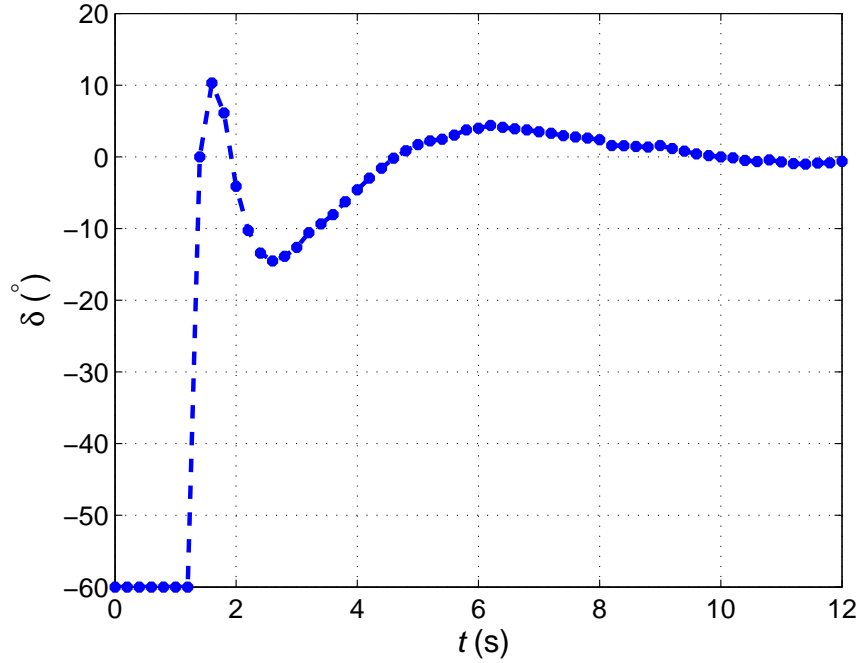
Figure 6.17: The trajectories in the comparative stabilization experiments with a proportional controller with $K_P = 2$. (a) pitch angle and gliding angle; (b) tail angle.
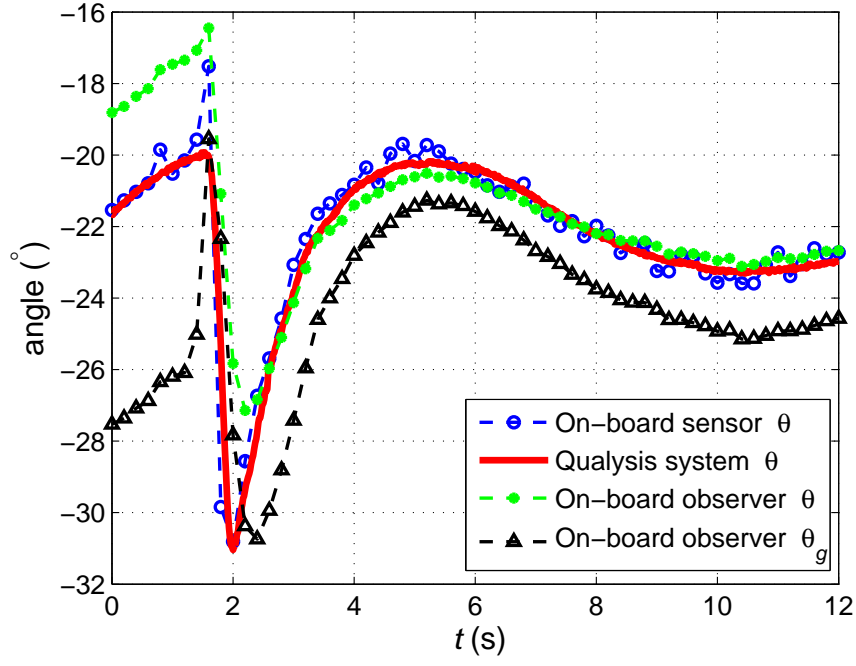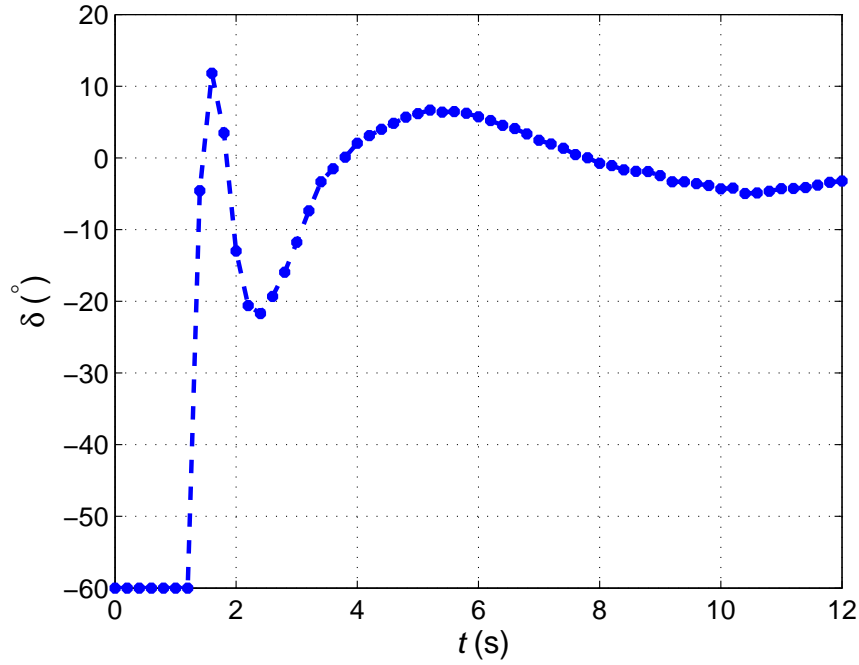
Figure 6.18: The trajectories in the comparative stabilization experiments with a proportional controller with $K_P = 3$. (a) pitch angle and gliding angle; (b) tail angle.
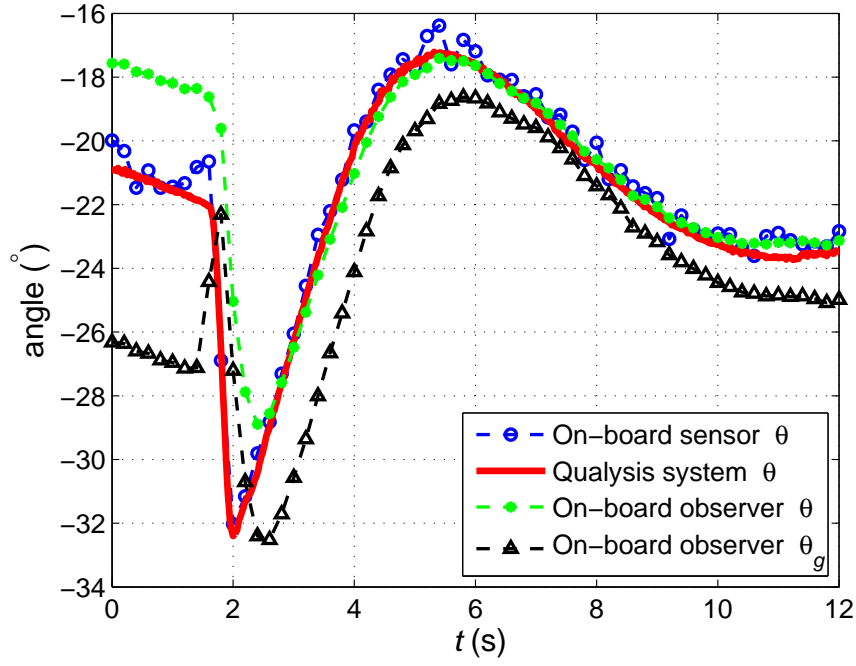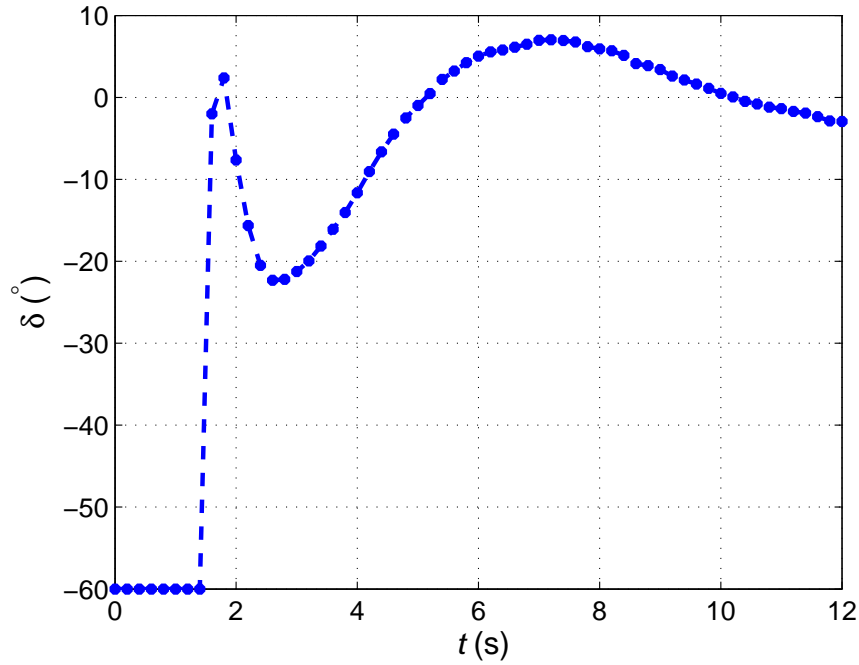
Figure 6.19: The trajectories in the comparative stabilization experiments with a PI controller with $K_P = 2, K_I = 1$. (a) pitch angle and gliding angle; (b) tail angle.
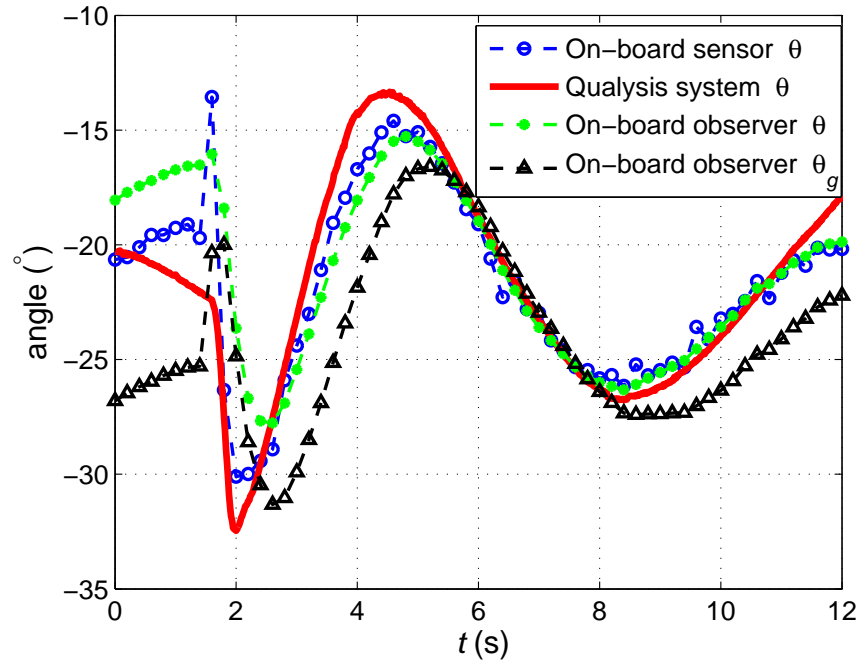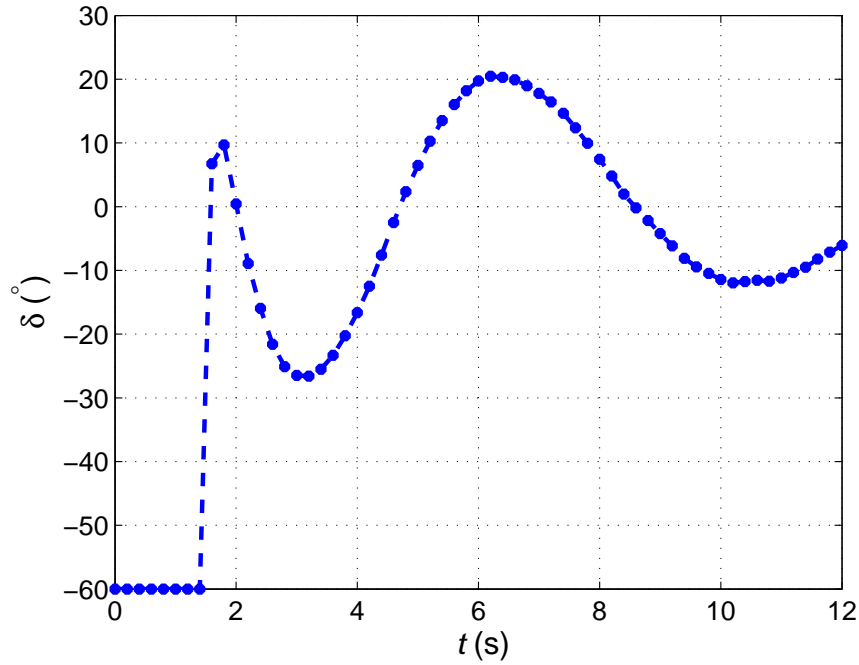
Figure 6.20: The trajectories in the comparative stabilization experiments with a PI controller with $K_P = 2, K_I = 3$. (a) pitch angle and gliding angle; (b) tail angle.

Chapter 7

Yaw Stabilization Using Sliding Mode Control

7.1   Problem Statement

Steady gliding motion is the most commonly used profile for underwater gliders, providing the capability of sampling water in the field while saving energy at the same time. Setting the right hand side of Eqs. (3.2)–(3.5) to zero, we can solve those equations for the steady glide path given a fixed movable mass displacement r_p and excess mass m_0, with zero tail angle [80]. Due to the existence of ambient currents or disturbances, the robot is susceptible to yaw deviation from its desired direction, beside the sagittal-plane perturbation discussed in Chapter 6, which makes yaw angle stabilization very important.

For succinctness, we first rewrite the system dynamics Eqs. (3.2)–(3.5) in a compact form

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}) + \Delta_1(\boldsymbol{x}) + g(\boldsymbol{x})(u + \Delta_2(t, \boldsymbol{x}, u)) \tag{7.1}$$

$$y = h(\boldsymbol{x}) = \psi \tag{7.2}$$

where $\boldsymbol{x}$ is the system state, $\boldsymbol{x} = \begin{bmatrix} \phi & \theta & \psi & v_1 & v_2 & v_3 & \omega_1 & \omega_2 & \omega_3 \end{bmatrix}^T$, $u = \delta$ is the tail angle, which is the control input in the current setting, and $\Delta_1(\boldsymbol{x})$ and $\Delta_2(t, \boldsymbol{x}, u)$ represent uncertainties.

The system output is chosen to be the yaw angle ψ. The function $g(\boldsymbol{x})$ is dependent on the state

$$g(\boldsymbol{x}) = \begin{bmatrix} \mathbf{0}_{3\times 1} \\ -\dfrac{1}{2m_1}\rho V^2 SC_{SF}^{\delta}\cos\alpha\sin\beta \\ \dfrac{1}{2m_2}\rho V^2 SC_{SF}^{\delta}\cos\beta \\ -\dfrac{1}{2m_3}\rho V^2 SC_{SF}^{\delta}\sin\alpha\sin\beta \\ -\dfrac{1}{2J_1}\rho V^2 SC_{M_Y}^{\delta}\sin\alpha \\ 0 \\ \dfrac{1}{2J_3}\rho V^2 SC_{M_Y}^{\delta}\cos\alpha \end{bmatrix} \tag{7.3}$$

The yaw motion stabilization problem is how to design the state-feedback controller for the tail angle δ, to stabilize the yaw angle ψ to a desired value r in the presence of disturbances.

7.2 Sliding Mode Control for Yaw Stabilization

Sliding model control is a practical nonlinear control method, especially for robust stabilization [96]. In this dissertation, we adopt sliding mode control for tail-enabled yaw motion stabilization of gliding robotic fish. Basic control design procedure follows the approach described in [96]. We further construct a simplified controller requiring only partial state feedback information based on the derived controller [99].

In order to obtain the relative degree of the system, we take the time derivatives of $h(\boldsymbol{x})$

$$\dot{h}(\boldsymbol{x}) = \dot{\psi} = \sin\phi\sec\theta\,\omega_2 + \cos\phi\sec\theta\,\omega_3 = L_f h(\boldsymbol{x}) \tag{7.4}$$

$$\ddot{h}(\boldsymbol{x}) = \ddot{\psi} = L_f^2 h(\boldsymbol{x}) + L_{\Delta_1} L_f h(\boldsymbol{x}) + L_g L_f h(\boldsymbol{x})(u + \Delta_2(t,\boldsymbol{x},u)) \tag{7.5}$$

where $L_f h(\cdot)$ represents the Lie derivative of function $h(\cdot)$ with respect to the vector field $f(\cdot)$ [96], and $L_f^2 h(\boldsymbol{x})$ is equal to $L_f L_f h(\boldsymbol{x})$.

The results that $\dot{h}(\boldsymbol{x})$ does not depend on control input u and $\ddot{h}(\boldsymbol{x})$ does, imply that

$$L_g h(\boldsymbol{x}) = 0 \tag{7.6}$$

$$L_g L_f h(\boldsymbol{x}) \neq 0 \tag{7.7}$$

so the relative degree of the system $\rho_{sys} = 2$.

From Frobenius Theorem [100], there exists a transform function $T(\boldsymbol{x})$, which converts the original system to the normal form with system states $\begin{bmatrix} \boldsymbol{\eta} & \boldsymbol{\xi} \end{bmatrix}^T$.

$$\begin{bmatrix} \boldsymbol{\eta} \\ \boldsymbol{\xi} \end{bmatrix} = \begin{bmatrix} \Phi(\boldsymbol{x}) \\ \Psi(\boldsymbol{x}) \end{bmatrix} = \begin{bmatrix} p_1(\boldsymbol{x}) \\ \vdots \\ p_7(\boldsymbol{x}) \\ \hline h(\boldsymbol{x}) \\ L_f h(\boldsymbol{x}) \end{bmatrix} = T(\boldsymbol{x}) \tag{7.8}$$

where

$$\frac{\partial p_i}{\partial \boldsymbol{x}} g(\boldsymbol{x}) = 0, \text{ for } i = 1, 2, \ldots, 7 \tag{7.9}$$

$$\dot{\xi}_1 = \xi_2 \tag{7.10}$$

Let r denote the reference trajectory for the yaw angle, which would be a constant number in

117

the stabilization problem. Take

$$\mathscr{R} = \begin{bmatrix} r \\ \dot{r} \end{bmatrix} \tag{7.11}$$

The yaw error vector $\boldsymbol{e}$ is

$$\boldsymbol{e} = \boldsymbol{\xi} - \mathscr{R} = \begin{bmatrix} \xi_1 - r \\ \xi_2 - \dot{r} \end{bmatrix} \tag{7.12}$$

Then the error dynamics is expressed as

$$\dot{\boldsymbol{\eta}} = f_0(\boldsymbol{\eta}, \boldsymbol{\xi}) \tag{7.13}$$

$$\dot{e}_1 = e_2 \tag{7.14}$$

$$\dot{e}_2 = L_f^2 h(\boldsymbol{x}) + L_{\Delta_1} L_f h(\boldsymbol{x})$$

$$+ L_g L_f h(\boldsymbol{x})(u + \Delta_2(t, \boldsymbol{x}, u)) - \ddot{r} \tag{7.15}$$

Assume that $\dot{\boldsymbol{\eta}} = f_0(\boldsymbol{\eta}, \boldsymbol{\xi})$ is bounded-input-bounded-state stable with $\boldsymbol{\xi}$ as the input. Then we design a sliding manifold

$$s = e_2 + k_0 e_1 \tag{7.16}$$

where k_0 is a positive constant.

Sliding mode control can be taken as the following to cancel the known terms as in feedback

118

linearization,

$$u = -\frac{1}{L_g L_f h(\boldsymbol{x})}\left(k_0 e_2 + L_f^2 h(\boldsymbol{x}) - \ddot{r}\right) + v \tag{7.17}$$

where v is the switching component, and

$$L_f^2 h(\boldsymbol{x}) = \frac{\partial \dot{\psi}}{\partial \boldsymbol{x}} f \tag{7.18}$$

$$L_g L_f h(\boldsymbol{x}) = \frac{\partial \dot{\psi}}{\partial \boldsymbol{x}} g \tag{7.19}$$

$$\frac{\partial \dot{\psi}}{\partial \boldsymbol{x}} = \begin{bmatrix} \cos\phi \sec\theta \, \omega_2 \\ \sec\theta \tan\theta (\cos\phi \omega_2 + \cos\phi \omega_3) \\ \boldsymbol{0}_5 \times 1 \\ \sin\phi \sec\theta \\ \cos\phi \sec\theta \end{bmatrix}^T \tag{7.20}$$

Or we can take the controller as the pure switching component,

$$u = v \tag{7.21}$$

Then in either case, the $\dot{s}$-equation can be written as

$$\dot{s} = L_g L_f h(\boldsymbol{x}) v + \Delta(t, \boldsymbol{x}, v) \tag{7.22}$$

Suppose that the uncertainty satisfies the following inequality,

$$\left|\frac{\Delta(t, \boldsymbol{x}, v)}{L_g L_f h(\boldsymbol{x})}\right| \le \rho(\boldsymbol{x}) + \kappa_0 |v|, \quad 0 \le \kappa_0 \le 1 \tag{7.23}$$

119

where $\rho(\boldsymbol{x})$ represents the upper bound of the uncertainty related to the system states.

Design the switching component

$$v = -\gamma(\boldsymbol{x})sat(s/\varepsilon) \tag{7.24}$$

where $sat(s/\varepsilon)$ is a high-slope saturation function with a small constant ε, used to reduce chattering, and $\gamma(\boldsymbol{x}) \geq \rho(\boldsymbol{x})/(1 - \kappa_0) + \gamma_0$ with constant γ_0 to deal with the non-vanishing disturbance $\Delta(t, \boldsymbol{x}, v)$ if that is the case.

In this dissertation, we choose

$$\gamma(\boldsymbol{x}) = k_1 \|\boldsymbol{x} - \boldsymbol{x_e}\|_2^{k_2} + k_3 \tag{7.25}$$

where $\boldsymbol{x}_e$ is the system equilibrium point, which can be calculated given a steady gliding profile [82], and k_1, k_2, k_3 are tunable controller parameters, determined by the uncertainty type and also capable of adjusting closed-loop dynamics performance.

Based on the fact that the yaw angle ψ is the state we really care about, we further simplify the sliding mode controller (Eq. (7.21),(7.24),(7.25)) to

$$u = -(k_1 \|\psi - \psi_e\|_2^{k_2} + k_3)sat(s/\varepsilon) \tag{7.26}$$

where the sliding mode controller only requires the information of yaw angle, making it easy to implement. The effect of this simplification on stability can be compensated by increasing controller parameters k_1, k_2 and especially the non-zero constant k_3. Although this will increase the tracking error in general, the controller implementation becomes much more simpler. The effectiveness of the designed controller will be evaluated in simulation and experiments in the

120

Table 7.1: Parameters of gliding robotic fish "Grace".

Parameter	Value	Parameter	Value
m_1	8.0 kg	m_2	19.8 kg
m_3	10.8 kg	$\bar{m}$	1.6 kg
C_{D0}	0.45	C_D^α	17.59 rad^{-2}
C_{FS}^β	-2 rad^{-1}	C_{FS}^δ	1.5 rad^{-1}
C_{L0}	0.075	C_L^α	19.58 rad^{-1}
J_1	1.27 kg·m^2	J_2	0.08 kg·m^2
J_3	0.13 kg·m^2	C_{M0}	0.0076 m
C_{MR}^β	-0.3 m/rad	C_{MP}^α	0.71 m/rad
C_{MY}^β	5 m/rad	C_{MY}^δ	-0.2 m/rad
K_{q1}	-0.16 m·s/rad	K_{q2}	-0.80 m·s/rad
K_{q3}	-0.16 m·s/rad	S	0.019 m^2

following sections.

7.3 Simulation

To evaluate the designed sliding mode controller, simulation is carried out in Matlab. The parameters used in the simulation is based on the gliding robotic fish prototype "Grace", obtained via scaling analysis on our previously developed prototype in [82] (Tab. 7.1).

The initial state values for the simulation are

$$\phi = 0, \qquad \theta = -30°, \qquad \psi = 30°, \qquad v_1 = 0.27 \, m/s, \qquad v_2 = 0,$$

$$v_3 = 0, \qquad \omega_1 = 0, \qquad \omega_2 = 0, \qquad \omega_3 = 0.$$

The controller parameters used in the simulation are

$$k_0 = 10, \qquad k_1 = 10/30/50, \qquad k_2 = 0.8/1/1.2, \qquad k_3 = 0.01.$$

Figure 7.1: Yaw angle trajectory with respect to different controller parameters k_1.

Figure 7.2: Tail angle trajectory with respect to different controller parameters k_1.

Figure 7.3: Sideslip angle trajectory with respect to different controller parameters k_1.

Figure 7.4: Yaw angle trajectory with respect to different controller parameters k_2.

In Figs. 7.1–7.3, the trajectories of yaw angle ψ, controller command δ, and sideslip angle β are shown for varying controller parameter k_1 while Figs. 7.4–7.6 shows the simulation results when varying controller parameter k_2. From the results, we can see that under proper controller parameter setting, the sliding mode controller is able to regulate the yaw angle, which is deviated from the desired orientation, back to the original, zero angle within a relative short time. Consequently, the trajectory of the glider is adjusted to the desired path, with the heading orientation being zero degree, as shown in Fig. 7.7. From the comparison under different controller parameters, we find that k_1 and k_2 control the balance between response speed and controller effort. With larger k_1 and smaller k_2, the system responses faster and requires bigger control output amplitude.
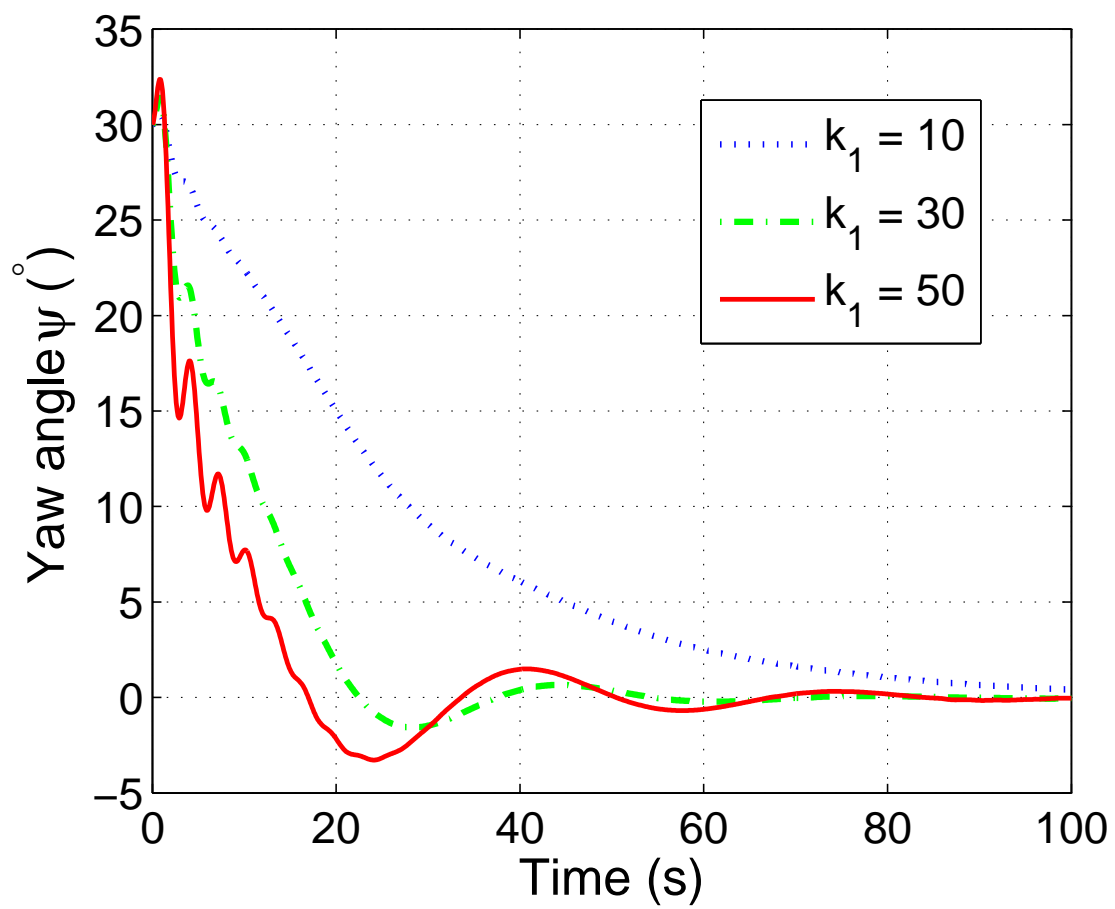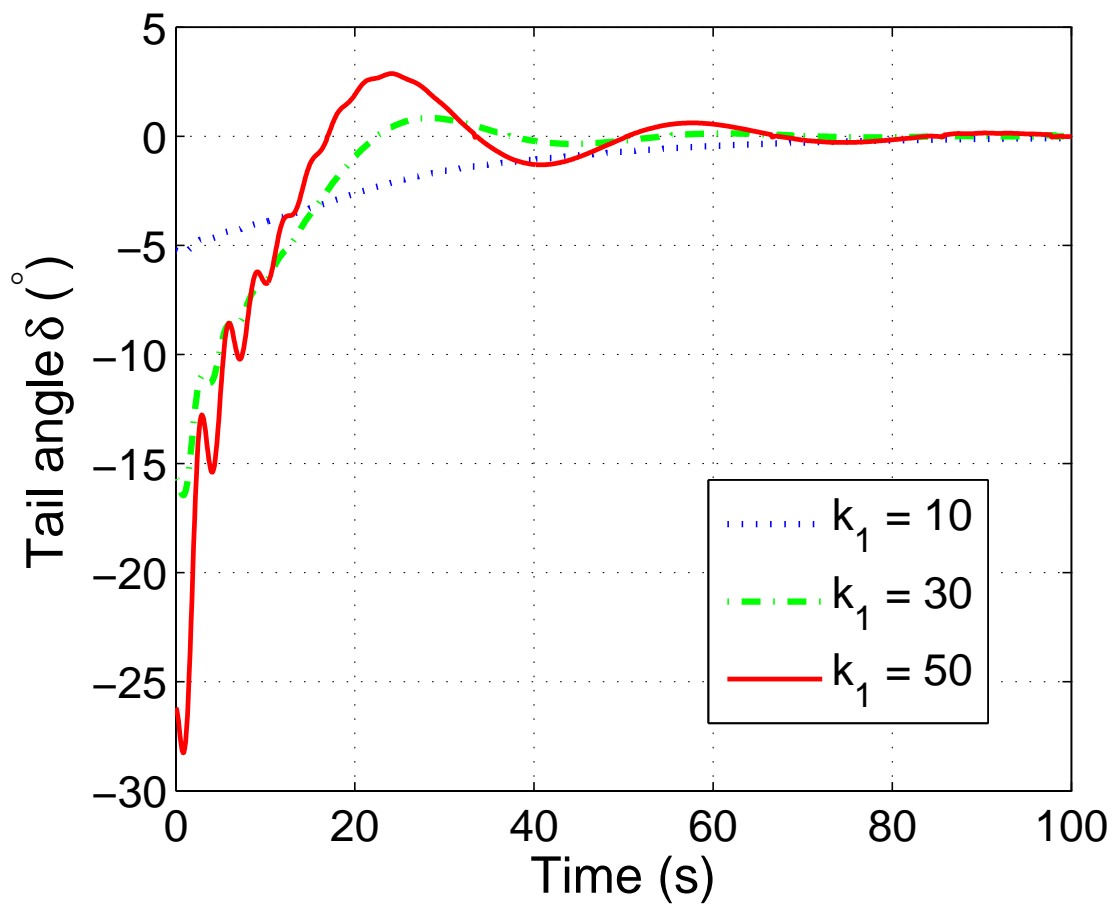
Figure 7.5: Tail angle trajectory with respect to different controller parameters k_2.

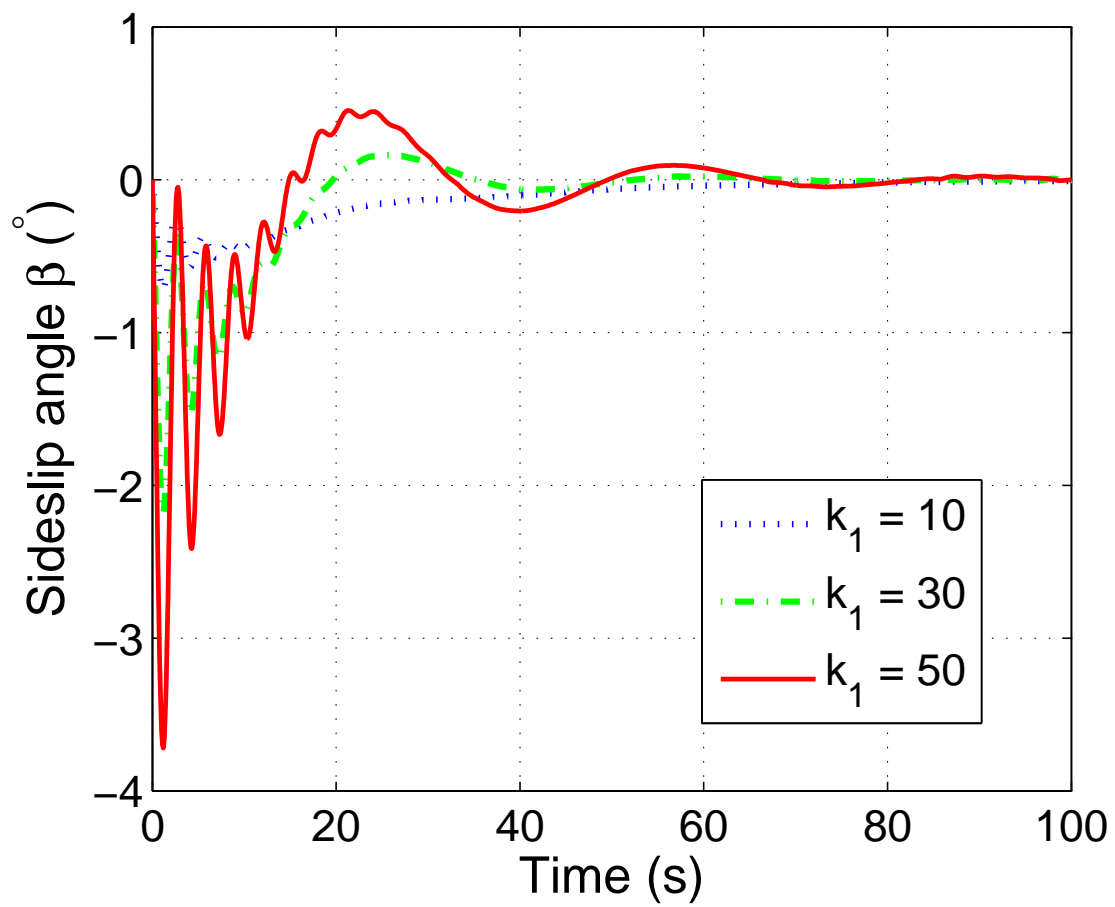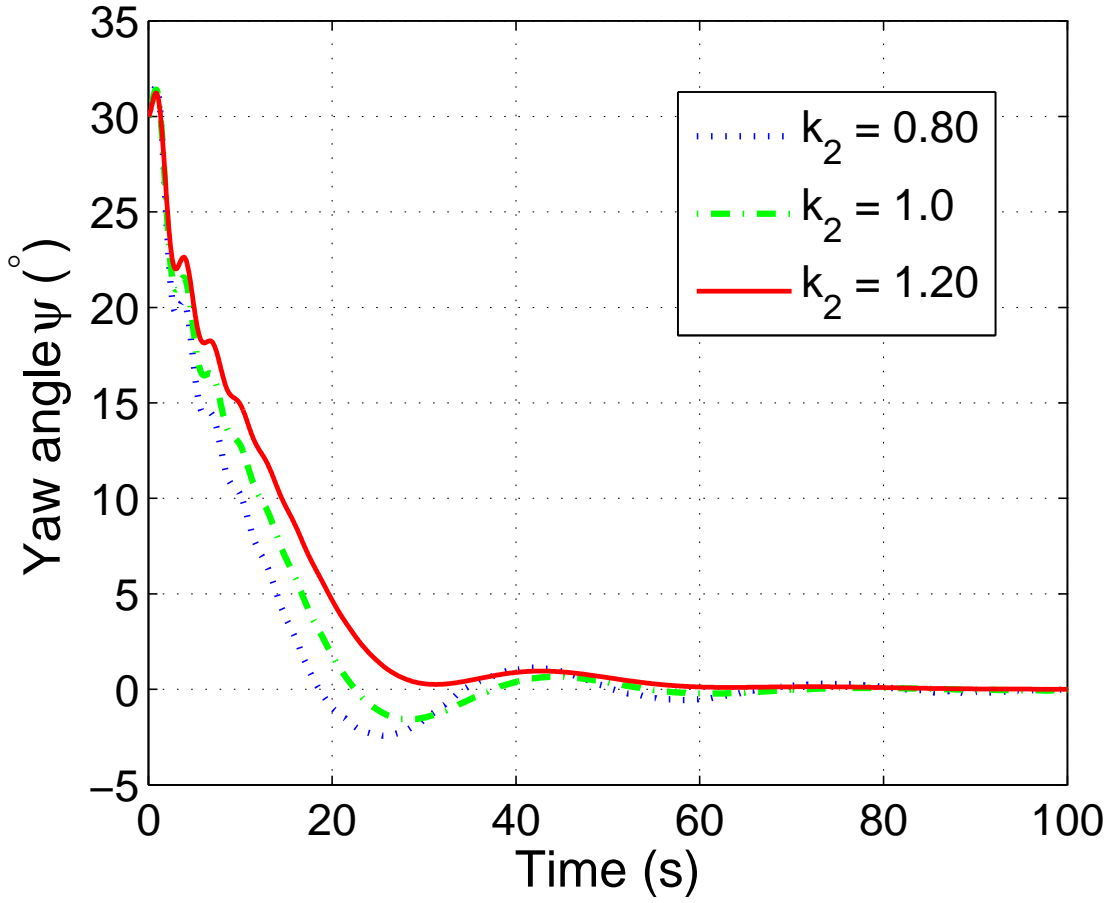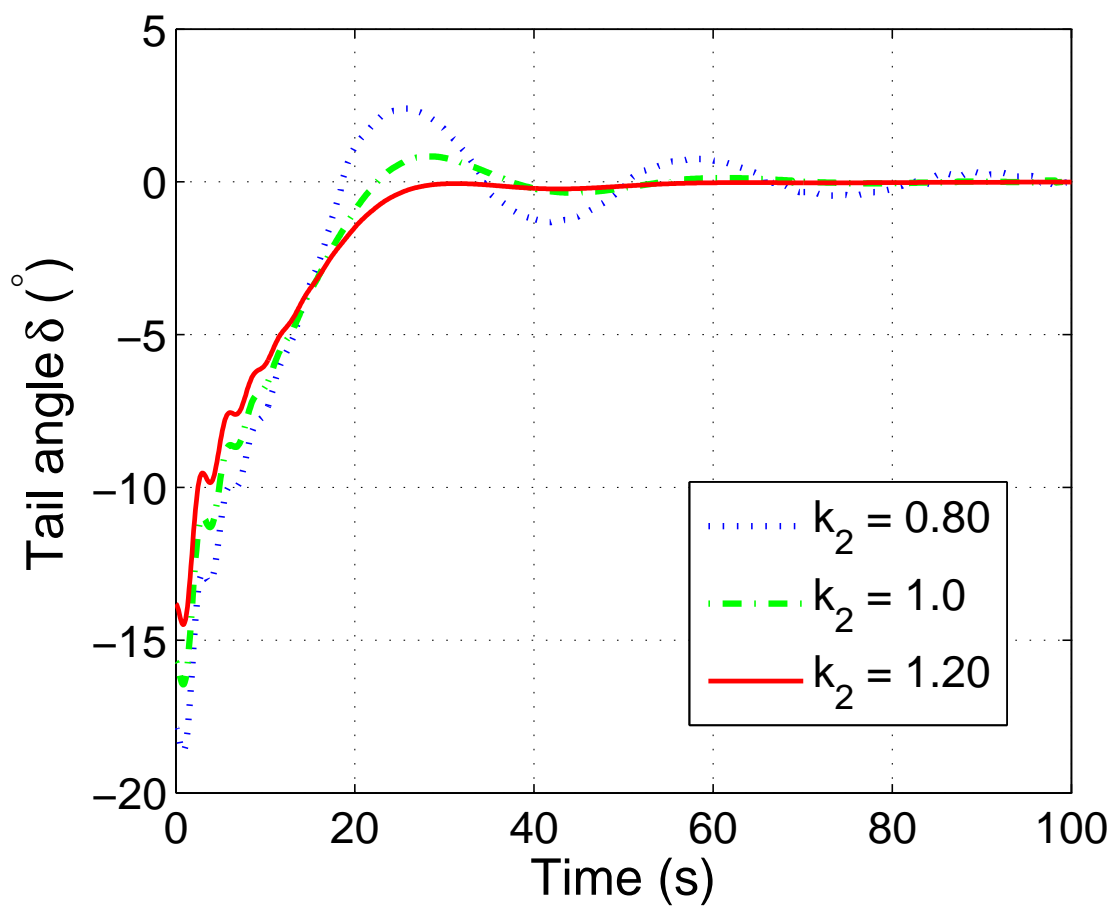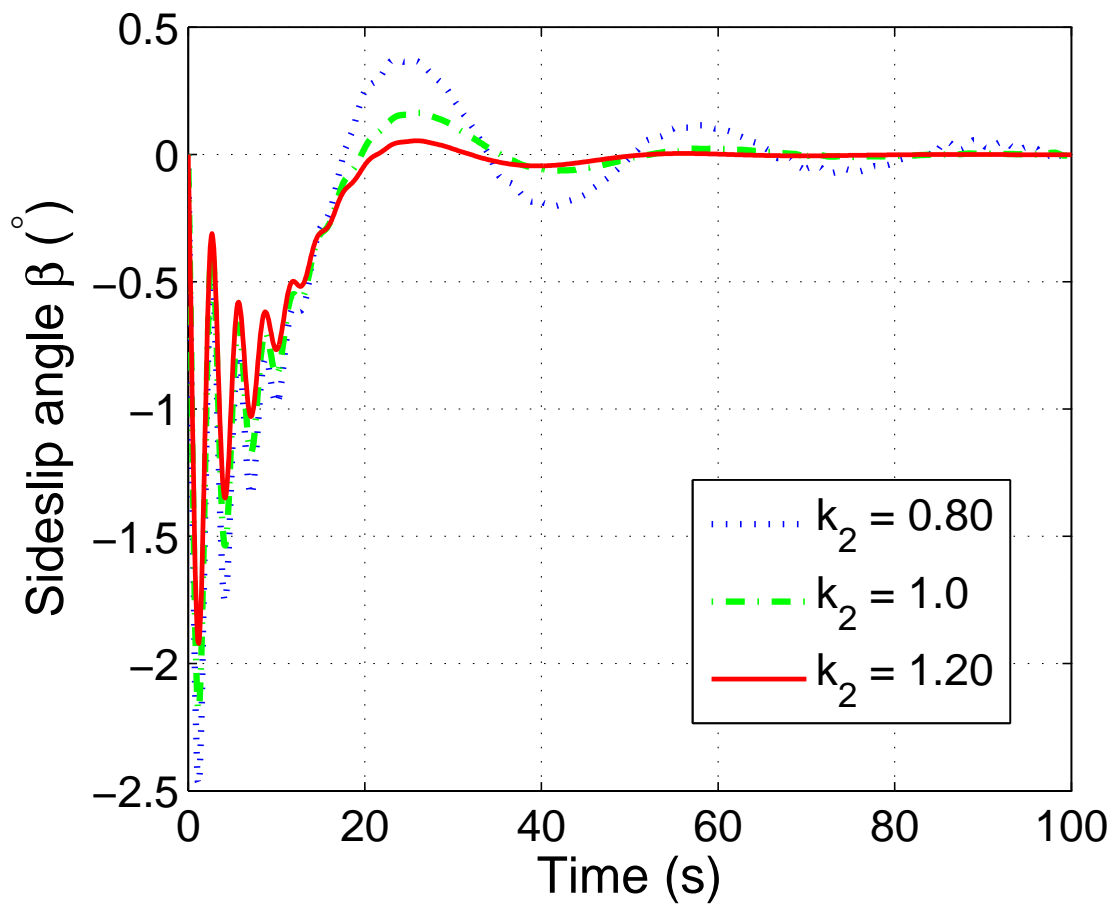Figure 7.6: Sideslip angle trajectory with respect to different controller parameters k_2.
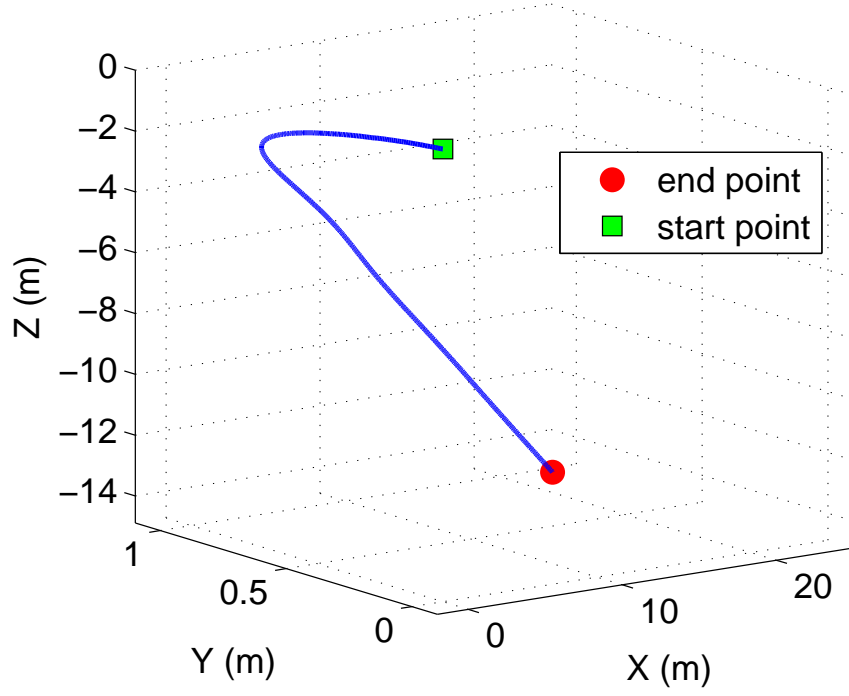
(a) Three-dimensional view
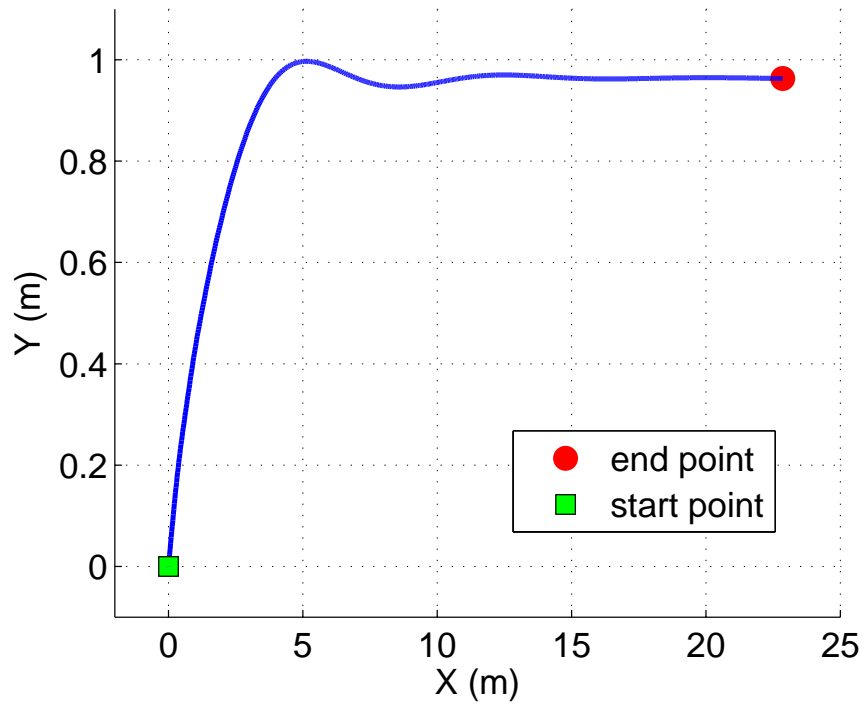
(b) Top view for X-Y plane

Figure 7.7: Trajectory of the gliding robotic fish for controller parameters $k_1 = 30$, $k_2 = 1$.

Regarding parameter k_3, as in the sliding mode design principle, it should balance the steady-state error and uncertainty tolerance capability. With a larger k_3, the controller is able to work under larger uncertainty while leading to bigger steady-state error.

7.4   Experimental Results

We implement the sliding mode controller on our lab-developed gliding robotic fish "Grace", and conduct yaw angle stabilization experiments in a large indoor water tank that measures 15-foot long, 10-foot wide, and 4-foot deep. Yaw angle is calculated in the on-board micro-controller using real-time sensor feedback from compass, accelerometer, and gyro. The sampling time for sensor reading is 100 ms. A discrete-time low pass filter is applied directly after the raw data to smooth out high frequency noise, with the difference equation

$$o(n) = a \times i(n) + (1 - a) \times o(n - 1) \tag{7.27}$$

where $i(n)$ and $o(n)$ are the low-pass filter's input and output at time n, respectively. Filter parameter $a \in (0, 1)$ is selected to be 0.5 in the experiment.

In the experiments, we release the gliding robotic fish on the water surface with a deviated yaw angle. Then the robot starts to pump water in and translate the movable mass forward to the set value, $m_0 = 40$ g, and $r_p = 5$ mm. After the robot is fully submerged and enters gliding down motion, the sliding mode controller is applied to regulate the yaw angle to the desired value, defined as zero degree. The robot motion in the whole process is recorded with a top-view camera hung on a guiding rail and a side-view underwater camera placed in the tank (Fig. 7.8).

The sensor reading is stored in the on-board memory and sent back to the laptop through
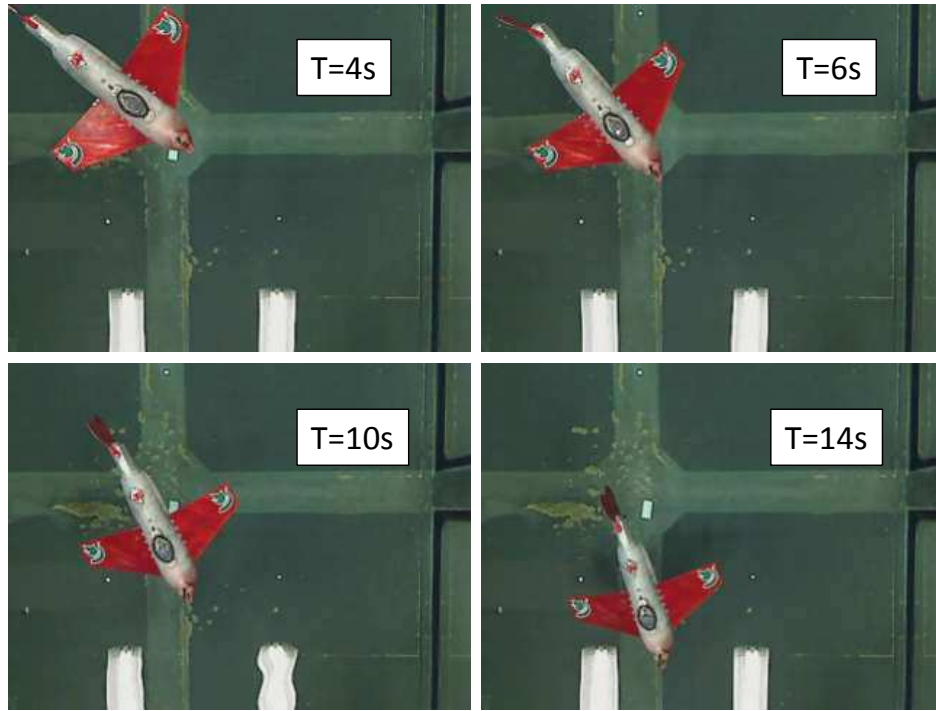
(a) Top view
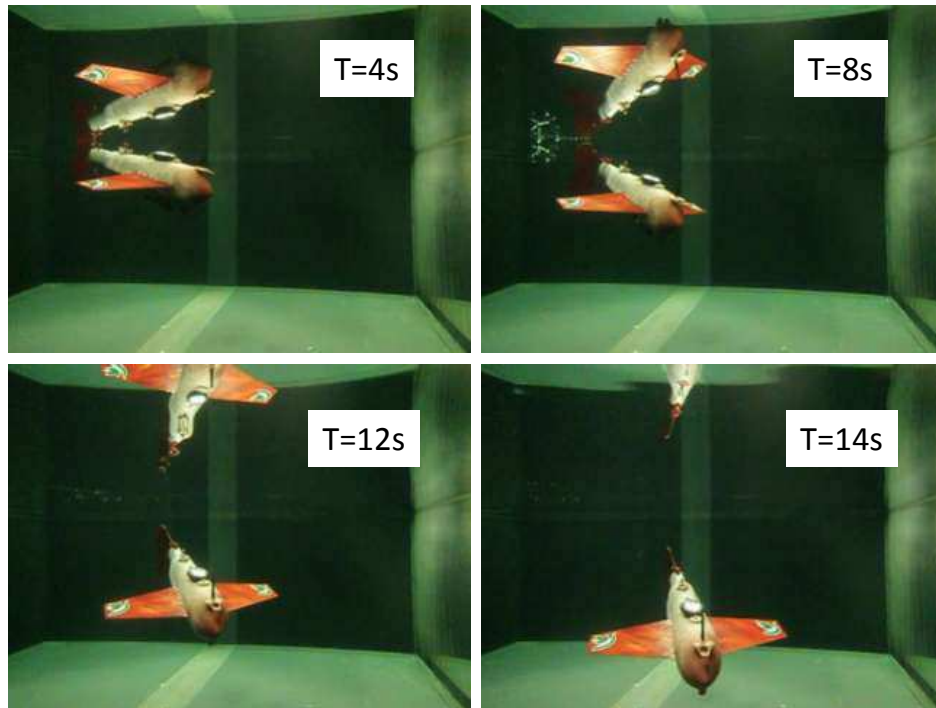
(b) Side view

Figure 7.8: Snapshots of controlled motion with yaw stabilization using sliding mode controller, under parameters $k_1 = 50$, $k_2 = 1$.
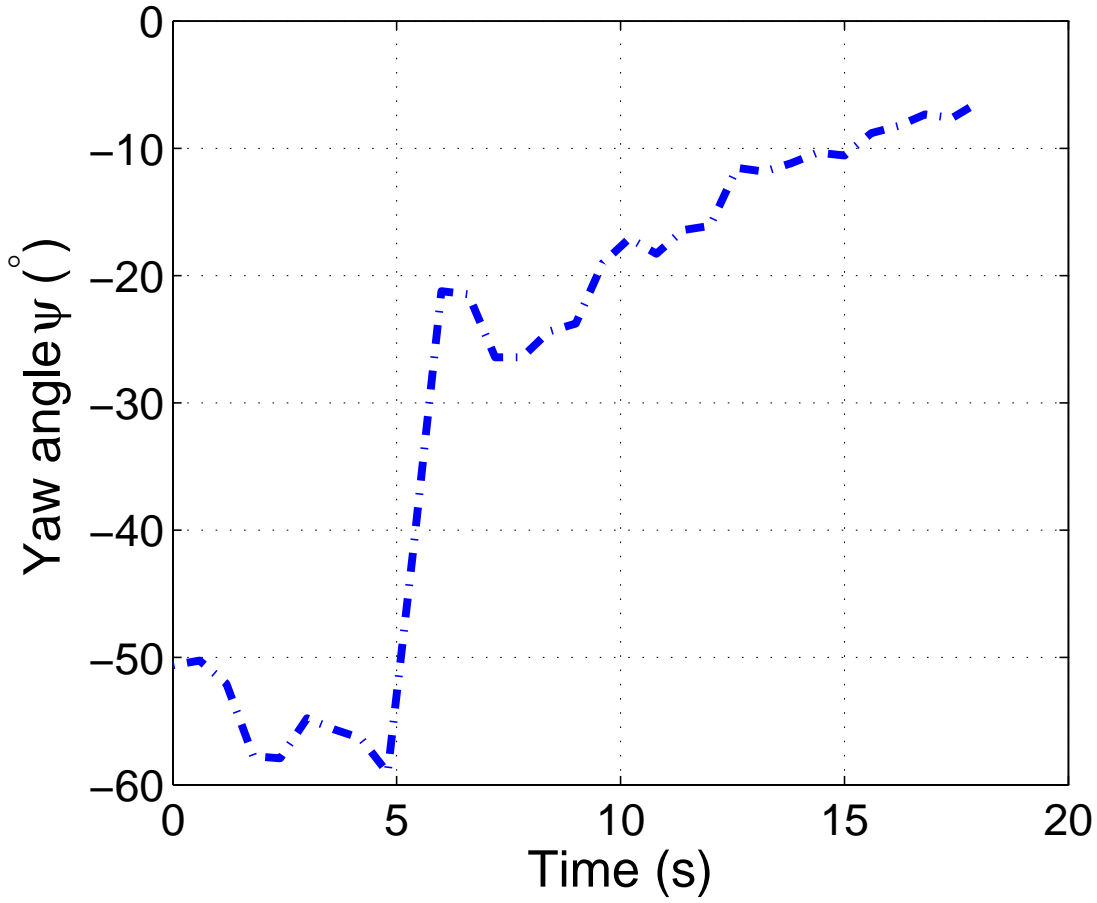
Figure 7.9: Yaw angle trajectory using "Grace"

wireless communication when the robot surfaces. Due to the range limitation of the tank, we conduct the experiment for about 15 seconds. That corresponds to the most of transient process, leaving the period that is close to the steady-state motion unrecorded.

In Figs. 7.9 and 7.10, we plot experimental results for both yaw angle ψ and tail angle δ. From the experiment, we can see that the proposed sliding mode controller is able to regulate the deviated yaw angle to the desired set value, with reasonable tail angle amplitude. There exist sensor noise and tail rotation dynamics, which contributes to the unsmooth yaw angle readings. More advanced sensor and filtering method could be used to solve the problem.
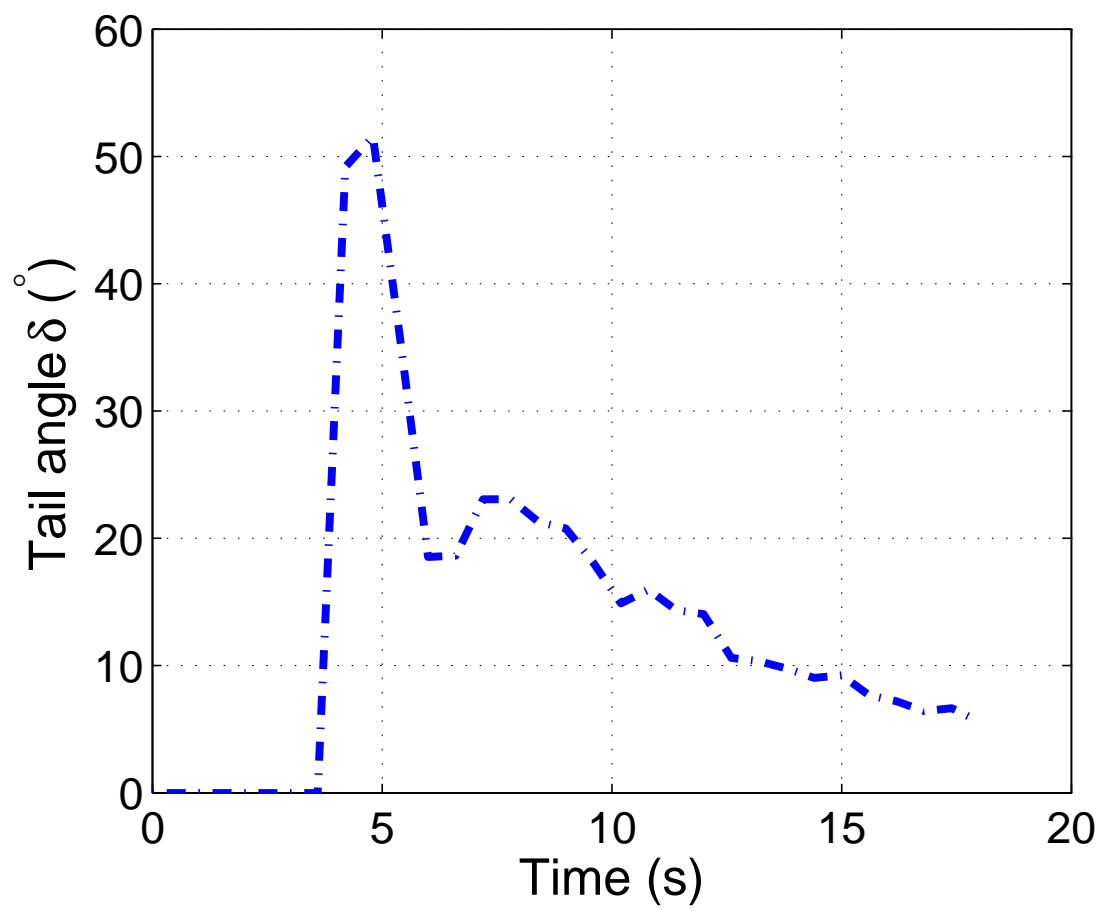
Figure 7.10: Tail angle trajectory using "Grace"

Chapter 8

Three-Dimensional Curve Tracking

8.1 Three-Dimensional Steady Spiral and Its Differential Geometry Features

The three-dimensional motion control for gliding robotic fish, in terms of curve tracking, is very challenging because the influences of the control inputs on the robot's locomotion are strongly nonlinear and coupled. It is more convenient to look into the influence of control inputs on the robot's differential geometry features, such as curvature and torsion, because we can examine the relationship between those geometric characteristic parameters and the control inputs by studying the steady-state spiral motions of the robot.

We decompose an arbitrary three-dimensional curve into a set of continuously evolving spirals. In this way, at any point of the space curve, there is an imaginary matching spiral curve with the same curvature and torsion. With this interpretation, instead of using the Euclidean positions, we will explore the task of three-dimensional curve tracking via designing and following continuously evolving spirals from the point of view of differential geometry [101].

First, Let us review the results of the steady spiral motion discussed in Chapter 4.

There are three control variables available to manipulate the robot's motion profile: the excess mass m_0, the position of the movable mass $\boldsymbol{r_p}$, and the tail angle δ. From [80] and [81], we know that when all three controls are fixed at non-zero values, the gliding robotic fish will perform three-

dimensional spiraling motion and finally enter a steady spiral, where the yaw angle ψ changes at a constant rate while the roll angle ϕ and pitch angle θ remain constant.

The dynamics of the spiral motion, derived from (3.2)-(3.5), can be presented in a compact form as

$$\dot{x}_s = f(x_s, u) = [f_i(x_s, u)]_{8 \times 1} \tag{8.1}$$

where $x_s = \begin{pmatrix} \phi & \theta & v_1 & v_2 & v_3 & \omega_1 & \omega_2 & \omega_3 \end{pmatrix}^T$, and control inputs $u = \begin{pmatrix} r_p & m_0 & \delta \end{pmatrix}^T$.

The steady-state spiraling equations can be obtained by setting time derivatives to zero in (8.1)

$$0 = f(x_s, u) = [f_i(x_s, u)]_{8 \times 1} \tag{8.2}$$

In a steady spiral, $R^T k$ is constant since

$$R^T k = R^T \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -\sin\theta \\ \sin\phi\cos\theta \\ \cos\phi\cos\theta \end{pmatrix} \tag{8.3}$$

The angular velocity has only one degree of freedom with ω_{3i} in Ox axis in the inertial frame

$$\omega_b = \omega_{3i}(R^T k) \tag{8.4}$$

Therefore, in the system of algebraic equations (8.2), there are nine independent variables (including control inputs) for describing the steady spiral motion: $\begin{pmatrix} \phi & \theta & \omega_{3i} & r_p & m_0 & \delta & V & \alpha & \beta \end{pmatrix}^T$. Hereafter we will use a state transformation on linear velocity variables for the sake of calculation

convenience

$$\boldsymbol{v_b} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix} = \boldsymbol{R}_{bv} \begin{pmatrix} V \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} V\cos\alpha\cos\beta \\ V\sin\beta \\ V\sin\alpha\cos\beta \end{pmatrix} \tag{8.5}$$

In the elementary differential geometry, a three-dimensional curve is captured by its curvature and torsion. The curvature κ is the amount by which a geometric object deviates from being flat, or the degree by which a geometric object bends, while torsion τ measures the departure of a curve from a plane, or how sharply a curve twists. Any time-trajectory of a smooth space curve can be completely described mathematically using curvature, torsion and velocity with Frenet-Serret formulas [102].

The geometric parameters (curvature, torsion, velocity) of a steady spiral can be expressed as

$$\kappa = \frac{r}{r^2 + c^2} \tag{8.6}$$

$$\tau = \frac{c}{r^2 + c^2} \tag{8.7}$$

where r is the steady spiral radius, and $2\pi c$ is the steady spiral pitch, or the vertical separation between two steady spirals. Furthermore,

$$r = \frac{V_h}{\omega_{3i}} \tag{8.8}$$

$$c = \frac{V_v}{\omega_{3i}} \tag{8.9}$$

where V_h and V_v are the horizontal velocity and vertical velocity, respectively, of the steady spiral motion.

We also have

$$V_h^2 + V_v^2 = V^2 \qquad (8.10)$$

From (8.6)-(8.10), the angular velocity ω_{3i} and the vertical velocity V_v can be described by the three geometric parameters, κ, τ, and V

$$\omega_{3i} = V\sqrt{\kappa^2 + \tau^2} \qquad (8.11)$$

$$V_v = V\sqrt{\frac{\tau^2}{\tau^2 + \kappa^2}} \qquad (8.12)$$

8.2 Influence of Control Inputs on Spiral Trajectory

As for the spiral motion, the three control inputs, including the movable mass displacement r_p, the net buoyancy m_0, and the tail angle δ, have different influences on both the steady-state motion profile and the transient dynamics. In this dissertation, we focus on the influences of control inputs on the steady-state spiral trajectory characteristics, which provide useful insight for path planning in three-dimensional curve tracking.

We study the relationship between three system control inputs and three trajectory character-istic parameters, including curvature κ, torsion τ and total speed V, which are used to completely describe any three-dimensional trajectory. With the system dynamic model in Chapter 3 and sys-tem parameters as in "Grace", we conduct simulation with different sets of values of system control inputs, and then record the corresponding steady-state spiral paths.

Fig. 8.1 shows the relationship between tail angle δ and the three trajectory characteristic parameters, while the net buoyancy m_0 and the displacement of movable mass r_p are fixed at

30 g and 0.5 cm, respectively. Simulation results of varying m_0 and r_p are shown in Figs. 8.2–8.3. From those figures, we see that all control inputs have significant influence on the motion profile, although the degree of influence varies. For example, δ and r_p have greater influence on κ and τ than m_0. Most relationships show monotonic trends, while non-monotonic relationships appear between κ/τ and m_0 (Fig.8.2a). The simulation results of the influences of control inputs on robot's spiral trajectory, provide insight into the capability of three-dimensional maneuvering as well as the feedback controller design for three-dimensional curve tracking.

In order to verify the relationship between the control inputs and the trajectory characteristic parameters of steady-state spirals, experiments are conducted using the prototype "Grace". The experiments are carried out in the Neutral Buoyancy Research Facility (NBRF), University of Maryland. The water tank measures 50 feet across and 25 feet deep.

In experiments, "Grace" are remotely controlled via XbeePro communication to perform spiral motions with different control input values (Fig. 8.4). The whole spiral process is recorded using a Qualysis underwater motion capture system. The motion capture system features 12 underwater cameras around the water tank, with 8 at a shallower depth and 4 at a deeper depth. Each camera captures the spiral motion from a different angle of view (Fig. 8.5). The robot is equipped with five markers, which the motion capture system uses to identify the rigid body. Some of the robot's states, such as linear and angular positions can be measured and outputted using the system (Fig. 8.6), and other system states, including translational and angular velocities can be estimated from those measurements.

Fig. 8.7a and Fig. 8.7b show the comparison results between the model prediction and the experimental results on the spiral curvature and torsion when varying δ from $20°$ to $50°$ with the m_0 and r_p fixed at 30 g and 0.5 cm, respectively. The results on the spiral total speed is not presented as the influence of the tail angle on that variable is not very obvious as shown in Fig. 8.1b.
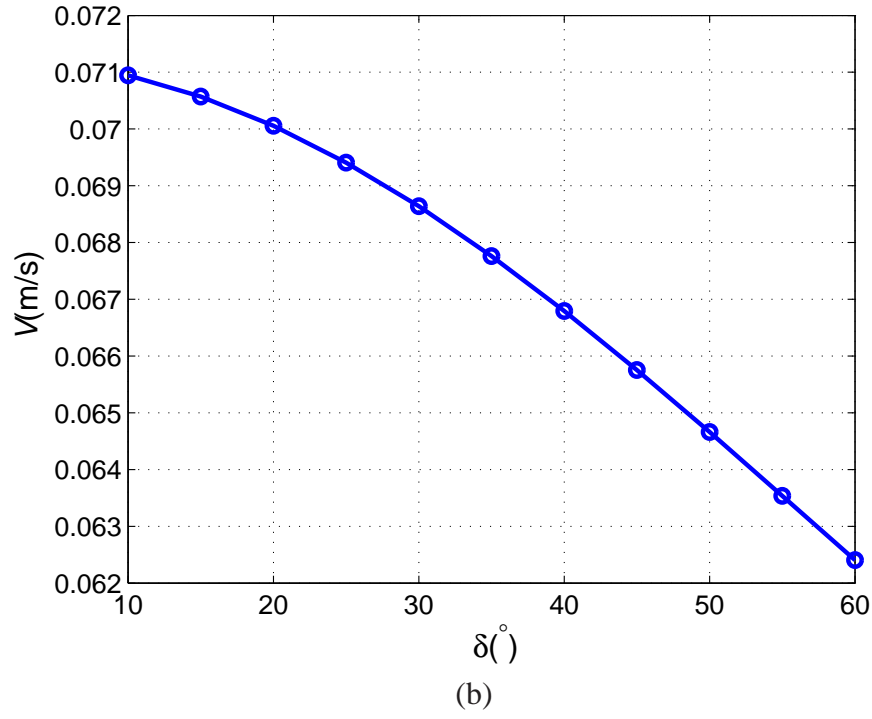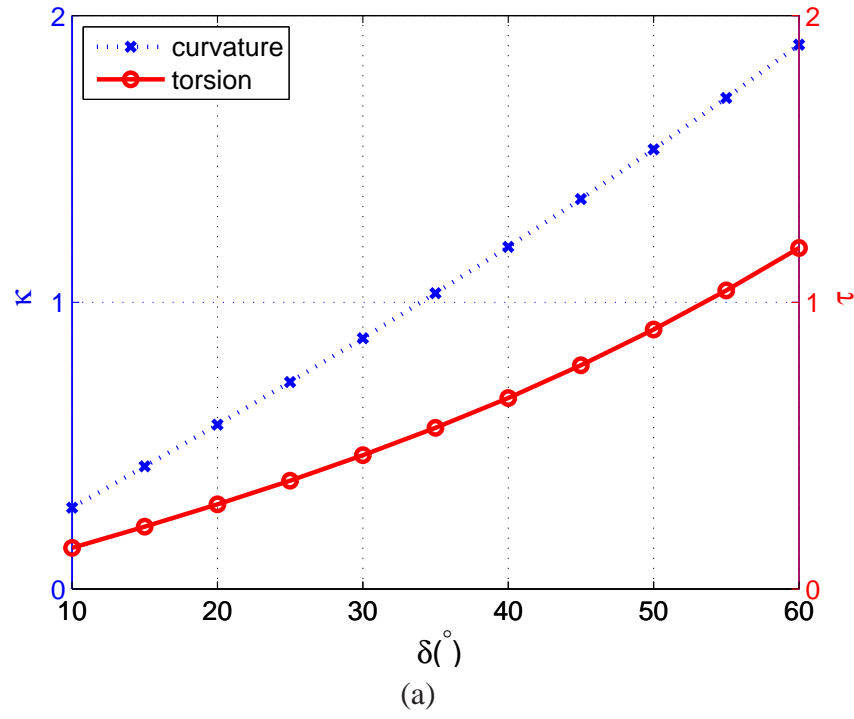
Figure 8.1: Trajectory characteristic parameters at different tail angles with $m_0 = 30$ g and $r_p = 0.5$ cm.
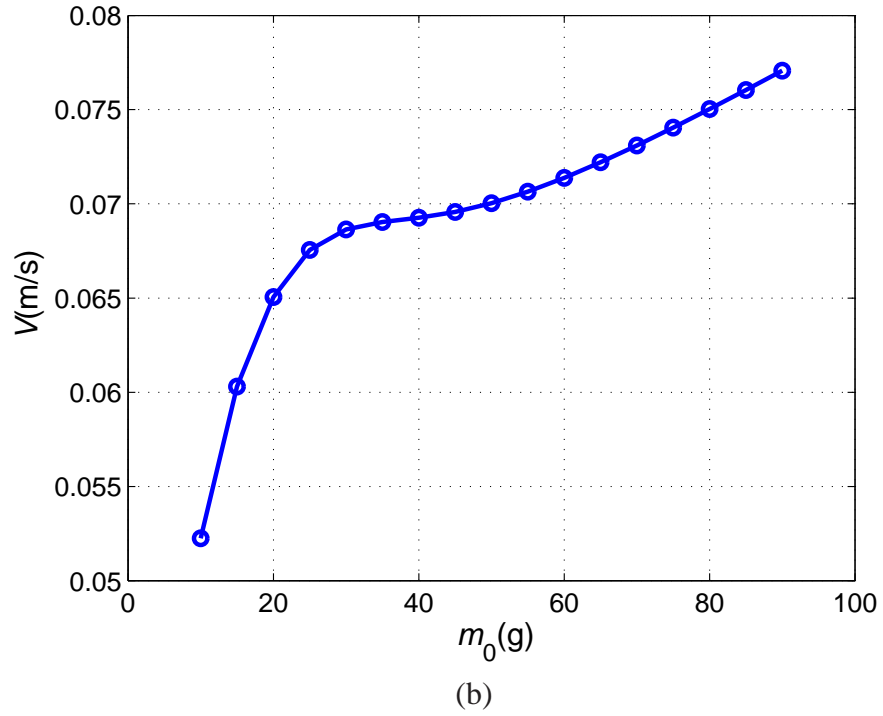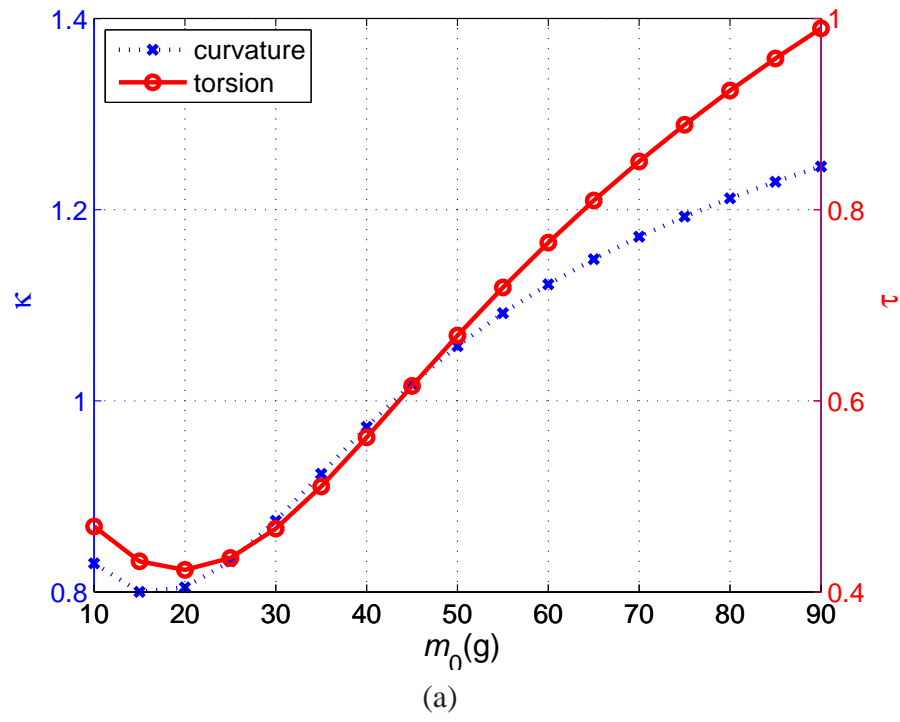
Figure 8.2: Trajectory characteristic parameters at different net buoyancy with $\delta = 30°$ and $r_p = 0.5$ cm.
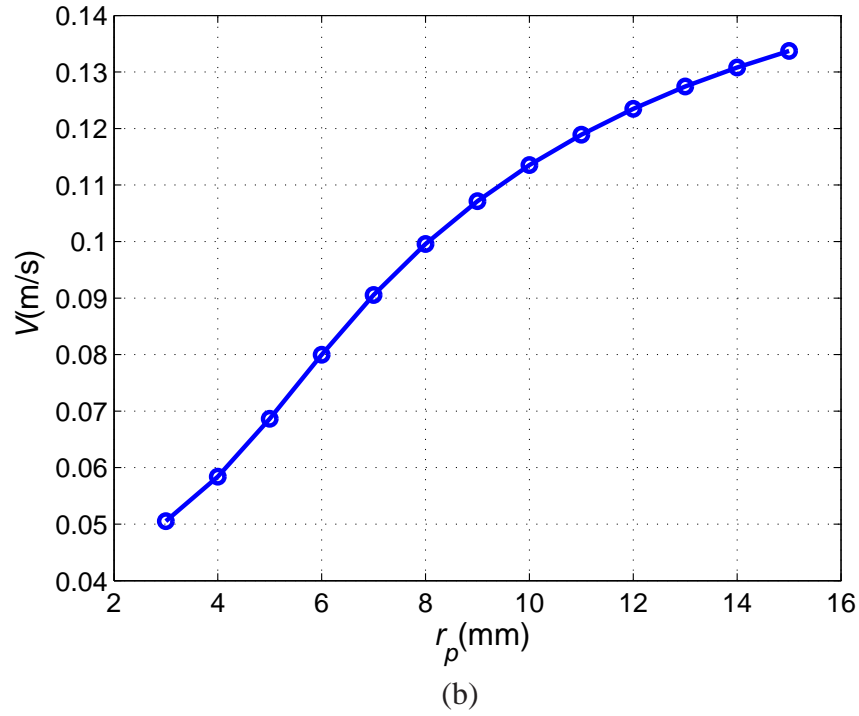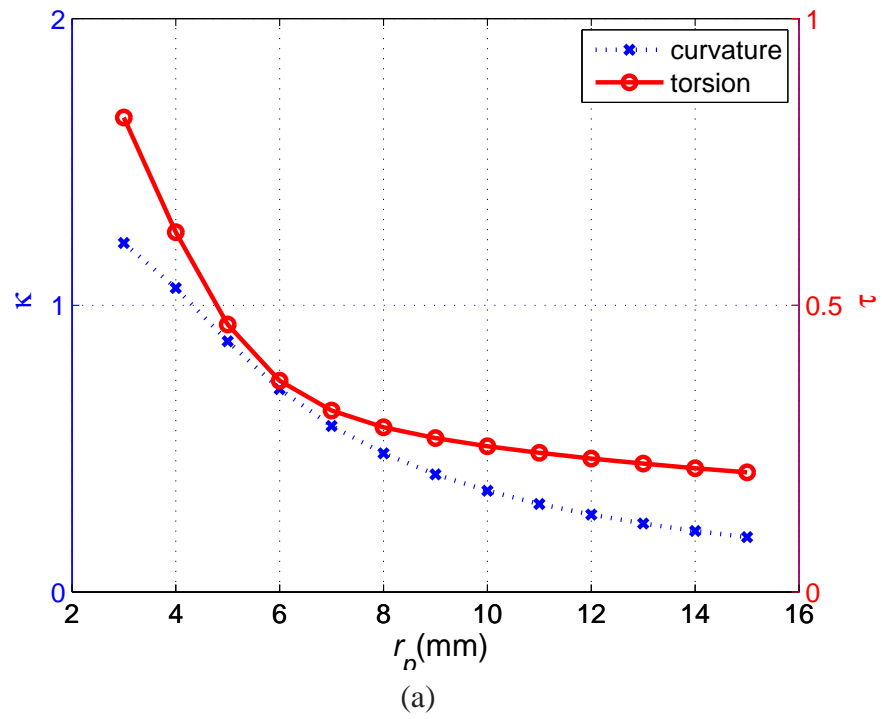
Figure 8.3: Trajectory characteristic parameters at different displacements of movable mass with $\delta = 30°$ and $m_0 = 30$ g.
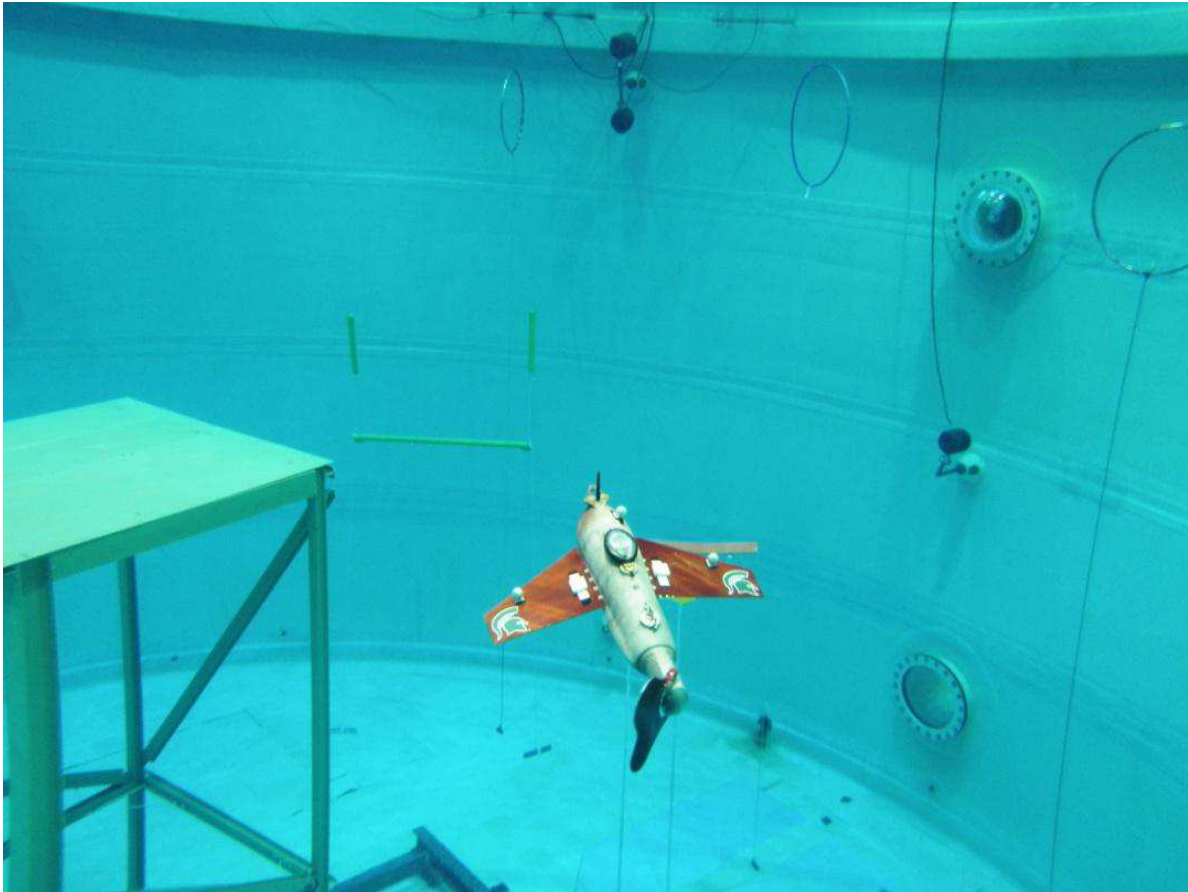
Figure 8.4: The gliding robotic fish "Grace" spiraling in Neutral Buoyancy Research Facility, University of Maryland.

Figure 8.5: Snapshots of spiral motion with 12 underwater cameras from different angles of view using a Qualysis underwater motion capture system.
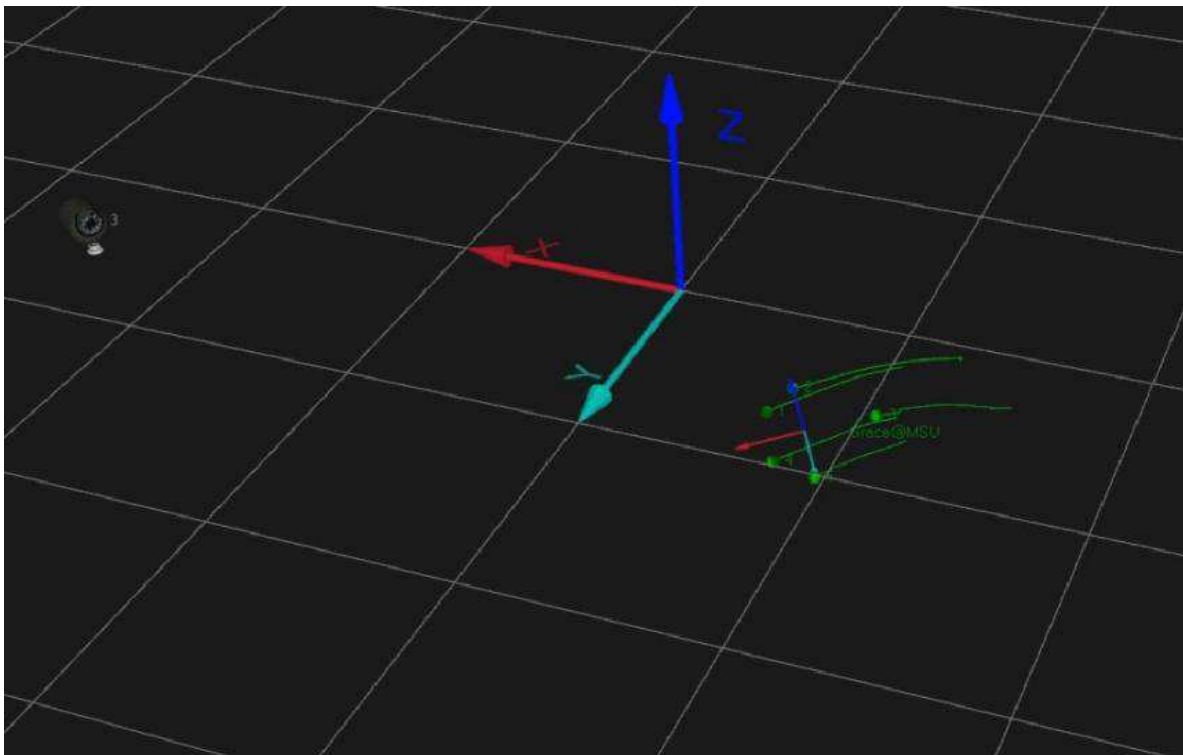
Figure 8.6: Illustration of robot's rigid body and coordinates in spiral motion using a Qualysis underwater motion capture system.

The spiral experiments at each set of control input values are conducted five times. The mean and standard deviation of κ and τ are provided with the error bar in the figures. Considering the current disturbance in the water tank due to the boundary effects and constantly active filtering system, the match between the model prediction and the experimental results is reasonable well, which further validates the derived system model.

The experiments for varying m_0 and r_p are not carried out and presented here because the speed and the gliding angle could be much increased so that the robot will bump into the metal frame located in the center of the water tank (setup for other experiments and not removable currently).

8.3   Two Degree-of-Freedom Control Design

In this section, we propose a 2-DOF control strategy for the curve tracking problem where an inverse mapping is used as a feedforward controller while a robust H_∞ controller is used as the feedback controller, which is designed based on the linearized model around the steady spiral trajectory. The open-loop feedforward controller obtained from inverse mapping of steady spiral motion serves as a driving force pushing the robot towards the desired steady spiral. However, the convergence time is relatively long, which will be demonstrated in simulation later. The feedback H_∞ controller aims to speed up the convergence and enhance the performance robustness. The idea of the 2-DOF controller is that with the feedforward inverse mapping, the dynamic nonlinearity is reduced so that a feedback H_∞ controller can be designed based on the linearized model to achieve improved transient performance.
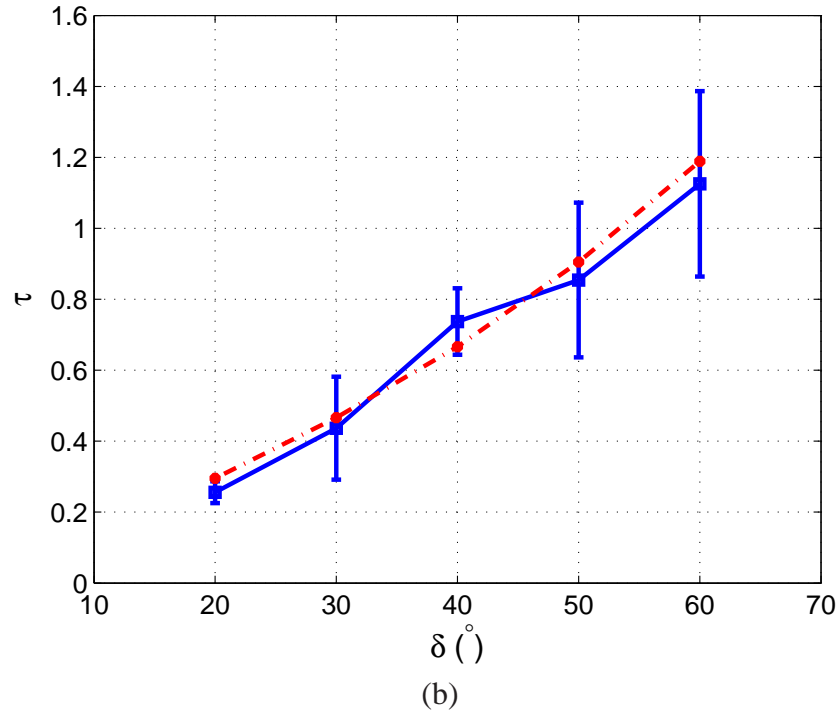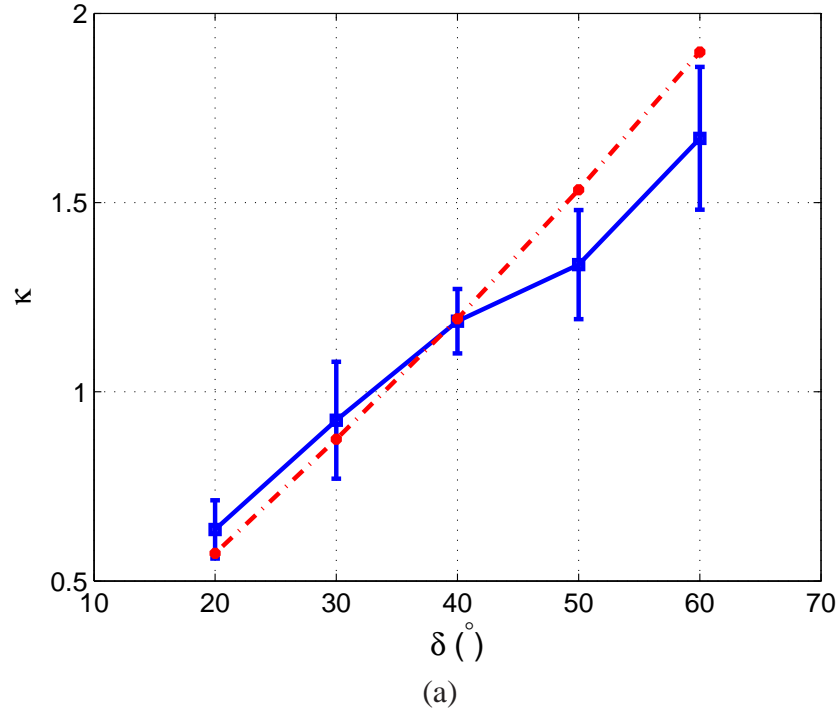
Figure 8.7: Model prediction and experimental results in spiral motion at different tail angles. The other two control inputs are fixed $m_0 = 30$ g, and $r_p = 0.5$ cm. (a) curvature; (b) torsion.

8.3.1 Feedforward Control via Inverse Mapping of Steady Spiral Motion

In this subsection, we study the feedforward control for 3D curve tracking for gliding robotic fish. Based on the fact that a three-dimensional curve can be decomposed into a set of continuously evolving spirals, we propose a 3D curve tracking method by tracking geometric characteristics of these spirals instead of following Euclidean positions. An open-loop feedforward controller is designed using inverse mapping of the steady spiral motion. This inverse controller will also become part of the proposed 2-DOF controller.

We will calculate the desired control inputs for a given steady spiral profile, which is parameterized by curvature, torsion and velocity. This inverse mapping solution can be adopted as an open-loop feedforward controller for the 3D curve tracking problem.

With (8.4), it can be shown that the first two equations in (8.2) always hold, thus redundant. With the value of V_v known from (8.12), we have one more constraint equation

$$V_v = \boldsymbol{R}_{bv} \begin{pmatrix} V \\ 0 \\ 0 \end{pmatrix} (\boldsymbol{R}^T \boldsymbol{k}) \tag{8.13}$$

Given κ, τ and V, we can calculate the value of the angular velocity ω_{3i} from (8.11). Knowing the values of V and ω_{3i}, there are seven unknown variables left out of nine independent states for the steady spiral motion $(\ \phi \quad \theta \quad \omega_{3i} \quad r_p \quad m_0 \quad \delta \quad V \quad \alpha \quad \beta \ )^T$. Correspondingly, there are seven independent algebraic equations from (8.2) and (8.12). The inverse mapping problem is then formulated as

$$0 = \boldsymbol{g}(\boldsymbol{x}) = [g_i(\boldsymbol{x})]_{7 \times 1} \tag{8.14}$$

where $\boldsymbol{x} = (\ \phi \quad \theta \quad \alpha \quad \beta \quad r_p \quad m_0 \quad \delta \ )^T$. The expansion of $g_i(\boldsymbol{x})$ is shown in (8.15)-(8.21).

$$0 = m_2 s\beta V c\phi c\theta \omega_{3i} - m_3 s\alpha c\beta V s\phi c\theta \omega_{3i} - m_0 g s\theta - 1/2\rho V^2 S(C_{D0} + C_D^\alpha \alpha^2 + C_D^\delta \delta^2) c\alpha c\beta$$

$$- 1/2\rho V^2 S(C_{SF}^\beta \beta + C_{SF}^\delta \delta) c\alpha s\beta + 1/2\rho V^2 S(C_{L0} + C_L^\alpha \alpha) s\alpha \tag{8.15}$$

$$0 = -m_3 s\alpha c\beta V s\theta \omega_{3i} - m_1 c\alpha c\beta V c\phi c\theta \omega_{3i} + m_0 g s\phi c\theta - 1/2\rho V^2 S(C_{D0} + C_D^\alpha \alpha^2 + C_D^\delta \delta^2) s\beta$$

$$+ 1/2\rho V^2 S(C_{SF}^\beta \beta + C_{SF}^\delta \delta) c\beta \tag{8.16}$$

$$0 = m_1 c\alpha c\beta V s\phi c\theta \omega_{3i} + m_2 s\beta V s\theta \omega_{3i} + m_0 g c\phi c\theta - 1/2\rho V^2 S(C_{D0} + C_D^\alpha \alpha^2 + C_D^\delta \delta^2) s\alpha c\beta$$

$$- 1/2\rho V^2 S(C_{SF}^\beta \beta + C_{SF}^\delta \delta) s\alpha s\beta - 1/2\rho V^2 S(C_{L0} + C_L^\alpha \alpha) c\alpha \tag{8.17}$$

$$0 = (J_2 - J_3) s\phi c\theta c\phi c\theta \omega_{3i}^2 + (m_2 - m_3) s\beta s\alpha c\beta V^2 + 1/2\rho V^2 S(C_{M_R}^\beta \beta - K_{q1} s\theta \omega_{3i}) c\alpha c\beta$$

$$- 1/2\rho V^2 S(C_{M_0} + C_{M_P}^\alpha \alpha + K_{q2} s\phi c\theta \omega_{3i}) c\alpha s\beta - m_w g r_w s\phi c\theta$$

$$- 1/2\rho V^2 S(C_{M_Y}^\beta \beta + K_{q3} c\phi c\theta \omega_{3i} + C_{M_Y}^\delta \delta) s\alpha \tag{8.18}$$

$$0 = (J_1 - J_3) s\theta c\phi c\theta \omega_{3i}^2 + (m_3 - m_1) c\alpha c\beta s\alpha c\beta V^2 - m_w g r_w s\theta - \bar{m} g r_p c\phi c\theta$$

$$+ 1/2\rho V^2 S(C_{M_R}^\beta \beta - K_{q1} s\theta \omega_{3i}) s\beta + 1/2\rho V^2 S(C_{M_0} + C_{M_P}^\alpha \alpha + K_{q2} s\phi c\theta \omega_{3i}) c\beta \tag{8.19}$$

$$0 = (J_2 - J_1) s\theta s\phi c\theta \omega_{3i}^2 + (m_1 - m_2) c\alpha c\beta s\beta V^2 + 1/2\rho V^2 S(C_{M_R}^\beta \beta - K_{q1} s\theta \omega_{3i}) s\alpha c\beta$$

$$- 1/2\rho V^2 S(C_{M_0} + C_{M_P}^\alpha \alpha + K_{q2} s\phi c\theta \omega_{3i}) s\alpha s\beta + \bar{m} g r_p s\phi c\theta$$

$$+ 1/2\rho V^2 S(C_{M_Y}^\beta \beta + K_{q3} c\phi c\theta \omega_{3i} + C_{M_Y}^\delta \delta) c\alpha \tag{8.20}$$

$$0 = V_v/V + c\alpha c\beta s\theta - s\beta c\theta s\phi - s\alpha c\beta c\theta c\phi \tag{8.21}$$

Unfortunately, there is no closed-form solution to this system of equations. In this dissertation, we use a Newton's method to find solutions recursively, which provide the desired open-loop

control inputs. The iterative algorithm for Newton's method reads [95]

$$\hat{x}_{i+1} = \hat{x}_i - J^{-1}(\hat{x}_i)g(\hat{x}_i) \qquad (8.37)$$

Here $\hat{x}_i$ is the ith-step iteration for x, and $J(x)$ is the Jacobian matrix of $g(x)$

$$J(x,u) = \frac{\partial g}{\partial x} = \left(\frac{\partial g_i}{\partial x_j} \right)_{7 \times 7} \qquad (8.38)$$

8.3.2 Linearized Model Around the Steady Spiral Trajectory

The linearized model can be obtained by linearization of the spiral dynamics around the equilibrium spiral trajectory. Recall the spiral dynamics (8.1)

$$\dot{x}_s = f(x_s, u) = [f_i(x_s, u)]_{8 \times 1}$$

where $x_s = \begin{pmatrix} \phi & \theta & v_1 & v_2 & v_3 & \omega_1 & \omega_2 & \omega_3 \end{pmatrix}^T$, and $u = \begin{pmatrix} r_p & m_0 & \delta \end{pmatrix}^T$. We define transformed system states $z = \begin{pmatrix} \phi & \theta & V & \alpha & \beta & \omega_1 & \omega_2 & \omega_3 \end{pmatrix}^T$ for the convenience of computation of the Jacobian matrix $J(x_s, u)$. The linearized system matrices are

$$A = [J(x_s, u)]_e = \left[\frac{\partial f}{\partial x} \right]_e = \left[\frac{\partial f}{\partial z} \left(\frac{\partial x}{\partial z} \right)^{-1} \right]_e \qquad (8.39)$$

$$B = \left[\frac{\partial f}{\partial u} \right]_e \qquad (8.40)$$

Here $[\cdot]_e$ means the matrix elements are evaluated at the equilibrium point.

We define the linearized system output as $y = \begin{pmatrix} \phi & \theta & v_1 \end{pmatrix}^T$, linear in system states. The

linearized system output matrices are

$$C = [\mathbf{1}_{3\times3} \ \mathbf{0}_{3\times5}] \tag{8.41}$$

$$D = \mathbf{0}_{3\times3} \tag{8.42}$$

The objective of the controller design for the linearized model is to drive the three selected system outputs to the desired values that are computed via inverse mapping (Section 8.3.1) at an improved speed.

8.3.3 H_∞ Controller Design

A 2-DOF control is adopted for the 3D curve tracking problem to increase the system bandwidth. The 2-DOF control system configuration is shown in Fig. 8.8. The transfer function $G(s)$ represents the spiral dynamics system. $K_{fb}(s)$ is the feedback controller. $\boldsymbol{r} = (\phi_r \ \theta_r \ v_{1r})^T$ is the perturbation of the reference signal from the nominal values. $\boldsymbol{u} = (r_p \ m_0 \ \delta)^T$ is the control input to the plant. $\boldsymbol{e} = \boldsymbol{r} - \boldsymbol{y}$ represents the tracking error. $W_e(s)$ is the user-defined weighting function to impose the requirements for the tracking bandwidth and tracking error amplitude. The state-space realization of $W_e(s)$ is as follows,

$$\dot{\boldsymbol{x}}_{\boldsymbol{w}} = A_w \boldsymbol{x}_{\boldsymbol{w}} + B_w \boldsymbol{u}_{\boldsymbol{w}} \tag{8.43}$$

$$z_w = C_w \boldsymbol{x}_{\boldsymbol{w}} + D_w \boldsymbol{u}_{\boldsymbol{w}} \tag{8.44}$$

$W_u = \text{diag}(w_{u1}, w_{u2}, w_{u3})$ is the weighting function to help control the magnitude of system control inputs, and $\boldsymbol{z_u} = W_u \boldsymbol{u}$.

The tracking performance can be characterized by the tracking error $\boldsymbol{e}$. Meanwhile, the control
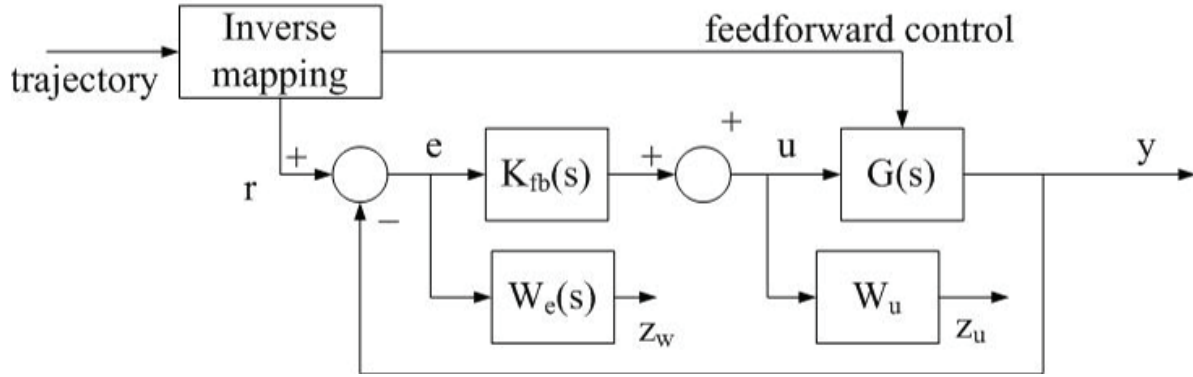
Figure 8.8: The control system diagram with combination of open-loop control and closed-loop control.
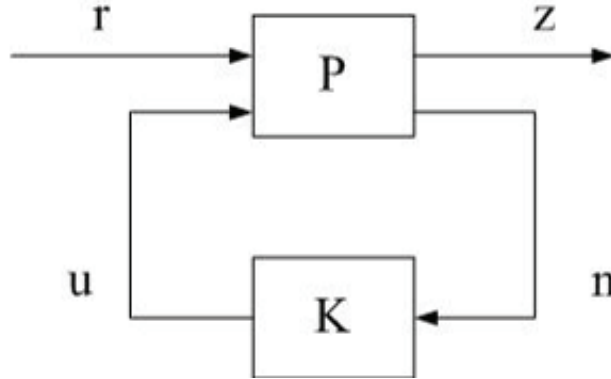
Figure 8.9: Transformed 2-DOF control configuration in H_∞ control framework .

effort can be characterized by the control input $\boldsymbol{u}$ which is desired to be small for the consideration of energy consumption. The objective of the feedback control design is to minimize those two signals $\boldsymbol{e}$ and $\boldsymbol{u}$.

This optimization problem with the feedback control for the linearized model can be transformed into an H_∞ robust control framework as shown in Fig. 8.9. In this H_∞ control system

configuration,

$$K(s) = K_{fb}(s) \tag{8.45}$$

$$\boldsymbol{z} = \begin{pmatrix} \boldsymbol{z_w} & \boldsymbol{z_u} \end{pmatrix}^T \tag{8.46}$$

$$\boldsymbol{n} = \boldsymbol{r} - \boldsymbol{y} \tag{8.47}$$

The interconnected system

$$P = \begin{pmatrix} A_p & B_p \\ C_p & D_p \end{pmatrix} \tag{8.48}$$

where

$$A_p = \begin{pmatrix} A & \mathbf{0} \\ -B_w C & A_w \end{pmatrix} \qquad B_p = \begin{pmatrix} \mathbf{0} & B \\ B_w & -B_w D \end{pmatrix}$$

$$C_p = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ -D_w C & C_w \\ -C & \mathbf{0} \end{pmatrix} \qquad D_p = \begin{pmatrix} 0 & W_u \\ D_w & -D_w D \\ \mathbf{1} & -D \end{pmatrix}$$

The design objective is then to minimizing the H_∞ norm of the transfer function from $\boldsymbol{r}$ to $\boldsymbol{z}$

$$\min_{K} \|T_r^z(s)\|_\infty \tag{8.49}$$

To solve the above H_∞ optimization problem, we adopt the command *hinfmix*(·) in Matlab LMI toolbox, the output of which provides the feedback controller $K(s)$.

150

8.4 Simulation Results

Given desired trajectories of curvature, torsion and velocity, simulation is conducted to test the effectiveness of the proposed 2-DOF control algorithm. For the purpose of comparison, we have also conducted simulation with pure inverse mapping control (see Section 8.3.1) and Proportional-Integral (PI) control. The PI controller is designed as

$$\delta \;=\; K_P^{\delta}\Delta\kappa + K_I^{\delta}\int\Delta\kappa \tag{8.50}$$

$$r_p \;=\; K_P^{rp}\Delta\tau + K_I^{rp}\int\Delta\tau \tag{8.51}$$

$$m_0 \;=\; K_P^{m0}\Delta V + K_I^{m0}\int\Delta V \tag{8.52}$$

where Δ stands for the difference between the desired value and actual value of the variable that follows. The particular form of the PI controller, where one control input is only dependent on the error feedback from one geometric parameter, is adopted for design convenience and based on the observed influences (in simulation) of the control inputs on the geometric parameters, where it appears that each control input has more pronounced impact on one of the geometric parameters than other two inputs. The PI controller coefficients are designed as $K_P^{\delta} = 0.1$, $K_I^{\delta} = 0.01$, $K_P^{rp} = 0.05$, $K_I^{rp} = 0.005$, $K_P^{m0} = 0.1$, $K_I^{m0} = 0.01$. There are three control inputs and three geometric variables to track, so the strong coupling between the control inputs makes the parameter tuning quite challenging. The PI control parameters are tuned in simulation in order to obtain the best tracking performance.

The model parameters used in simulation are based on the lab-developed gliding robotic fish

prototype "Grace" as in Table 7.1. The initial values of system states used in simulation are

$$\theta = -7.2° \qquad \phi = 0 \qquad v_1 = 0.1 \text{ m/s} \qquad v_2 = 0$$

$$v_3 = 0.04° \qquad \omega_1 = 0 \qquad \omega_2 = 0 \qquad \omega_3 = 0$$

This represents a sagittal-plane glide motion.

The weighting functions for the feedback H_∞ control design are chosen as

$$A_w = \text{diag}(-500, -100, -200) \qquad B_w = \text{diag}(9, 9, 9)$$

$$C_w = \text{diag}(-1, -1, -1) \qquad D_w = \text{diag}(0.1, 0.1, 0.1)$$

$$W_u = \text{diag}(0.2, 0.05, 0.2)$$

The solution to the H_∞ optimization problem is a eleventh-order linear system for the controller $K(s)$. Through model reduction techniques by investigating the dominant singular values, a seventh-order controller is used in simulation. Saturation is also imposed to restrict the control inputs to the feasible range of actuators.

Figures 8.10 and 8.11 show the simulation results of the tracking performance of three geometric parameters and the control efforts of three control inputs, respectively, for tracking a steady spiral trajectory with constant geometric parameters and Fig. 8.12 shows the tracking trajectory in the 3D view under the proposed 2-DOF controller. From the simulation results, we see that both PI controller and feedforward inverse mapping controller are able to stabilize the system to the desired steady-spiral trajectory. However, both have convergence times between 30 and 40 seconds. There is also noticeable steady-state tracking error with the PI controller. With the proposed 2-DOF controller, the system tracking performance is improved significantly. Convergence

time decreases to less than 10 seconds with smaller steady-state errors. Meanwhile, the 2DOF controller uses more control effort than the open-loop feedforward strategy. Figs. 8.13 and 8.14 show the simulation results of the tracking performance of three geometric parameters and the control efforts of three control inputs, respectively, when the reference velocity changes as a sinusoid function with respect to time while the curvature and torsion are kept constant. There is a large time/phase delay in the tracking of velocity for the feedforward control, which is expected due to the observed slow convergence speed. Besides the time delay, there is also significant tracking error with the PI control. But with the 2-DOF controller, the tracking performance shows significant improvement in terms of the time/phase delay and the tracking error. Besides, in Figs. 8.13a and 8.13b, the variables fluctuate even though the reference is a constant. This shows coupling among control inputs on the spiral geometric features.
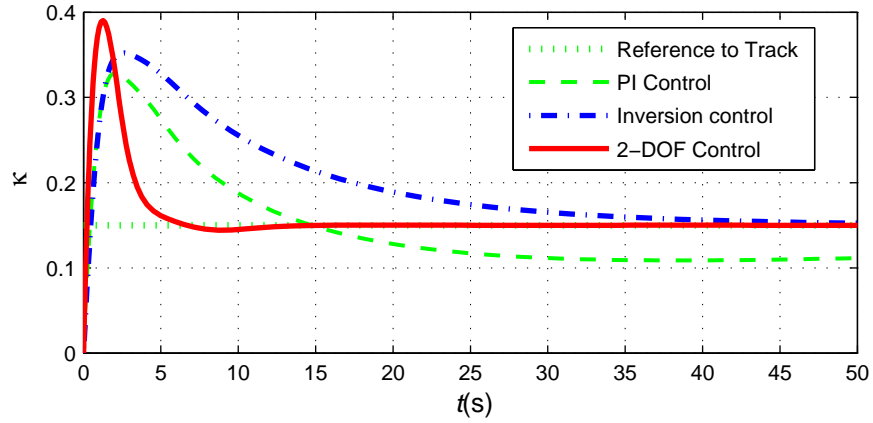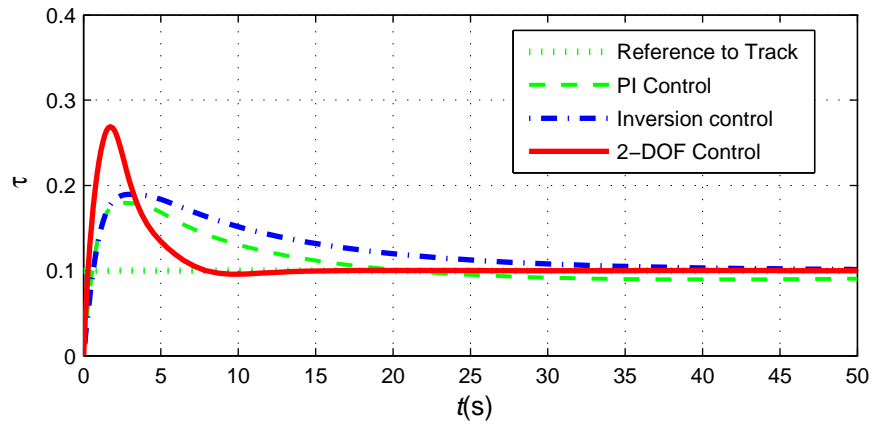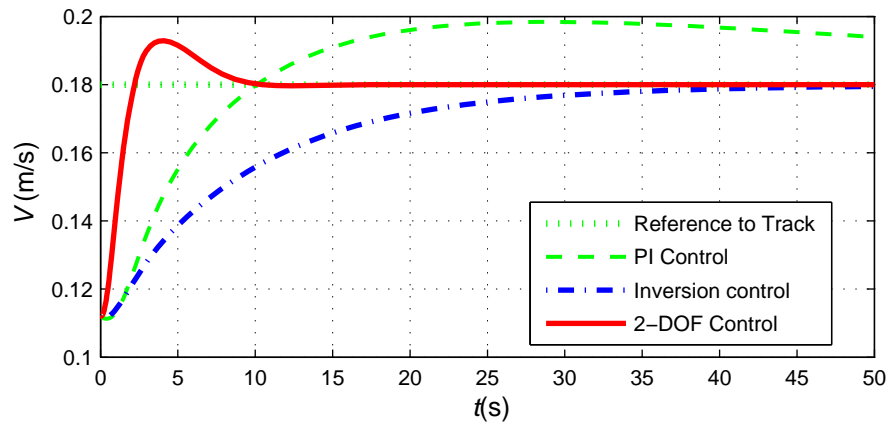
Figure 8.10: The simulation results of the geometric parameters when tracking a steady spiral trajectory. (a) curvature; (b) torsion; (c) velocity.
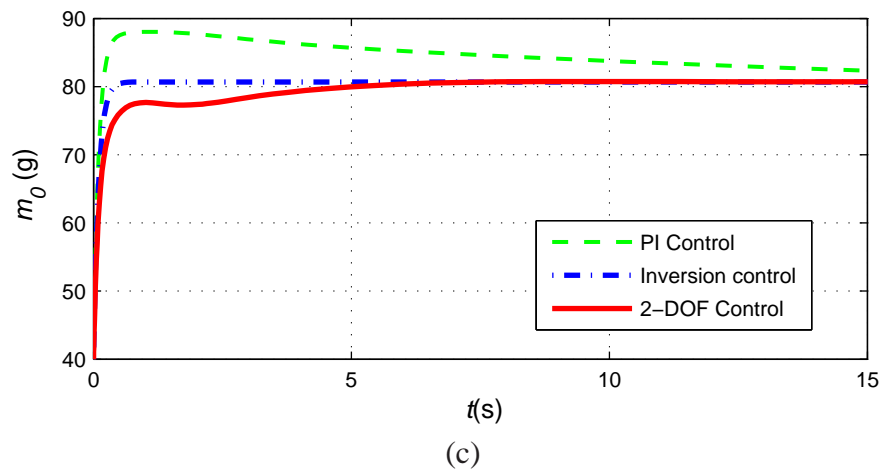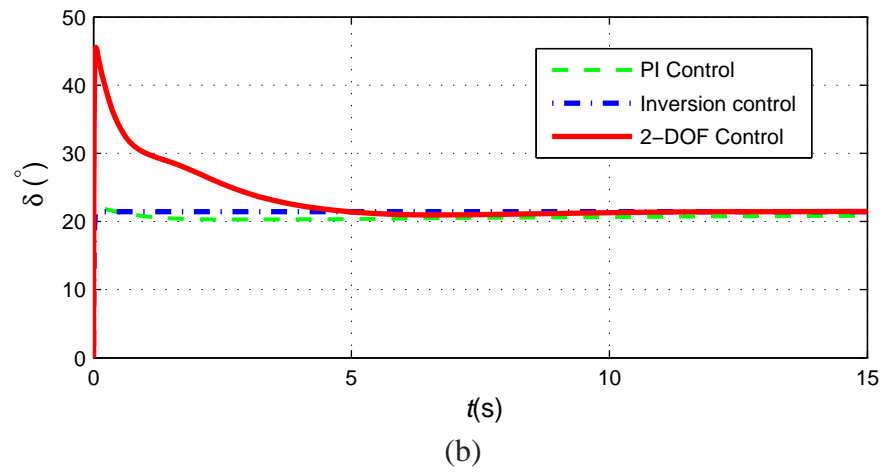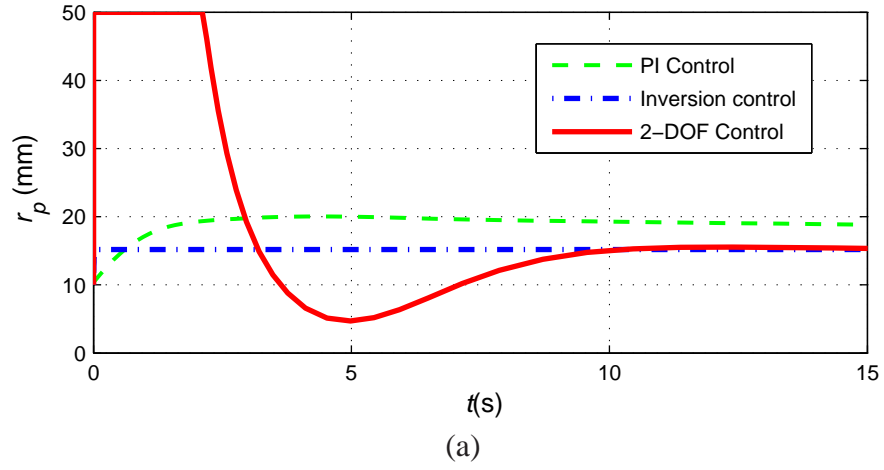
Figure 8.11: The simulation results of control inputs on when tracking a steady spiral trajectory. (a) displacement of movable mass; (b) tail angle; (c) net buoyancy.
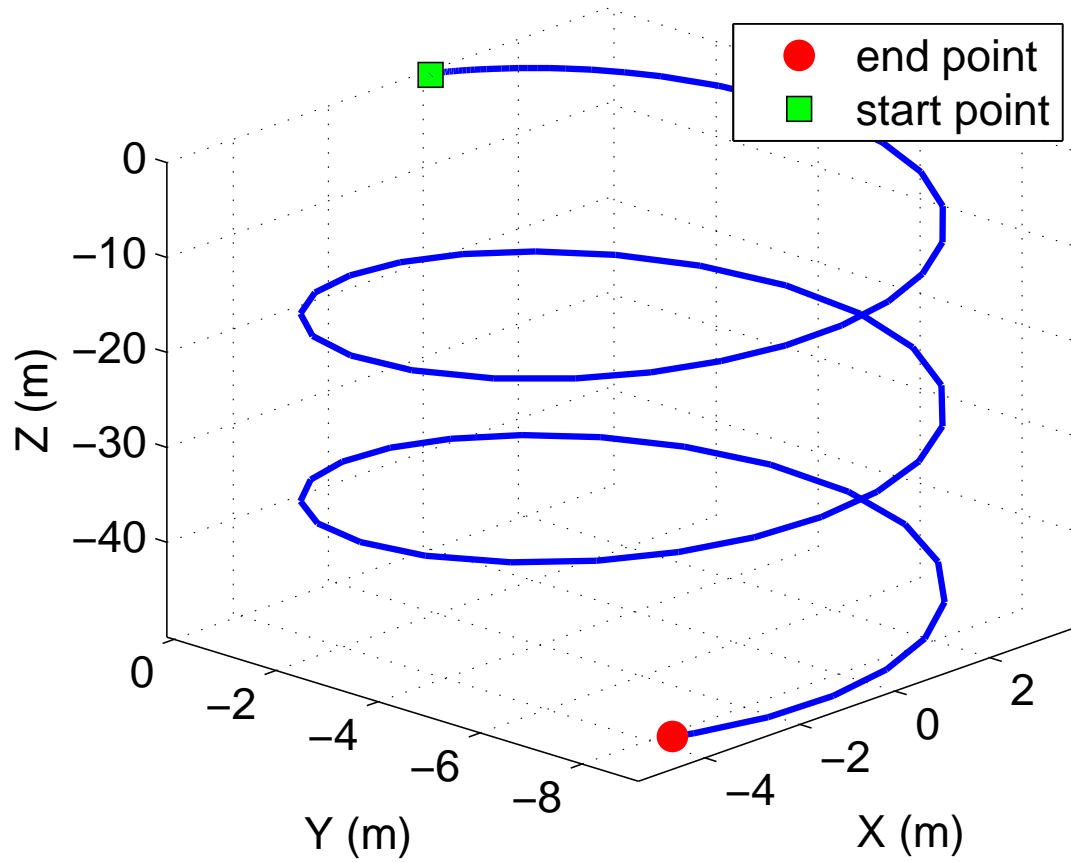
Figure 8.12: 3D trajectory when tracking a steady spiral under the 2-DOF controller.
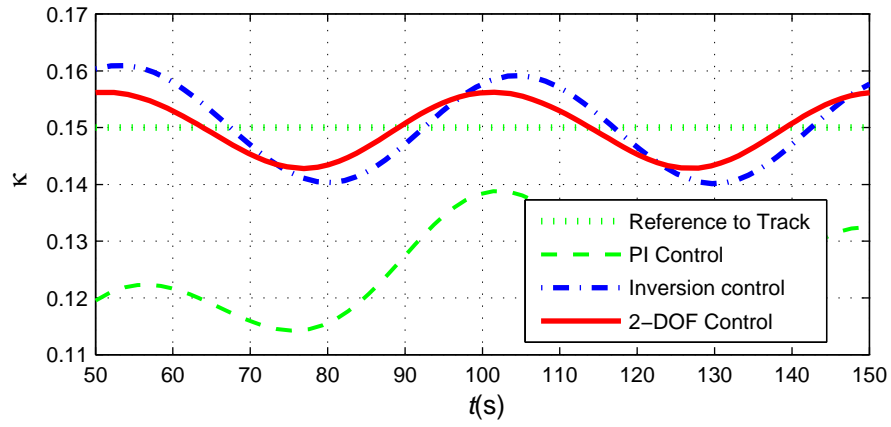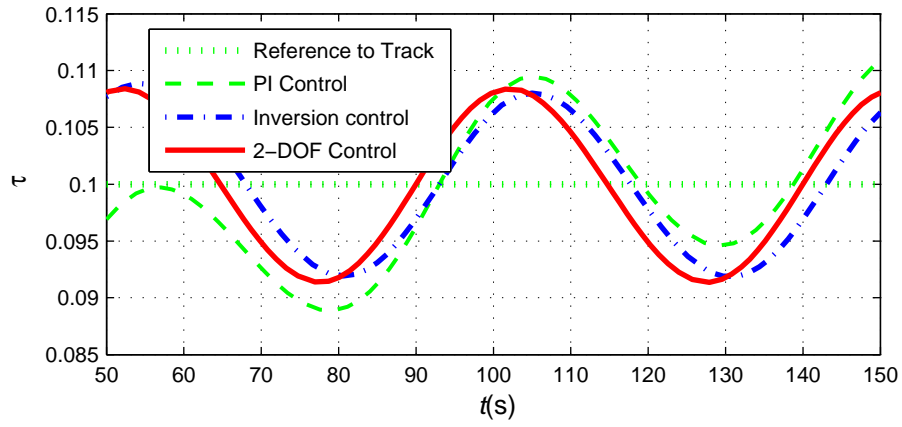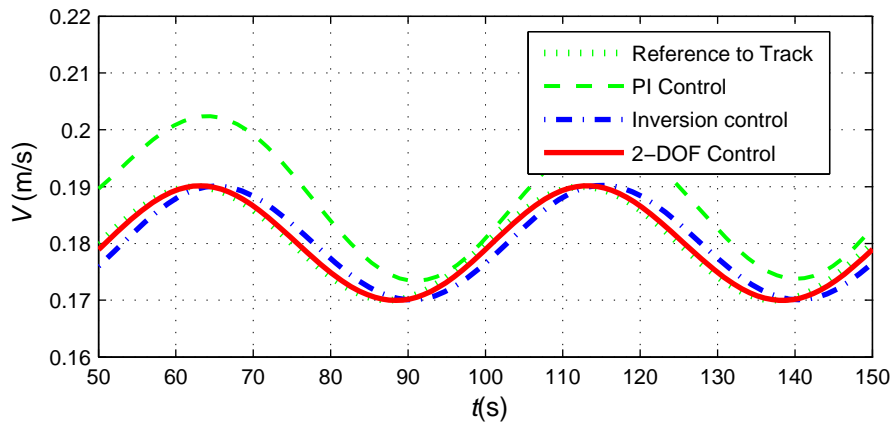
Figure 8.13: The simulation results of the geometric parameters when the reference velocity changes as a sinusoid function with respect to time while curvature and torsion are kept constant. (a) curvature; (b) torsion; (c) velocity.
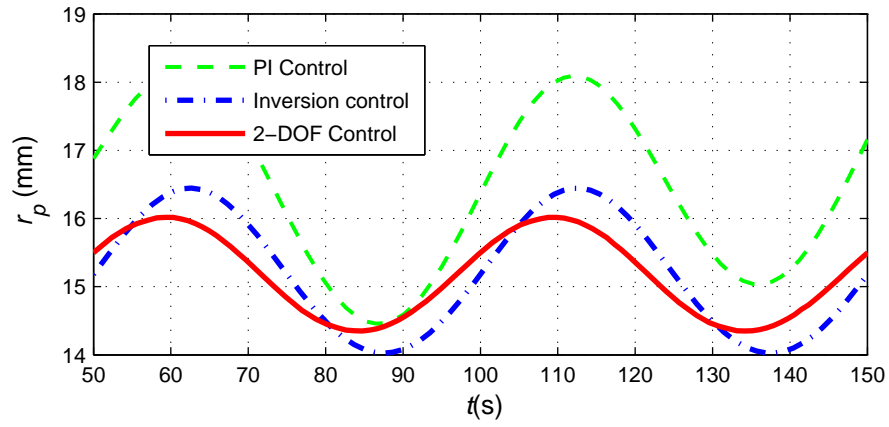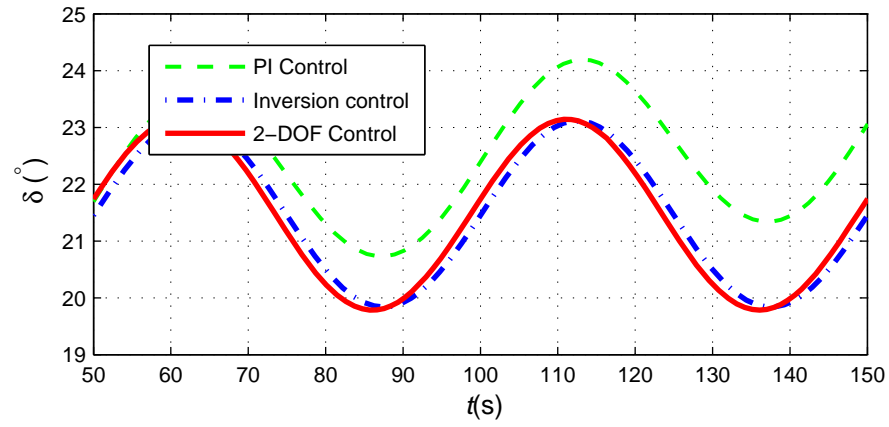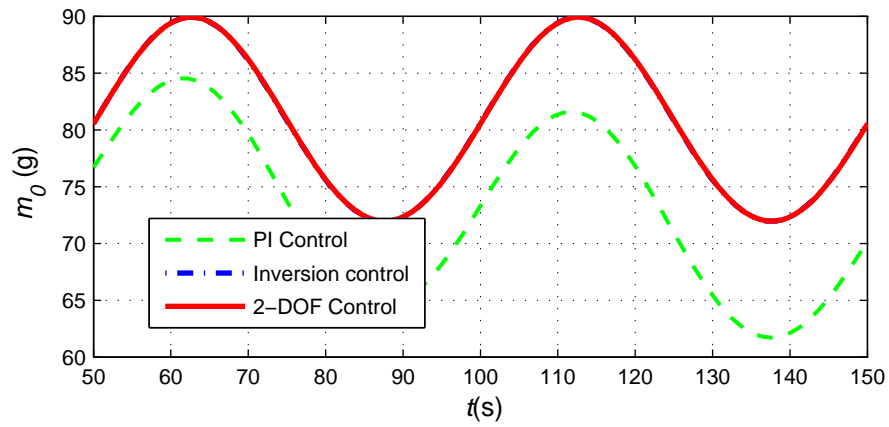
Figure 8.14: The simulation results of control inputs on when the reference velocity changes as a sinusoid function with respect to time while curvature and torsion are kept constant. (a) displacement of movable mass; (b) tail angle; (c) net buoyancy.

Chapter 9

Field Test Results for Environmental Monitoring

In this chapter, we present some preliminary field testing results for oil spill detection in the Kalamazoo River, Michigan, and algae bloom monitoring in the Wintergreen Lake, Michigan, where basic functions of gliding robotic fish, the swimming, the gliding, and the spiraling are tested in the field. It is found that the crude oil sensor readings were different at the three selected locations along Kalamazoo River near the spill site, with slightly higher values at the downstream sites. For the algae concentration, the sampling results were consistent with readings from a traditional hand-held device [84].

9.1 Kalamazoo River Test

"Grace" was deployed in Kalamazoo River in November, 2012, to detect the crude oil content near the 2010 Enbridge oil spill site near Marshall, Michigan. A Turner Designs Cyclops-7 crude oil sensor was used to sample the water. We tested three locations along the Kalamazoo River (Fig. 9.1). The spot A was downstream of the oil spill spot which was an open river area (Fig. 9.2); the spot B was upstream which was in the woods (Figs. 9.3); and the spot C was also upstream which was under a bridge (Fig. 9.4). "Grace" swam in the Kalamazoo river by flapping its tail fin, collecting data in the selected locations. The readings from the sensor are recorded onboard
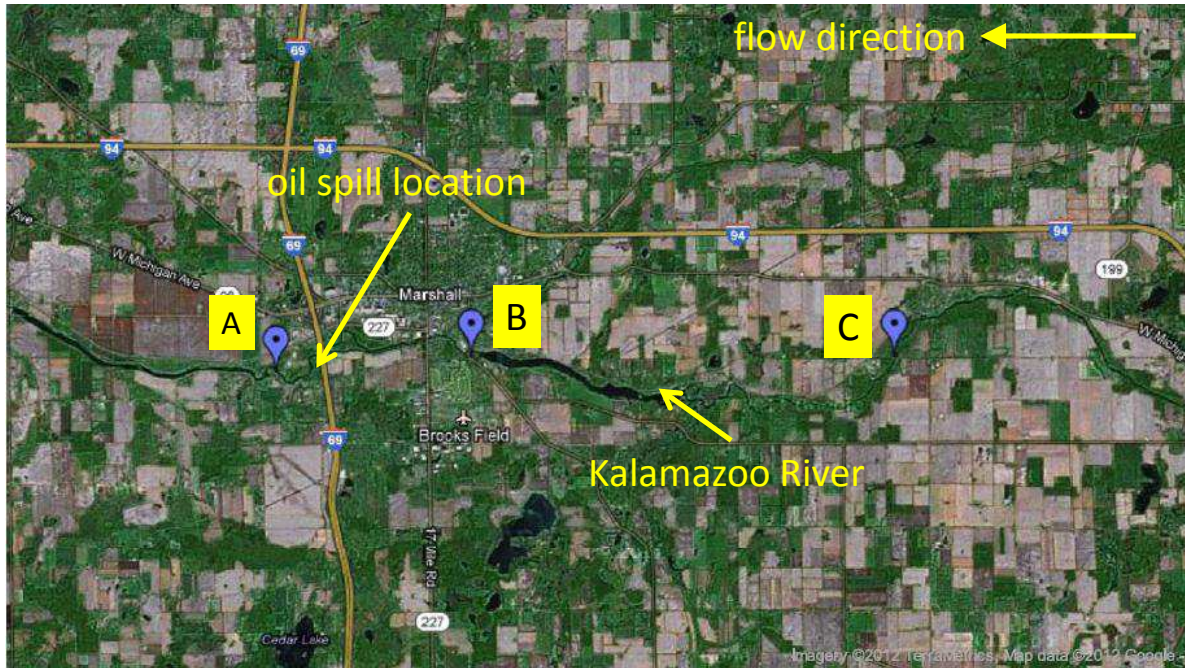
Figure 9.1: Three selected sampling locations illustrated in Google Map.

Table 9.1: Locations and sensor readings in Kalamazoo River test.

Spot	GPS Coordinate	Sensor Reading
A	(42.258655, -84.99888333)	3.33 V
B	(42.26156167, -84.953565)	3.04 V
C	(42.2616, -84.85534667)	2.87 V

and transmitted wirelessly to the base station onshore. The GPS position and the average sensor reading at each spot are shown in Table 9.1.

The results show that the sensor readings were slightly higher at the downstream sites of the spill spot than the upstream. There is a relationship between the sensor's output (in voltage) and the real concentration of crude oil. In this dissertation, we provide only the raw readings from the crude oil sensor. The detection spectrum of the sensor is relatively wide so that it may not precisely reflect the actual crude oil concentration.

Figure 9.2: Swimming trajectory of gliding robotic fish in an open water area at testing spot A.

Figure 9.3: Swimming trajectory of gliding robotic fish in the woods at testing spot B.

Figure 9.4: Swimming trajectory of gliding robotic fish under a bridge at testing spot C.

9.2 Wintergreen Lake Test

Gliding robotic fish "Grace" was also used to sample the Wintergreen Lake, Michigan for the blue-green algae (cyanobacteria) concentration in July, 2013. The sensor we used was Turner Designs Cyclops-7 Freshwater Blue-Green Algae Sensor. The sensor readings in this field test are compared with those from a manually deployed profiler with the blue-green algae sensor (Hydrolab) that has been traditionally used in the sampling of harmful algae.

Surface swimming was tested first at Wintergreen Lake using "Grace" (Fig. 9.5), with sensor readings recorded for selected locations. Table 9.2 shows the GPS position and the sensor outputs of algae concentration at those points.

Since algae concentration varies at different depths, we ran a number of steady glide and steady spiral motions using the robot to detect the algae bloom in the three-dimensional water space. Figs. 9.6–9.9 show the results of steady glide tests. The start (submerging) and the end (surfacing)

162
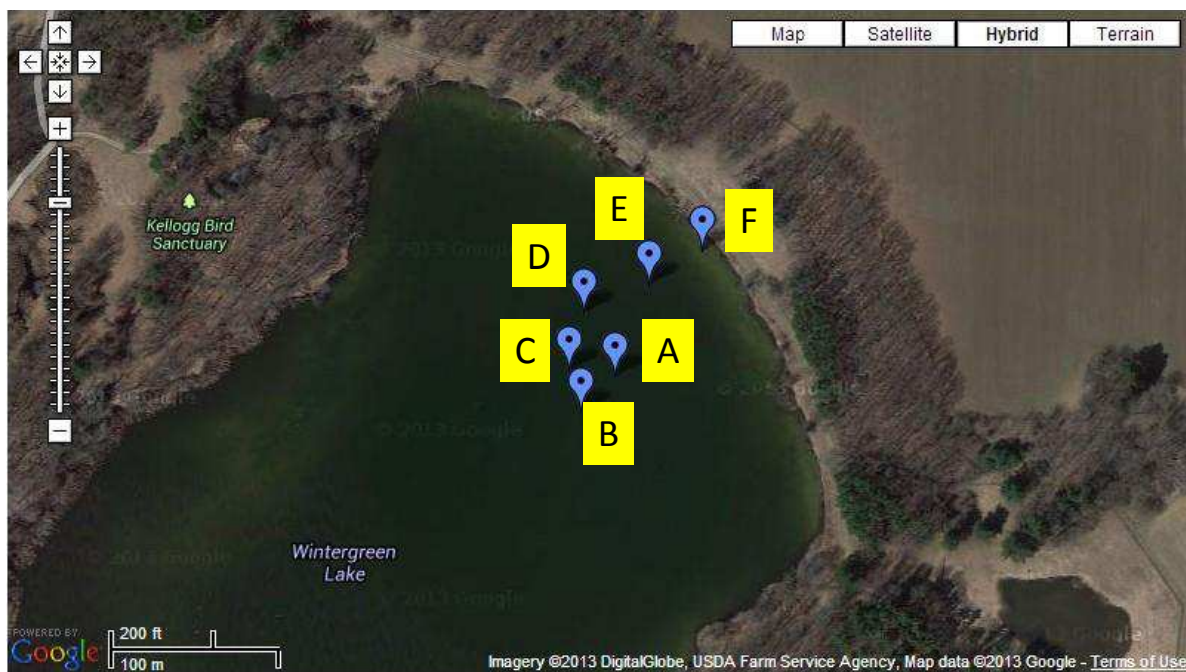
Figure 9.5: Selected sampling points in the Wintergreen Lake with surface swimming.

Table 9.2: Locations and sensor readings in Wintergreen Lake test.

Spot	Sensor Reading	Hydrolab Chl	GPS Coordinate
A	0.252979 V	2.19	(42.3987, -85.38334)
B	0.336768 V	2.45	(42.39851, -85.38359)
C	0.289233 V	2.33	(42.39874, -85.38367)
D	0.263452 V	3.06	(42.39904, -85.38356)
E	0.269897 V	2.60	(42.39919, -85.3831)
F	0.422168 V	3.89	(42.39938, -85.38271)
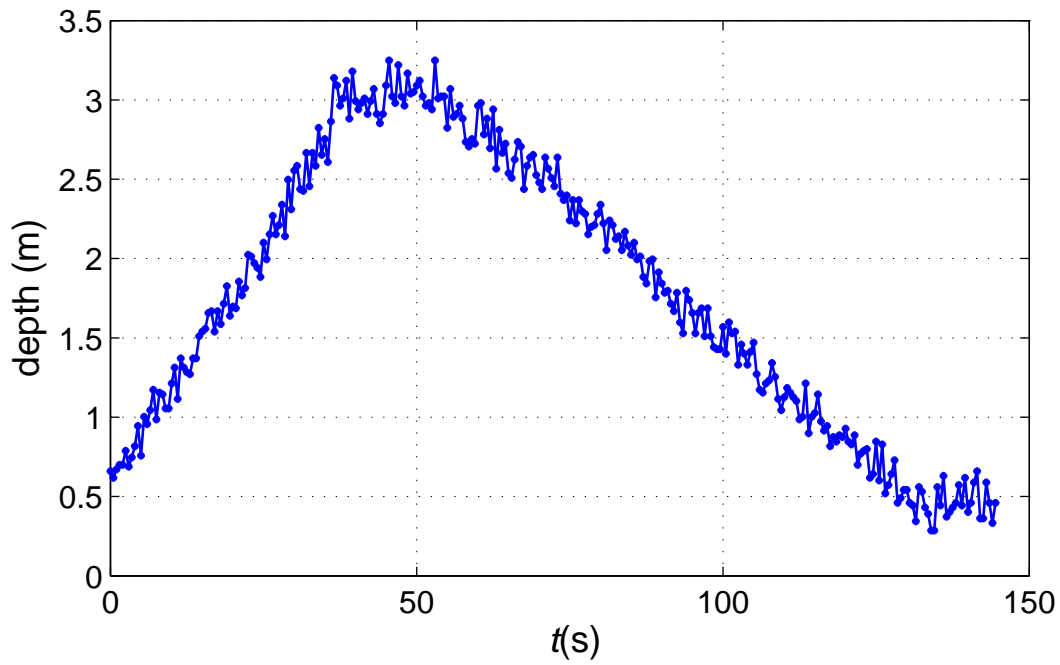
Figure 9.6: Depth trajectory of steady glide when sampling water in Wintergreen Lake, Michigan.

points are (42.398228, -85.384644), and (42.398361, -85.384636), respectively, with a travel dis-

tance of 14 meters. The readings of Chlorophyll from the Hydrolab device at this location are 3.11

at 1.26 m, 5.69 at 2.45 m, and 8.73 at 3.55 m. Figs 9.10–9.13 show the results for the steady spiral

test with the spiral center at (42.398338, -85.384727). The readings of Chlorophyll from the Hy-

drolab device at this location are 2.89 at 0.97 m, 6.59 at 3.48 m, and 10.26 at 4.05 m. The readings

from the robot were found to be consistent in general with those from the Hydrolab device.
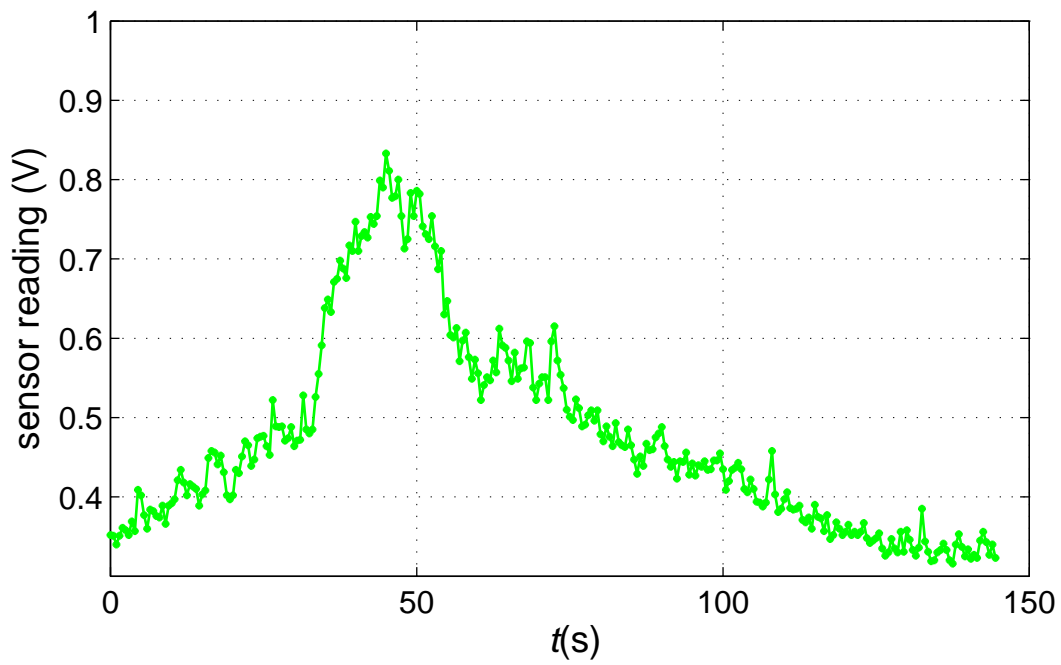
Figure 9.7: Sensor readings of steady glide when sampling water in Wintergreen Lake, Michigan.
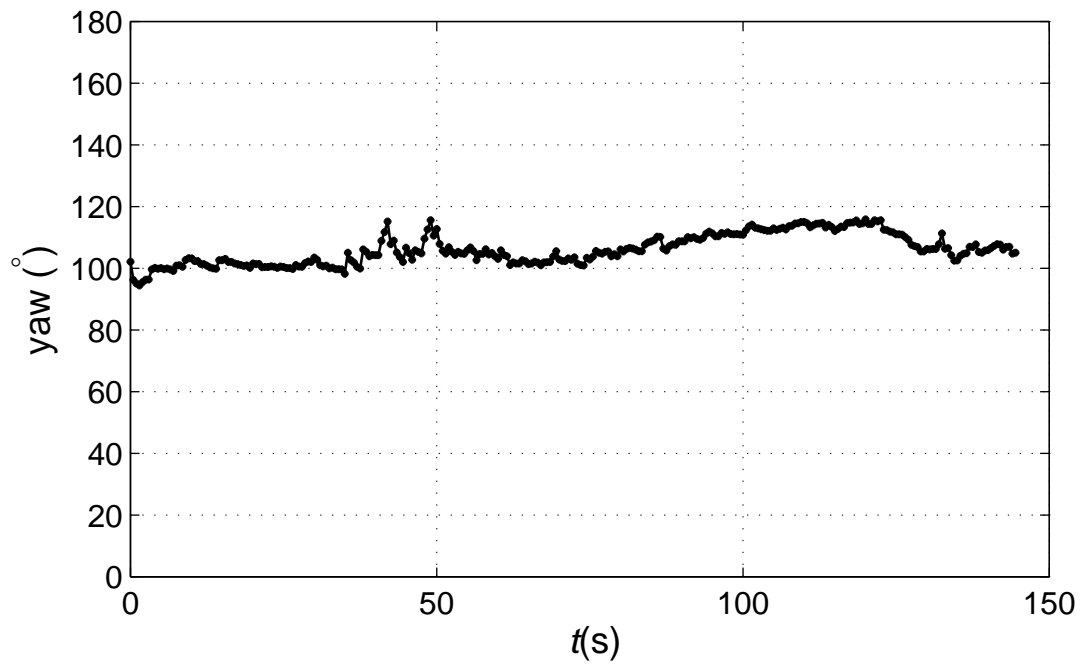
Figure 9.8: Yaw angle trajectory when sampling water in Wintergreen Lake, Michigan.

165
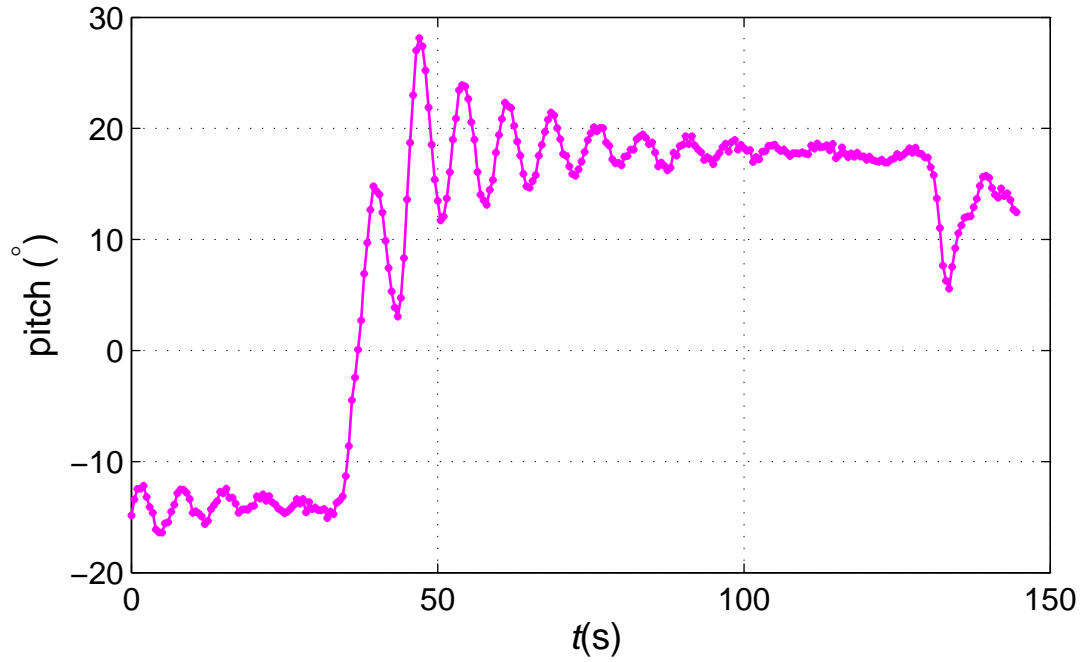
Figure 9.9: Pitch angle trajectory when sampling water in Wintergreen Lake, Michigan.
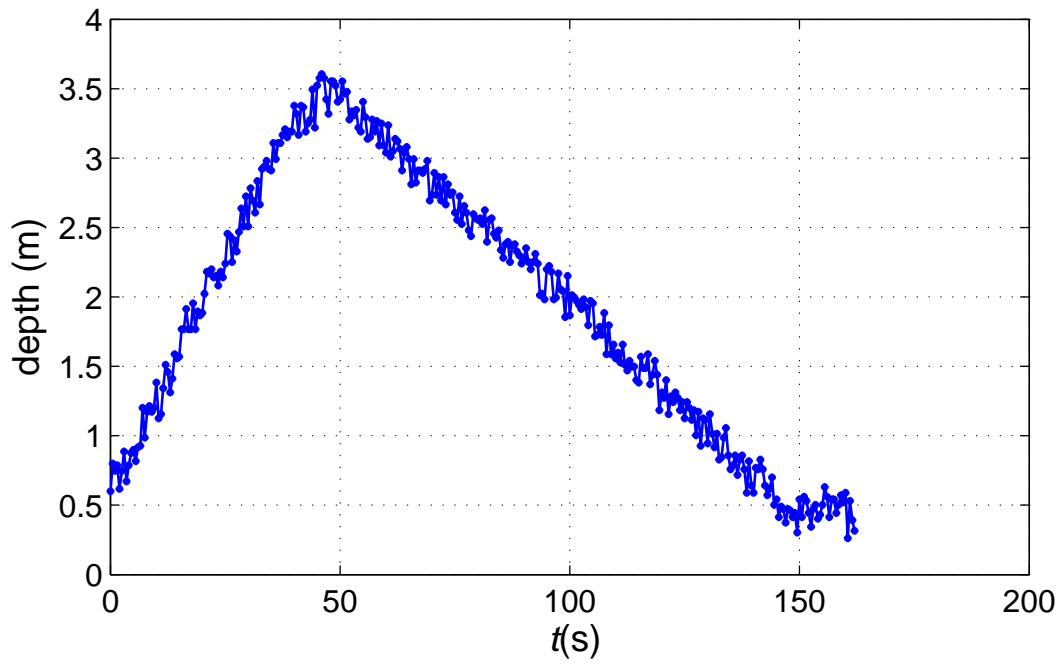
Figure 9.10: Depth trajectory of steady spiral when sampling water in Wintergreen Lake, Michigan.
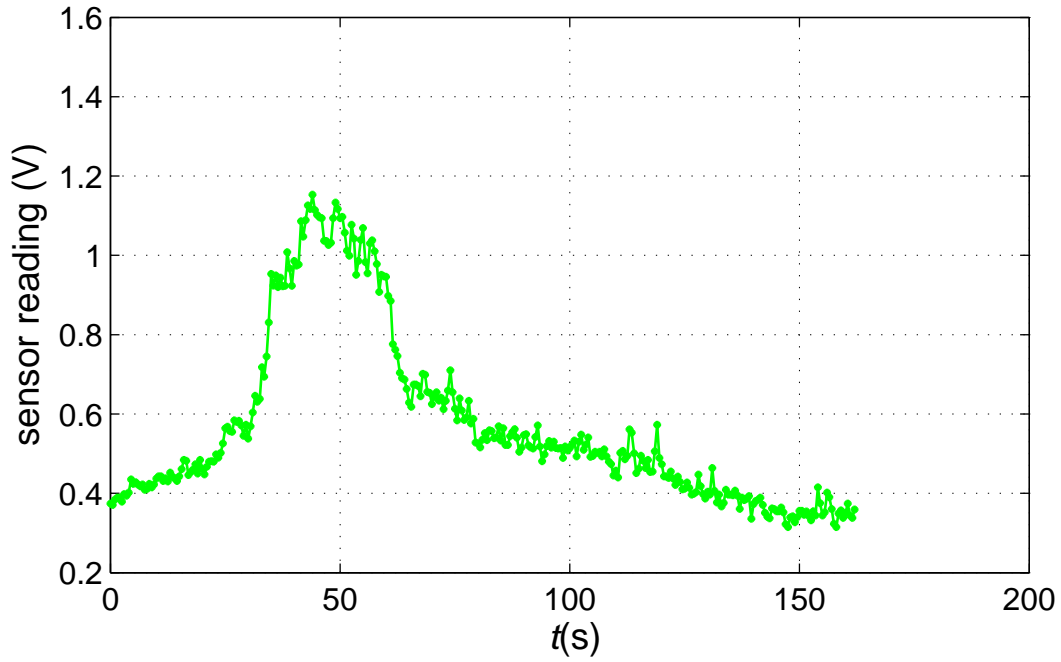
Figure 9.11: Sensor readings of steady spiral when sampling water in Wintergreen Lake, Michigan.
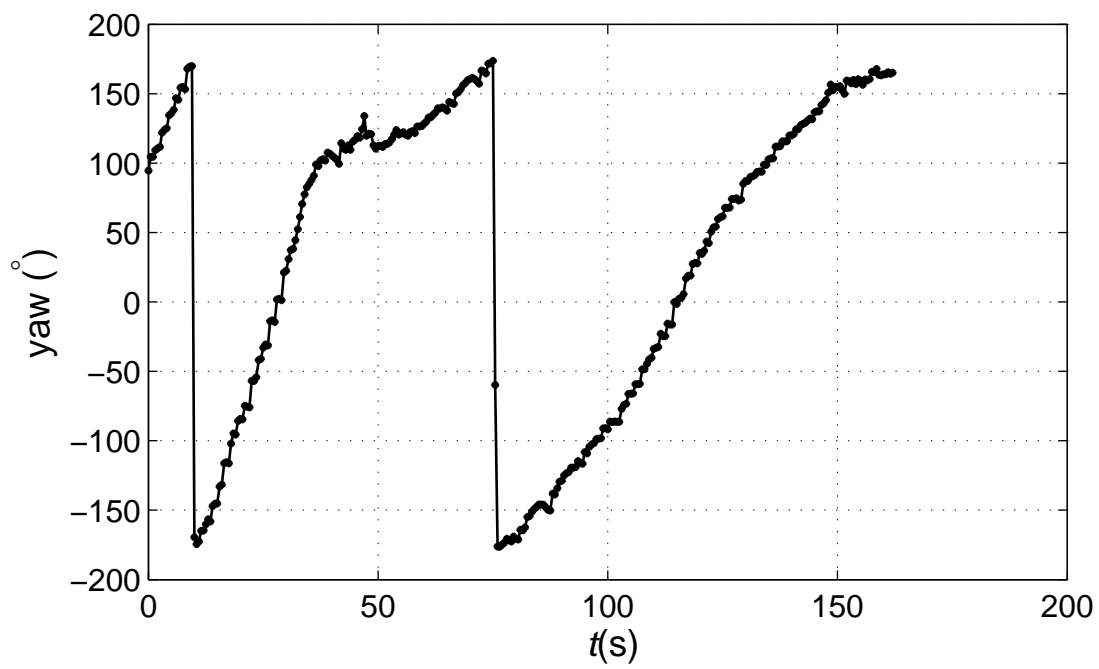
Figure 9.12: Yaw angle trajectory of steady spiral when sampling water in Wintergreen Lake, Michigan.
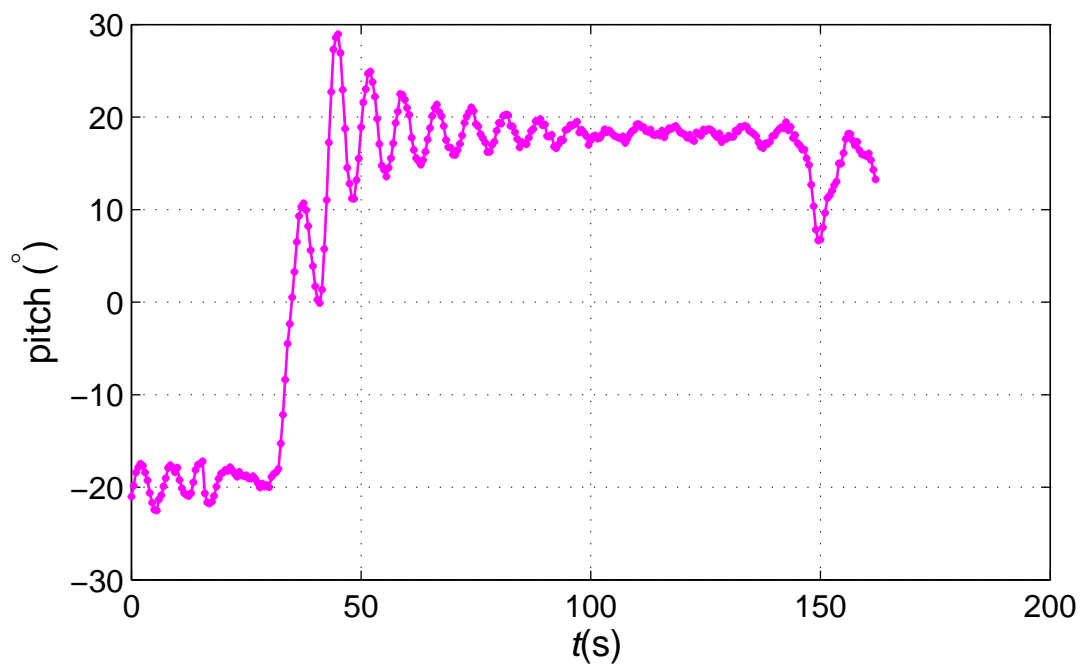
Figure 9.13: Pitch angle trajectory of steady spiral when sampling water in Wintergreen Lake, Michigan.

Chapter 10

Conclusions and Future Work

10.1 Conclusions

We reported a new type of underwater robots, gliding robotic fish, designed for aquatic environmental monitoring. Combing design features of buoyancy-driven propulsion of underwater gliders and tail-fin actuation of robotic fish, the gliding robotic fish shows a great potential in shallow water sampling with energy efficiency and high maneuverability.

We first introduced two lab-developed prototypes, a miniature underwater glider and a gliding robotic fish "Grace". The actuation systems were discussed including the buoyancy system, mass distribution system and tail-fin system, to explain the locomotion mechanism of the robot. The sensor system was introduced, including the inertial measurement units and environmental sensing units. The mechanical design of the robot was also discussed.

Dynamic model of gliding robotic fish was derived and further reduced according to two special steady-state motions, the sagittal-plane glide and 3D steady sprial. Solutions of the robot's gliding path were provided for both motions. Particularly, for steady spiral, we adopted Newton's method and numerically explored the basins of attraction for this recursive algorithm. Experiments were conducted on the miniature underwater glider prototype. Comparison between model prediction and experimental results of the glide path was carried out to validate the derived model and proposed recursive solving method.

Stabilization problems were studied, for both sagittal-plane and lateral motions. For the sagittal-

plane gliding path stabilization, a passivity-based controller using a whale-like tail fin was proposed with partial state feedback. A nonlinear observer was designed to estimate the velocity-related system states in controller implementation. Both simulation and experiments were conducted to evaluate the effectiveness of the designed controller and observer. For lateral motion stabilization, a sliding mode control scheme was adopted with a fish-like tail as the system control input, which depended only on the pitch angle information. Both simulation and experimental results were presented.

A 2-DOF control strategy was proposed for three-dimensional curve tracking problem for gliding robotic fish, by following the trajectory characteristic parameters (curvature, torsion and speed). We investigated the steady spiral motion and its geometric characteristics. A feedforward controller was designed first via inverse mapping of steady spiral motion. A 2-DOF control design was then proposed, which included an inverse mapping feedforward controller and a robust H_∞ feedback controller, designed based on the linearized model. Simulation was conducted with comparison to PI control and pure feedforward inverse mapping control. The simulation results were presented to verify the effectiveness of the proposed 2-DOF control design.

At last, full functions of the developed gliding robotic fish prototype were tested in field. The results of field tests, including crude oil detection in Kalamazoo River and algae sampling in Wintergreen Lake, were presented.

10.2 Future Work

First, in both the lab/pool experiments and the field tests, significant drifting of the robot was observed that was caused by the currents and waves of its aquatic environment, e.g., the water jets shooting from the side wall of the swimming pool. Thus an environmental current flow estimation

or prediction is desirable in disturbance rejection and precise locomotion. The information of the surrounding environments will benefit the controller design and the path following.

Second, a comprehensive path planning and path following control strategy is needed in water sampling tasks. Take the algae sampling in Wintergreen Lake as an example. It is of interest to develop adaptive sampling strategies and explore the combination of elementary energy-efficient motions (steady gliding and steady spiraling), to cover the whole lake with high sampling resolution and and low energy consumption.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] C. C. Ericksen, T. J. Osse, R. D. Light, T. Wen, T. W. Lehman, P. L. Sabin, J. W. Ballard, and A. M. Chiodi, "Seaglider: A long-range autonomous underwater vehicle for oceanographic research," *IEEE Journal of Oceanic Engineering*, vol. 26, pp. 424–436, 2001.

[2] J. Sherman, R. E. Davis, W. B. Owens, and J. Valdes, "The autonomous underwater glider "Spray"," *IEEE Journal of Oceanic Engineering*, vol. 26, pp. 437–446, 2001.

[3] D. C. Webb, P. J. Simonetti, and C. P. Jones, "SLOCUM, an underwater glider propelled by environmental energy," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 447–452, 2001.

[4] S. American, "Researchers spawn a new breed of robotic fish." [Online]. Available: http://www.scientificamerican.com/article/robotic-fish/

[5] R. P. Schwarzenbach, T. Egli, T. B. Hofstetter, U. Von Gunten, and B. Wehrli, "Global water pollution and human health," *Annual Review of Environment and Resources*, vol. 35, pp. 109–136, 2010.

[6] N. F. Gray *et al.*, *Water technology: an introduction for environmental scientists and engineers.* IWA Publishing, 2010, no. Ed. 3.

[7] R. Rajagopalan, *Environmental studies: from crisis to cure.* Oxford University Press, 2011.

[8] United States National Oceanographic and Atmospheric Administration, "Erma gulf response." [Online]. Available: http://response.restoration.noaa.gov/maps-and-spatial-data/ environmental-response-management-application-erma/erma-gulf-response.html

[9] Office of the Maritime Administrator, "Deepwater horizon marine casualty investigation report," August 2011, Retrieved on February 25, 2013.

[10] USEPA, "EPA Response to Enbridge Spill in Michigan." [Online]. Available: http://www.epa.gov/enbridgespill/documents.html

[11] Wikipedia, "Xingang port oil spill." [Online]. Available: http://en.wikipedia.org/wiki/ Xingang_Port_oil_spill

[12] United Nations Environment Programme, "United Nations report on Nigeria oil spills." [Online]. Available: http://www.shell.com.ng/environment-society/our-response/ unep-response.html

[13] Wikipedia, "Oil Spill." [Online]. Available: http://en.wikipedia.org/wiki/Oil_spill

[14] S. A. Ruberg, R. W. Muzzi, S. B. Brandt, J. C. Lane, T. C. Miller, J. J. Gray, S. A. Constant, and E. J. Downing, "A wireless internet-based observatory: The Real-time Coastal Observation Network (ReCON)," in *Proceedings of Marine Technology Society/IEEE Oceans 2007 Conference*, 2007, 6 pp.

[15] H. Hongji, A. Kaneko, and K. Kawatate, "Self-governing profiling system," *Cont. Shelf Res.*, vol. 7, pp. 1257–1265, 1987.

[16] R. E. Davis, D. C. Webb, L. A. Regier, and J. Dufour, "The autonomous Lagrangian circulation explorer (ALACE)," *Journal of Atmospheric and Oceanic Technology*, vol. 9, pp. 264–285, 1992.

[17] R. A. Luettich, "PSWIMS, a profiling instrument system for remote physical and chemical measurements in shallow-water," *Estuaries*, vol. 16, pp. 190–197, 1993.

[18] K. W. Doherty, D. E. Frye, S. P. Liberatore, and J. M. Toole, "A moored profiling instrument," *J Atmos Ocean Technol.*, vol. 16, pp. 1816–1829, 1999.

[19] J. V. Reynolds-Fleming, J. G. Fleming, and R. A. Luettich, "Portable autonomous vertical profiler for estuarine applications," *Estuaries*, vol. 25, pp. 142–147, 2002.

[20] J. B. Newman and B. H. Robison, "Development of a dedicated ROV for ocean science," *Marine Technology Society Journal*, vol. 26, pp. 46–53, 1994.

[21] J. Lee, M. Roh, K. Kim, and D. Lee, "Design of autonomous underwater vehicles for cage aquafarms," in *Proceedings of the 2007 IEEE Intelligent Vehicles Symposium*, Istanbul, Turkey, 2007, pp. 938–943.

[22] ScienceDaily, "Rensselaer researchers experiment with solar underwater robots," December 2004. [Online]. Available: http://www.sciencedaily.com/releases/2004/12/041212081548. htm

[23] G. S. Sukhatme, A. Dhariwal, B. Zhang, C. Oberg, B. Stauffer, and D. A. Caron, "Design and development of a wireless robotic networked aquatic microbial observing system," *Environmental Engineering Science*, vol. 24, no. 2, pp. 205–215, 2007.

[24] D. R. Yoerger, A. M. Bradley, M. Jakuba, C. R. German, T. Shank, and M. Tivey, "Autonomous and remotely operated vehicle technology for hydrothermal vent discovery, exploration, and sampling," *Oceanography*, vol. 20, no. 1, pp. 152–161, 2007.

[25] J. R. Higinbotham, J. R. Moisan, M. Schirtzinger, C.and Linkswiler, and P. Yungel, J.and Orton, "Update on the development and testing of a new long duration solar powered autonomous surface vehicle," in *Proceedings of the 2008 OCEANS Conference*, Quebec City, QC, 2008, 10 pp.

[26] M. Stealey, A. Singh, M. Batalin, B. Jordan, and W. Kaiser, "NIMS-AQ: A novel system for autonomous sensing of aquatic environments," in *Proceedings of the 2008 IEEE International Conference on Robotics and Automation*, Pasadena, CA, 2008, 621-628.

[27] A. Goldstein and S. Bentley, "Use of highly portable micro-sized remotely operated vehicles for environmental monitoring and mapping," in *Proceedings of OCEANS 2010 Conference*, Seattle, WA, 2010, 6 pp.

[28] J. Das, H. Heidarsson, A. Pereira, F. Arrichiello, I. Cetnic, L. Darjany, M.-E. Garneau, M. Howard, C. Oberg, M. Ragan, E. Seubert, E. Smith, B. Stauffer, A. Schnetzer, G. Toro-Farmer, D. Caron, B. Jones, and G. Sukhatme, "USC CINAPS builds bridges," *IEEE Robot. Automat. Mag.*, vol. 17, no. 1, pp. 20–30, 2010.

[29] R. N. Smith, Y. Chao, P. P. Li, D. A. Caron, B. H. Jones, and G. S. Sukhatme, "Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean model," *International Journal of Robotics Research*, vol. 29, no. 12, pp. 1475–1497, 2010.

[30] F. S. Hover, R. M. Eustice, A. Kim, B. Englot, H. Johannsson, M. Kaess, and J. J. Leonard, "Advanced perception, navigation and planning for autonomous in-water ship hull inspection," *The International Journal of Robotics Research*, vol. 31, no. 12, pp. 1445–1464, 2012.

[31] M. Dunbabin and A. Grinham, "Experimental evaluation of an Autonomous Surface Vehicle for water quality and greenhouse gas emission monitoring," in *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*, Anchorage, AK, 2010, pp. 5268–5274.

[32] L. Elkins, D. Sellers, and W. R. Monach, "The Autonomous Maritime Navigation (AMN) project: Field tests, autonomous and cooperative behaviors, data fusion, sensors, and vehicles," *J. Field Robot.*, vol. 27, no. 6, pp. 790–818, 2010.

[33] S. J. M. Hughes, D. O. B. Jones, C.Hauton, A. R. Gates, and L. E.Hawkins, "An assessment of drilling disturbance on echinus acutus var. norvegicus based on in-situ observations and

experiments using a remotely operated vehicle (ROV)," *J. Exp. Marine Biol. Ecol.*, vol. 395, no. 1-2, pp. 37–47, 2010.

[34] S. B. Williams, O. Pizarro, J. Webster, R. Beaman, I. Mahon, M. Johnson-Roberson, and T. Bridge, "AUV-assisted surveying of drowned reefs on the shelf edge of the Great Barrier Reef, Australia," *J. Field Robot.*, vol. 27, no. 5, pp. 675–697, 2010.

[35] G. Hitz, F. Polerleau, M. Gameau, C. Pradalier, T. Posch, J. Pernthaler, and R. Y. Siegwart, "Autonomous inland water monitoring: Design and application of a surface vehicle," *IEEE Robot. Automat. Mag*, vol. 19, no. 1, pp. 62–72, 2012.

[36] S. B. Williams, O. R. Pizarro, M. V. Jakuba, C. R. Johnson, N. S. Barrett, R. C. Babcock, G. A. Kendrick, P. D. Steinberg, A. J. Heyward, P. J. Doherty, I. Mahon, M. Johnson-Roberson, D. Steinberg, and A. Friedman, "Monitoring of benthic reference sites: Using an autonomous underwater vehicle," *IEEE Robot. Automat. Mag.*, vol. 19, no. 1, pp. 73–84, 2012.

[37] P. R. Bandyopadhyay, "Trends in biorobotic autonomous undersea vehicles," *IEEE Journal of Oceanic Engineering*, vol. 30, no. 1, pp. 109–139, 2005.

[38] B. Bingham, B. Foley, H. Singh, R. Camilli, K. Delaporta, R. Eustice, A. Mallios, D. Mindell, C. Roman, and D. Sakellariou, "Robotic tools for deep water archaeology: Surveying an ancient shipwreck with an autonomous underwater vehicle," *J. Field Robot.*, vol. 27, no. 6, pp. 702–717, 2010.

[39] L. L. Whitcomb, M. V. Jakuba, J. C. Kinsey, S. C. Martin, S. E. Webster, J. C. Howland, C. L. Taylor, D. Gomez-Ibanez, and D. Yoerger, "Navigation and control of the Nereus hybrid underwater vehicle for global ocean science to 10,903 m depth: Preliminary results," in *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*, Anchorage, AK, 2010, 594-600.

[40] G. V. Lauder and E. G. Drucker, "Morphology and experimental hydrodynamics of fish fin control surfaces," *IEEE Journal of Oceanic Engineering*, vol. 29, no. 3, pp. 556–571, 2004.

[41] F. E. Fish and G. V. Lauder, "Passive and active flow control by swimming fishes and mammals," *Annual Review of Fluid Mechanics*, vol. 38, pp. 193–224, 2006.

[42] A. Punning, M. Anton, M. Kruusmaa, and A. Aabloo, "A biologically inspired ray-like underwater robot with electroactive polymer pectoral fins," in *International IEEE Conference on Mechatronics and Robotics*, vol. 2004, 2004, pp. 241–245.

[43] R. Pfeifer, M. Lungarella, and F. Iida, "Self-organization, embodiment, and biologically inspired robotics," *science*, vol. 318, no. 5853, pp. 1088–1093, 2007.

[44] R. Altendorfer, N. Moore, H. Komsuoglu, M. Buehler, H. Brown Jr, D. McMordie, U. Saranli, R. Full, and D. E. Koditschek, "Rhex: A biologically inspired hexapod runner," *Autonomous Robots*, vol. 11, no. 3, pp. 207–213, 2001.

[45] K. Melsaac and J. P. Ostrowski, "A geometric approach to anguilliform locomotion: modelling of an underwater eel robot," in *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, vol. 4. IEEE, 1999, pp. 2843–2848.

[46] E. Garcia, M. A. Jimenez, P. G. De Santos, and M. Armada, "The evolution of robotics research," *Robotics & Automation Magazine, IEEE*, vol. 14, no. 1, pp. 90–103, 2007.

[47] D. L. Rudnick, R. E. Davis, C. C. Eriksen, D. M. Fratantoni, and M. J. Perry, "Underwater gliders for ocean research," *Marine Technology Society Jouranl*, vol. 38, no. 2, pp. 73–84, 2004.

[48] M. S. Triantafyllou and G. S. Triantafyllou, "An efficient swimming machine," *Scientific America*, vol. 273, no. 3, pp. 64–70, 1995.

[49] N. Kato, "Control performance in the horizontal plane of a fish robot with mechanical pectoral fins," *IEEE Journal of Oceanic Engineering*, vol. 25, no. 1, pp. 121–129, 2000.

[50] J. Anderson and N. Chhabra, "Maneuvering and stability performance of a robotic tuna," *Integrative and Comparative Biology*, vol. 42, no. 1, pp. 118–126, 2002.

[51] P. R. Bandyopadhyay, "Maneuvering hydrodynamics of fish and small underwater vehicles," *Integrative and Comparative Biology*, vol. 42, pp. 102–117, 2002.

[52] J. Yu, M. Tan, S. Wang, and E. Chen, "Development of a biomimetic robotic fish and its control algorithm," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 4, pp. 1798–1810, 2004.

[53] J. H. Long, A. C. Lammert, C. A. Pell, M. Kemp, J. A. Strother, H. C. Crenshaw, and M. J. McHenry, "A navigational primitive: Biorobotic implementation of cycloptic helical klinotaxis in planar motion," *IEEE Journal of Oceanic Engineering*, vol. 29, no. 3, pp. 795–806, 2004.

[54] P. V. Alvarado and K. Youcef-Toumi, "Design of machines with compliant bodies for biomimetic locomotion in liquid environments," *Journal of Dynamic Systems, Measurement, and Control*, vol. 128, pp. 3–13, 2006.

[55] J. Liu and H. Hu, "Biologically inspired behavior design for autonomous robotic fish," *International Journal of Automation and Computing*, vol. 4, pp. 336–347, 2006.

[56] M. Epstein, J. E. Colgate, and M. A. MacIver, "Generating thrust with a biologically-inspired robotic ribbon fin," in *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, China, 2006, pp. 2412–2417.

[57] M. Krieg and K. Mohseni, "Dynamic modeling and control of biologically inspired vortex ring thrusters for underwater robot locomotion," *IEEE Transactions on Robotics*, vol. 26, no. 3, pp. 542–554, 2010.

[58] K. A. Morgansen, B. I. Triplett, and D. J. Klein, "Geometric methods for modeling and control of free-swimming fin-actuated underwater vehicles," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1184–1199, 2007.

[59] G. V. Lauder, E. J. Anderson, J. Tangorra, and P. G. A. Madden, "Fish biorobotics: Kinematics and hydrodynamics of self-propulsion," *Journal of Experimental Biology*, vol. 210, pp. 2767–2780, 2007.

[60] P. Kodati, J. Hinkle, A. Winn, and X. Deng, "Microautonomous robotic ostraciiform (MARCO): Hydrodynamics, design, and fabrication," *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 105–117, 2008.

[61] K. H. Low and C. W. Chong, "Parametric study of the swimming performance of a fish robot propelled by a flexible caudal fin," *Bioinsp. Biomim.*, vol. 5, p. 046002, 2010.

[62] Z. C. S. Shatara and X. Tan, "Modeling of biomimetic robotic fish propelled by an ionic polymer-metal composite caudal fin," *IEEE/ASME Transactions on Mechatronics*, vol. 15, no. 3, pp. 448–459, 2010.

[63] M. Aureli, V. Kopman, and M. Porfiri, "Free-locomotion of underwater vehicles actuated by ionic polymer metal composites," *IEEE/ASME Transactions on Mechatronics*, vol. 15, no. 4, pp. 603–614, 2010.

[64] J. Yu, R. Ding, Q. Yang, M. Tan, W. Wang, and J. Zhang, "On a bio-inspired amphibious robot capable of multimodal motion," *IEEE/ASME Transactions on Mechatronics*, vol. PP, no. 99, pp. 1–10, 2011.

[65] P. C. Strefling, A. M. Hellum, and R. Mukherjee, "Modeling, simulation, and performance of a synergistically propelled ichthyoid," *IEEE/ASME Transactions on Mechatronics*, vol. 17, no. 1, pp. 36–45, 2012.

[66] F. Liu, K.-M. Lee, and C.-J. Yang, "Hydrodynamics of an undulating fin for a wave-like locomotion system design," *Mechatronics, IEEE/ASME Transactions on*, vol. 17, no. 3, pp. 554–562, 2012.

[67] C. Zhou and K. H. Low, "Design and locomotion control of a biomimetic underwater vehicle with fin propulsion," *IEEE/ASME Transactions on Mechatronics*, vol. 17, no. 1, pp. 25–35, 2012.

[68] S. Marras and M. Porfiri, "Fish and robots swimming together: attraction towards the robot demands biomimetic locomotion," *Journal of The Royal Society Interface*, vol. 9, no. 73, pp. 1856–1868, 2012.

[69] M. Schwartz, *Encyclopedia of coastal science*.   Kluwer Academic Pub, 2005, vol. 24.

[70] R. E. Davis, N. E. Leonard, and D. M. Fratantoni, "Routing strategies for underwater gliders," *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 56, no. 3-5, pp. 173 – 187, 2009.

[71] D. Blidberg, "The development of autonomous underwater vehicles (AUV); a brief summary," in *IEEE ICRA*, Seoul, Korea, 2001.

[72] N. Mahmoudian, "Efficient motion planning and control for underwater gliders," Ph.D. dissertation, Virginia Polytechnic Institute and State University, 2009.

[73] N. E. Leonard and J. G. Graver, "Model-based feedback control of autonomous underwater gliders," *IEEE Journal of Oceanic Engineering*, vol. 26, no. 4, pp. 633–645, 2001.

[74] N. Mahmoudian and C. A. Woolsey, "Underwater glider motion control," in *Proceedings of IEEE Conference on Decision and Control*, Cancun, Mexico, 2008, pp. 552–557.

[75] P. Bhatta, "Nonlinear stability and control of gliding vehicles," Ph.D. dissertation, Princeton University, 2006.

[76] P. Bhatta and N. Leonard, "Nonlinear gliding stability and control for vehicles with hydrodynamic forcing," *Automatica*, vol. 44, no. 5, pp. 1240–1250, 2008.

[77] J. G. Graver, "Underwater gliders: Dynamics, control, and design," Ph.D. dissertation, Princeton University, 2005.

[78] F. Zhang, X. Tan, and H. K.Khalil, "Passivity-based controller design for stabilization of underwater gliders," in *Proceedings of the 2012 American Control Conference*, Montreal, Canada, 2012, pp. 5408–5413.

[79] N. Mahmoudian, J. Geisbert, and C. Woolsey, "Approximate analytical turning conditions for underwater gliders: Implications for motion control and path planning," *IEEE Journal of Oceanic Engineering*, vol. 35, no. 1, pp. 131–143, 2010.

[80] S. Zhang, J. Yu, A. Zhang, and F. Zhang, "Spiraling motion of underwater gliders: Modeling, analysis, and experimental results," *Ocean Engineering*, vol. 60, pp. 1–13, 2013.

[81] F. Zhang, F. Zhang, and X. Tan, "Steady spiraling motion of gliding robotic fish," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2012, pp. 1754–1759.

[82] F. Zhang, J. Thon, C. Thon, and X. Tan, "Miniature underwater glider: Design, modeling, and experimental results," in *Proceedings of the 2012 IEEE International Conference on Robotics and Automation*, St. Paul, MN, 2012, pp. 4904–4910.

[83] ——, "Miniature underwater glider: Design and experimental results," *IEEE/ASME Transactions on Mechatronics*, 2014, vol. 19, no. 1, pp. 394-399.

[84] F. Zhang, J. Wang, J. Thon, C. Thon, E. Litchman, and X. Tan, "Gliding robotic fish for mobile sampling of aquatic environments," in *Proceedings of 11th IEEE International Conference on Networking, Sensing and Control*, Miami, FL, 2014, accepted.

[85] J. Milgram, "Strip theory for underwater vehicles in water of finite depth," *Journal of Engineering Mathematics*, vol. 58, no. 1, pp. 31–50, 2007.

[86] R. L. Panton, *Incompressible Flow*. New York: Wiley, 2005.

[87] D. T. Greenwood, *Principles of Dynamics*. Englewood Cliffs, NJ.: Prentice Hall, 1988.

[88] J. D. Anderson, *Aircraft Performance and Design*. New York: McGraw-Hill, 1998.

[89] F. O. Smetana, *Introductory aerodynamics and hydrodynamics of wings and bodies: a software-based approach*. AIAA, 1997, vol. 1.

[90] R. Von Mises, *Theory of flight*. Dover Pubns, 1959.

[91] I. G. Currie, *Fundamental Mechanics of Fluids*. New York: McGraw-Hill, 1974.

[92] K. Fujii, "Progress and future prospects of CFD in aerospace wind tunnel and beyond," *Prog Aerospace Sci*, vol. 41, pp. 455–470, 2005.

[93] The Autonomous Underwater Vehicle Applications Center (AUVAC), "AUVs database in AUVAC." [Online]. Available: http://auvac.org/explore-database/simple-search

[94] S. Zhang, J. Yu, A. Zhang, and F. Zhang, "Steady three dimensional gliding motion of an underwater glider," in *2011 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2011, pp. 2392–2397.

[95] C. Kelley, *Solving Nonlinear Equations with Newton's Method*. Society for Industrial Mathematics, 2003.

[96] H. Khalil, *Nonlinear systems*. Upper Saddle River, N.J.: Prentice Hall, 2002.

[97] P. Kokotović, H. Khalil, and J. O'Reilly, *Singular perturbation methods in control: analysis and design*. Society for Industrial Mathematics, 1999.

[98] F. Zhang and X. Tan, "Nonlinear observer design for stabilization of gliding robotic fish," in *Proceedings of the 2014 American Control Conference*, Portland, OR, 2014, accepted.

[99] ——, "Gliding robotic fish and its tail-enabled yaw motion stabilization using sliding mode control," in *Proceedings of the 2013 ASME Dynamic Systems and Control Conference*, Palo Alto, CA, 2013, paper DSCC2013-4015 (10 pp).

[100] F. W. Warner, *Foundations of differentiable manifolds and Lie groups*. Springer, 1983, vol. 94.

[101] F. Zhang and X. Tan, "Three-dimensional spiral tracking control for gliding robotic fish," in *Proceedings of the 53rd IEEE Conference on Decision and Control*, Los Angeles, CA, 2014, accepted.

[102] A. Pressley, *Elementary differential geometry*. Springer, 2010.