Design of an Underwater Vertical Glider for Subsea Equipment Delivery

by

Charles Kirby Ambler

B.S., Yale University (2005)

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2010

Author .....
Department of Mechanical Engineering
December 9, 2009

Certified by .....
Franz S. Hover
Assistant Professor of Mechanical and Ocean Engineering
Thesis Supervisor

Certified by .....
Julio C. Guerrero, Ph.D.
Principal Research Scientist, Schlumberger-Doll Research
Thesis Supervisor

Accepted by .....
David E. Hardt
Chairman, Committee on Graduate Students

Design of an Underwater Vertical Glider for Subsea Equipment Delivery

by

Charles Kirby Ambler

Submitted to the Department of Mechanical Engineering
on December 9, 2009, in partial fulfillment of the
requirements for the degree of
Master of Science in Mechanical Engineering

Abstract

Delivery of subsea equipment and sensors is generally accomplished with unguided sinking platforms or powered autonomous underwater vehicles (AUVs). An alternative would be to augment existing platforms with navigation and guidance capability, enabling them to actively guide themselves to their destination, with minimal added complexity and power consumption. This defines a new class of AUV having no propulsion, which we call the Vertical Glider.

This thesis investigates the challenges posed by this deployment concept, and describes in detail a prototype vertical glider that was built for initial tests. We explore through computer simulation the specific roles of various operating parameters, such as control gain, measurement noise, and process noise, on the overall vehicle performance. The prototype vehicle has been successfully pool-tested, and serves as a baseline platform for open water operations and multi-vehicle deployments.

Thesis Supervisor: Franz S. Hover
Title: Assistant Professor of Mechanical and Ocean Engineering

Acknowledgments

I would first like to thank my advisor, Professor Franz Hover, who was amazingly helpful throughout the course of my project. Your advice and guidance were the best I could have hoped for from an advisor. I would also like to thank our industry support, Schlumberger-Doll Research, and Dr. Julio Guerrero, who was instrumental in the development of the initial concept for this project as well as advocating our research. Also thank you to the Singapore-MIT Alliance (SMART), who supported me early in my MIT career.

I would like to thank all the members of the Hover group, who have helped me in my research and been great friends during my time at MIT, including Brendan, Josh, Lynn, Kyle, and Eric. Best of luck with the rest of your studies. Thanks to the undergrad and grad students would worked on some part of the project with me, including Michael, Victor, Stephanie, Brooks, and Rob for your help in pool testing, hardware, and software development. Many thanks to Mark Belanger at the Edgerton student machine shop, who was a valuable source of knowledge while I was building the prototype vehicle.

Finally, I would like to thank my parents, my sister, and my grandmother for their love and support throughout my life. Without you none of this would be possible.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Project Motivations, Applications, and Challenges

There is strong interest within the subsea oil exploration and oceanographic communities in the autonomous delivery of equipment and sensor systems to precise locations on the seafloor. Current methods of subsea delivery include powered underwater vehicles and unguided platforms, or landers, which provide a stable frame on which to mount sensors and other equipment. Oil exploration makes use of a variety of systems such as electromagnetic sensors (EMS), which are integrated into landers and placed on the seafloor to detect deposits of oil and gas. These landers often operate at full ocean depths of up to 4 km, and are generally deployed in a passive manner, whereby the surface vessel is positioned over the target, the platform is deployed from the ship and allowed to free fall to the bottom. Once it reaches the bottom, the lander's position can be determined by ship-based sensors. However, missions of this type would benefit greatly from a guided delivery. By adding active control and navigation to the lander, we can eliminate the need to survey deployed sensors, and greatly improve the regularity of spacing between multiple sensors, which will aid in the detection accuracy of the sensor system.

In addition to oil exploration, the field of oceanography would benefit from guided

deployments by allowing researchers to place sensors very accurately near spots of interest. Lander vehicles are used to study fish behavior and population patterns, as well as to capture live fish to bring back to the surface. Additionally, landers can be used to investigate the water currents near the ocean floor [16]. A guided platform would allow researchers to accurately return sensors to previously explored areas, or to aid in the retrieval of scientific samples.

The limitations of current systems present the opportunity for a new class of vehicle that could fill the gap left by current methods. However, there is still a variety of challenges faced by subsea deployment, including efficient power usage, data retrieval, navigation, and control. In this thesis we seek to:

1. Analyze the broad operational challenges facing subsea equipment delivery missions with minimum control, through theory and simulations.

2. Build a prototype of a new class of AUV's dubbed the Vertical Glider Robot (VGR), and test it to address certain practical questions.

1.2 Current and Prior Research

The field of subsea equipment delivery makes use of a variety of technologies and methods related to underwater vehicles, communications and navigation. Before embarking on the task of designing a new platform and analyzing its behavior, we must first discuss the current state of the art and how it can inform our design process.

1.2.1 Powered Underwater Vehicles

Current subsea equipment delivery methods make use of remotely operated vehicles (ROV), autonomous underwater vehicles (AUV), and unguided methods such as tow cables and free fall drops. However, for the purposes of autonomous delivery, these methods are less than ideal. Because of the high bandwidth needed for manual control, ROVs require a tether attached to a surface support vessel for communications, video relay, and depending on the mission, power as well. AUVs can operate without

a tether, but are expensive and thus cannot be deployed in the quantities needed for equipment delivery missions. Deep ocean AUVs such as the Autonomous Benthic Explorer (ABE) [27] have the depth capability for a seafloor equipment delivery mission, but they are also very complex, and often have more capabilities than what is needed for equipment delivery.

For equipment delivery missions, the main disadvantage of AUV's and ROV's is that operating and material costs are often on par with the value of the equipment or sensors being deployed. It is not very cost effective to use a single deep sea AUV worth $1-2 million to deploy a group of sensors worth only $50k-70k each, especially considering the costs of maintaining a support vessel during the mission, and the risk of damaging the vehicle over many successive deployments. By outfitting sensor platforms with inexpensive steering mechanisms, operators could also deploy multiple vehicles simultaneously instead of relying on a single AUV.

The REMUS AUV is an underwater vehicle originally developed by the Woods Hole Oceanographic Institution (WHOI). While small and relatively inexpensive, it requires manpower to deploy and monitor the vehicle during missions, often operating in rough conditions. Researchers at WHOI have developed a docking platform to enable the REMUS to dock, recharge its batteries, and transfer data, while being protected from potentially damaging ocean conditions [22]. The main feature of this docking system is the utilization of a USBL system as a homing beacon to guide the vehicle into the dock. While the vertical glider system described in this thesis travels further away from its navigation system as it dives, future subsea equipment delivery missions may involve tracking acoustic beacons on the seafloor, and may involve docking mechanisms to provide power and to collect data from seafloor sensors before returning to the surface.

1.2.2 Passively Falling Vehicles and Gliders

The task of precision underwater delivery does not necessarily require propulsion, but only steering. There is a class of underwater vehicles called gliders that have no propulsion system other than their ability to change buoyancy, enabling gradual

descent and ascent. Most applications to glider technology are geared towards long term, long distance missions over open ocean, which take advantage of the gliders' strengths, namely that of minimal power usage. One of the more widely known glider concepts is the Slocum [24], developed by Douglas Webb and researchers at WHOI. The Slocum was developed as two models: one is powered by internal batteries while the other uses a thermal engine, exploiting the temperature gradients in the ocean for energy. Other well known glider vehicles are the Spray [19], developed by Scripps and WHOI, and the Seaglider [7], developed at the University of Washington and currently being commercialized by iRobot Corporation.

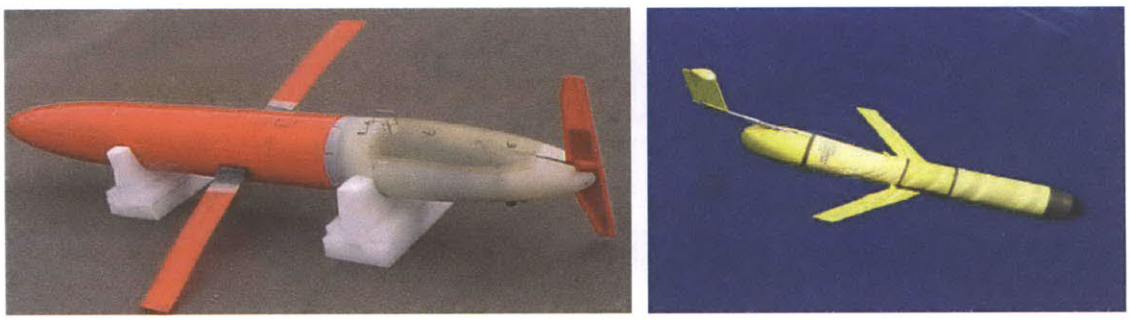

Figure 1-1: The Spray[1] (left) and Slocum[2] gliders (right)

The field of horizontal gliders has been well researched, and this large body of work can assist in the design of vertical gliders. While not as dependent on fine tuned buoyancy changes, a vertical glider performing an equipment delivery mission would benefit from precise knowledge of vehicle dynamics, and the design challenges faced by other glider vehicles that need to change orientation during a mission.

In addition to horizontal gliders, we also sought out current research on vehicles intended to move primarily in the vertical direction. The most closely related work on vertically oriented AUV's was conducted by researchers at the University of Rhode Island, on the vehicle called Mini Ocean Elevator, or MiniOE [6]. The MiniOE is a testbed vehicle for investigating the dynamics of a vertical/horizontal AUV concept. A model for the vehicle was adapted from the REMUS vehicle model originally developed at MIT [15]. Simulations and open water testing were conducted to determine

[1] http://www.bluefinrobotics.com/bluefin_glider.htm
[2] http://www.whoi.edu/page.do?pid=7545&tid=282&cid=37008&ct=162

righting moments and vehicle stability, although no active control was performed by the vehicle. The MiniOE is intended as a depth profiler for a variety of sensor measurements and is not intended to track a target or provide controlled flight. However, the model developed for the vehicle is specifically oriented towards operating in the vertical direction, and would be valuable for developing a full six degree of freedom model for a vertically oriented glider.

1.2.3 Near Seafloor Operations and Lander Vehicles

In addition to gliders, there is also work being conducted with powered AUVs for use in near seafloor operations. Researchers at the University of Limerick developed a vehicle to perform survey and sensor delivery missions in uncertain terrain and current conditions [13], that can operate in a hybrid AUV/ROV mode. Vertical gliders operating in complex environments could make use of this strategy to enable the glider to operate autonomously or manually if more precise control is required.

Lander vehicles could be useful not only in subsea oil and gas exploration, but in the environmental community as well. On-board sensors can provide critical data about ocean conditions, such as temperature, salinity, and pH levels. There is also significant interest in using lander vehicles for monitoring schools of fish, seafloor boundary layer flow, and sediment analysis [16]. One notable development is the Aberdeen University Deep Ocean Submersible (AUDOS), which has been used to deliver bait traps to the seafloor to observe the eating habits of fish, and to record the types of fish encountered using cameras [17].

Another lander platform. the IFM-GEOMAR Deep-Sea Lander (Figure 1-2), is a large aluminum-framed platform for deep ocean sensing [14]. It is generally deployed from the surface and allowed to free fall, although it can also be towed underwater, and released from the tow cable when the lander is in range of a suitable target. However, the current state of lander vehicles is as a passive platform on which to attach sensor packages. There is no control of the vehicle's path beyond surface deployment and releasing the lander once its mission is completed.

[1]http://www.ifm-geomar.de/index.php?id=1200&L=1

Figure 1-2: IFM-GEOMAR Deep-sea Lander[1]

1.2.4 Underwater Vehicle Navigation

Underwater vehicles face unique challenges in navigation. Methods of navigation used by land and air vehicles such as GPS are infeasible, due to the inability of radio waves to penetrate water more than a few meters. For this reason, almost all underwater vehicle platforms use acoustic signals to perform navigation and transmit information. Several systems are in use that seek to provide absolute positioning to vehicles beneath the surface, using precisely timed acoustic signals sent from the AUV to several acoustic transceivers. These systems are generally classified according to the distance between the elements of the transponder array.

A common navigation system in use by AUVs is the Ultra-Short Baseline (USBL) system, so called because the transmitting and receiving elements of the system are on the order of centimeters apart from each other. This allows the system to be contained within a single apparatus, except for the remote transponder, and it is usually mounted to the hull of the surface vessel operating the AUV. Combined with roll, pitch, compass, and GPS sensors mounted on the surface vessel, USBL systems can be configured to report the absolute position of the AUV, or its position relative to the boat in Cartesian space. The USBL system operates by sending pings out from

the boat to the AUV, and listening for a response with three or more transponders. By measuring the time needed to receive a response, and comparing the differences in response times between the transponders, the system can determine the range and bearing to the target vehicle [23]. The main limitations of USBL navigation are that X-Y position accuracy worsens with increasng distance between the vehicle and transponder, and the rate of position updates becomes less frequent. These two factors will be discussed in further detail in simulation in Chapter 3. The performance parameters of some commerical USBL systems are describe in Table 1.1. The systems referenced feature navigation accuracies of approximately 0.2-0.3% of range.

The other system most commonly used is the Long Baseline (LBL) acoustic positioning system. The LBL system operates in a manner very similar to GPS, measuring the range to a target from several transponders. LBL systems provide position accuracy that is independent of depth, as long as the vehicle of interest is within the net of transponders [23]. However, due to the use of multiple transponders, LBL systems require more effort to setup and calibrate. The GPS Intelligent buoy (GIB)[2] provides LBL navigation coupled with GPS receivers to enable easy sensor calibration.

Additional navigation methods for underwater vehicles include inertial navigation, depth, altitude, and magnetic sensors. While inertial navigation is common on air- and spacecraft, the equipment required is generally very expensive, and are only suited for installation on larger underwater vehicles [9]. However, depth sensors, sonic altimeters, and magnetic compasses are all relatively inexpensive sensors that are useful not only for providing high-rate measurements, but are also free from the integrator drift associated with inertial sensors.

The main challenges of underwater navigation are the physical limitations inherent in acoustic systems, namely noisy measurements and delayed updates due to the low speed of sound in water ($\approx$1500 m/s). The key to providing a vehicle with accurate position and velocity information is to either use a system that has minimal uncertainties for the operating conditions intended, or to utilize multiple systems that complement each other. Recent research has begun to explore the uses of such

[2]http://www.underwater-gps.com/

System	Type	Position Accuracy (% of range)	Range (km)	Note
HiPAP 100	USBL	0.2	6.5	low freq, ultra deep water
HiPAP 350	USBL	0.3	3	medium depth, medium accuracy
HiPAP 500	USBL	0.2	4	medium depth, high accuracy
IXSEA GAPS	USBL	0.2	4	portable, integrated GPS and INS
IXSEA Posidonia	USBL	0.3	8	deep, long range

Table 1.1: Survey of Acoustic Navigation Systems [1,2]

"hybrid" navigation systems, combining traditional acoustic navigation systems with either inertial sensors or Doppler velocity loggers (DVL), which are capable of high resolution tracking of solid surfaces [18,26]. While acoustic navigation estimates are noisy and slow to update, they are geo-referenced measurements, and so do not drift over time. The DVL sensor can provide very accurate, quick updates to position, but will develop systematic errors over time unless they are corrected. By combining the data from multiple sensors, operators can develop a more accurate picture of vehicle position.

Underwater communications face the same challenges as navigation systems due to limitations set by the speed of sound. The bulk of long range underwater communications is still carried out through acoustic modems. However, there are significant limitations to current technology, notably low data rates and unpredictable link conditions, caused by sound signals bouncing off undersea structures and the water's surface. Past research in underwater communications has sought to make more efficient use of bandwidth to overcome signal attenuation and multipath effects, achieving data rates as high as 40 kb/s [21]. Researchers at WHOI have also implemented combination systems consisting of a surface buoy with radio and acoustic transceivers, which allows operators to remain on land and communicate with multiple vehicles [20].

The common method of overcoming acoustic limitations is to use fiber optic tethers for communication. Vehicles that run on internal power can use much lighter tethers

than those that draw power from the surface. The Nereus vehicle, in use by WHOI, uses a fiber optic communications tether the width of a human hair to communicate with operators on the surface [12]. The tether is able to unspool up to 40 km, and if broken during a mission the vehicle is able to automatically return to the surface. If the bandwidth provided by acoustic links is insufficient, a fiber optic tether could provide a subsea delivery vehicle with the capability needed for accurate delivery, as well as send back video of the target area during deployment.

1.3 Outline of Thesis

This thesis is organized into the following chapters:

- Chapter 2: Concept Generation and Selection - The development of possible solutions to the problem of precision equipment delivery, ranging from simple operational or software changes to entirely new underwater vehicles to accomplish the task.

- Chapter 3: Control Systems and Simulation - An analysis of the dynamics of a vertical glider robot, the governing physical parameters, and various control strategies to effectively guide it; a discussion of the results of computer simulations intended to better explore the effects of these parameters on vehicle performance.

- Chapter 4: Vehicle Design, Fabrication, and Testing - The design and fabrication of the prototype vehicle, as well as the results of pool tests with the vehicle.

- Chapter 5: Conclusions - A discussion of the implications of this research, results, and future work to be conducted.

Chapter 2

Concept Generation and Selection

2.1 Functional Requirements

The problem of accurately delivering sensor or equipment packages without fine control to the seafloor is very open-ended, with many solutions of varying complexity and cost. We sought to identify a number of solutions that would meet the needs of the mission, but first we needed to define what the primary mission goals are. Consulting with researchers at Schlumberger Corporation, we identified a reference platform for delivering electromagnetic sensors (EMS), which operates at depths of 4-6 km, at a dive rate of 1 m/s. In free fall delivery, this platform has a placement accuracy of 50 meters.

1. Deliver the package with an increased accuracy, from current error of 50 meters down to 5 meters.

2. Protect the package from physical damage during handling, deployment, and landing.

3. Maintain vertical orientation of the package, for proper bottom contact

2.2 Seven Design Ideas

During the initial concept generation phase, we developed several ideas that could accomplish the stated goals, while minimizing cost and complexity to the current apparatus. They range from simple changes in operational procedure and software to additional hardware to enable full navigation and control of existing vehicles.

2.2.1 Tracking During Descent

Current methods for deployment of electromagnetic sensors (EMS) involve an unguided drop, with no tracking during the deployment. Once the EMS has reached the seafloor, the USBL system on the surface ship begins sending pings to the EMS, and awaits a response. As an alternative, the operators can begin pinging the EMS as soon as it is dropped in the water. This provides several advantages. First, the operators can obtain a larger sample of position data as the vehicle falls, hopefully having enough data to reliably estimate the vehicle's position by the time it reaches the seafloor. Currently, the surface vessel must spend approximately 30 minutes surveying the area around the EMS drop location before a position can be determined. Secondly, the position updates during descent can provide valuable information on the dynamics of the EMS craft as it falls. Horizontal drift, random perturbations, and dive rate can all be tracked using this method. If the vehicle's motion is relatively stable, this method can also provide information about ocean currents at various depths, and can inform operators about conditions for future deployments. This method is shown in Figure 2-1, with the vehicle responding to acoustic pings from the surface during the descent.

This method is the most technically simple to implement. It only requires turning on the USBL navigation system when the vehicle is deployed, as opposed to waiting until the vehicle reaches the bottom. Software changes may need to be made to take advantage of this, but they are relatively simple and inexpensive. On the downside, this method does not provide any new control of the sensor vehicle, nor does it improve the stability of the vehicle while in motion. Additionally, this method requires that

the ship remain stationary during deployment. If engine power is required to keep the ship in place, the performance of the USBL system may be affected.

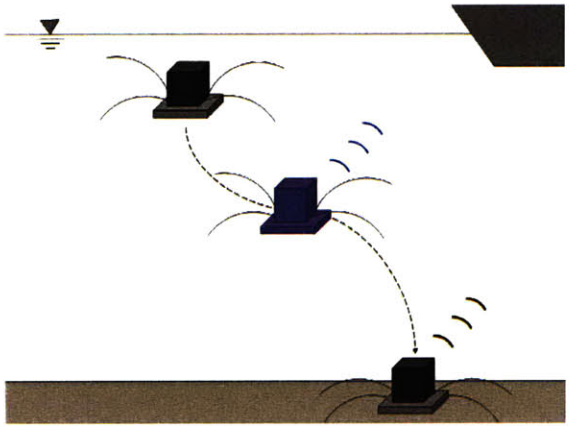

Figure 2-1: Tracking During Descent

2.2.2 Utilize One-Way Travel Time

The USBL navigation system operates on the principle of measuring range and bearing to a target via the time-of-flight of an acoustic signal. By knowing the speed of sound in water, operators can determine the range by sending out an acoustic signal and measuring the time it takes to receive a response. This requires waiting for the signal to travel to the vehicle and then return, which at full ocean depth can be a non-trivial delay, as shown in Equation 2.1.

$$\begin{aligned} \text{(Speed of sound in seawater) } v_s &\approx 1500 \text{ m/s} \\ \text{(Range to surface) } R &= 4000 \text{ m} \\ \text{(Two-way time of flight) } TOF &= \frac{2R}{v_s} \\ TOF &= 5.33 \text{ seconds} \end{aligned} \tag{2.1}$$

A solution to this problem is to find a way to use the USBL navigation system, but configure it so that only a one-way travel time is required. This will double the frequency with which operators receive navigation updates. To accomplish this,

both the surface vessel and the deployed vehicle must be fitted with accurate clocks whose times are synchronized. The vehicle can be programmed to start sending out acoustic signals at regular intervals as soon as it is deployed. The benefits of this method include significantly increased navigation update rates, with minimal hardware modification. The main technical requirement is installing accurate clocks on both vehicle and surface vessel, and ensuring they are synchronized before each mission. This is an important step, as a millisecond difference between the clocks can translate to a 1.5 meter range error. Recent research and experimental results [8] have shown that with proper clock configuration, one-way travel time navigation can achieve navigation accuracies on par with 12 kHz LBL systems. Figure 2-2 shows a diagram of the vehicle sending out signals at fixed intervals once reaching the seafloor.

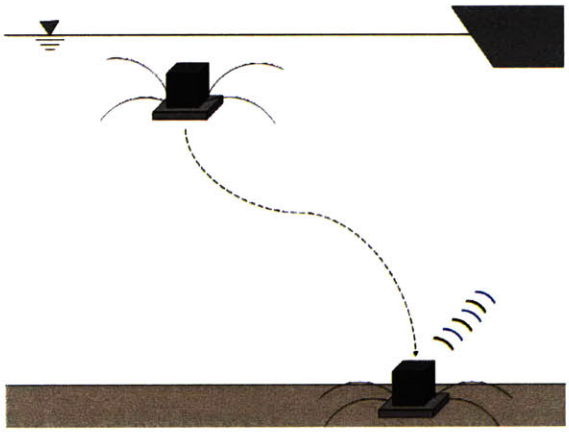

Figure 2-2: Utilize One-Way Travel Time

2.2.3 Large Simultaneous Deployment

The current method of EMS deployment is mainly one of trial and error. The vehicle has no control system to speak of, and so the operators must estimate the ocean currents and dynamics of the vehicle, and make an educated guess on where to drop the vehicle so as to ensure the best chance of the vehicle landing on its desired location. However, once the vehicle is in the water, there is nothing the operators can do but wait. This method currently yields drop accuracy on the order of 50 meters. If for some reason the vehicle lands outside of a desired target area, operators send an

acoustic signal to order the vehicle to release its weights and return to the surface. The process is then repeated. This entire operation takes about 30 minutes to complete.

A possible solution to this is a simultaneous deployment of several vehicles in the target area. By representing the vehicle landing zone as a normally distributed random variable, we can explore the effects of multiple simultaneous drops. Assume from this argument, for a single vehicle the likelihood of the vehicle landing within one standard deviation (in this case, 50 meters) of the target is 68%. Also assume that there is no correlation between separate drops. If two vehicles are deployed at the same time, the chance of at least one of them landing withing 50 meters of the target is $1 - .32^2 = .8976$. For a deployment of five vehicles, this probability increases to $1 - .32^5 = .9966$. This relation is depicted on the left side of Figure 2-4. As the number of vehicles (N) increases, the chance of one landing significantly closer to the target also increases. Whereas a single vehicle has a 68% chance of landing within 50 meters of the target, with five vehicles there is a 68% chance that at least one of them lands within 13 meters. This relation is depicted on the right side of Figure 2-4. Figure 2-3 shows a diagram of the deployment of three separate vehicles simultaneously. Upon reaching the seafloor, the vehicle denoted by the red "X" is determined to be closest to the target. This vehicle is left to complete the mission, while the other two are returned to the surface.

The idea of simultaneous deployment has several drawbacks, namely the greatly increased risk of collision of vehicles. If several vehicles are deployed in succession, there is an increased chance of them coming in contact both during descent and upon landing. Collision could cause damage to electronics, structural components, or affect the orientation of the vehicle upon reaching the seafloor, rendering its sensors ineffective. Additionally there is an increased demand on the crew of the surface vessel for simultaneous deployment, who are required to prepare and deploy more vehicles in a short period of time.

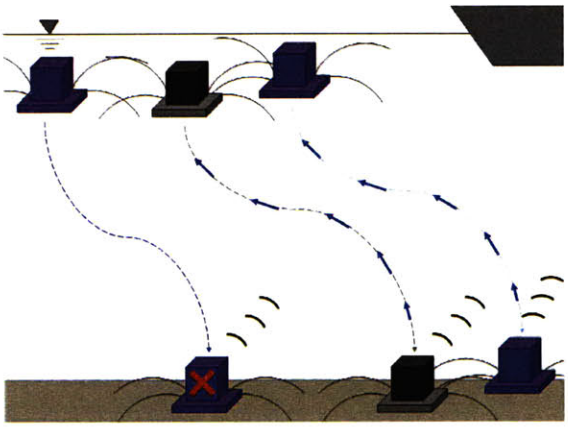

Figure 2-3: Large Simultaneous Deployment

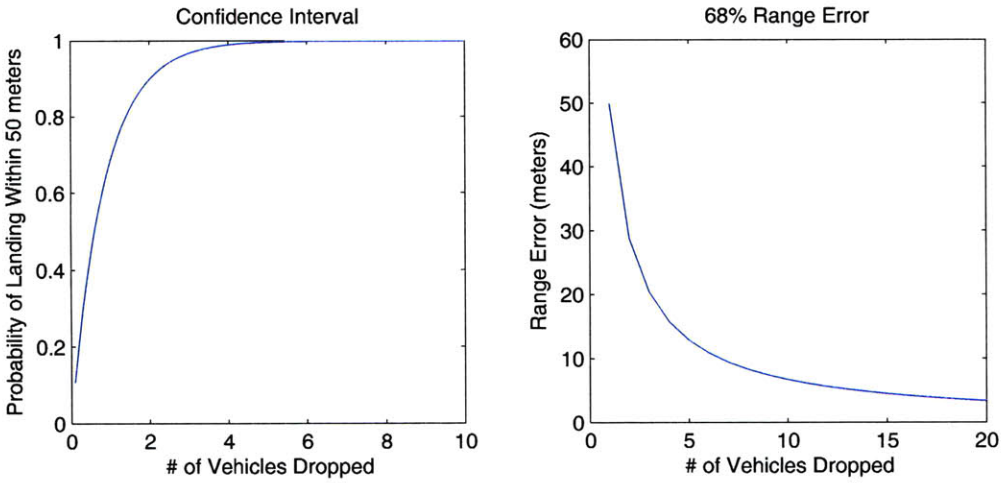

Figure 2-4: Plots of Simultaneous Deployment Performance Probability

2.2.4 Heavier, Streamlined Vehicle

The hydrodynamic behavior of the current EMS vehicle is generally unpredictable. Experimental observations have shown that the vehicle falls in a manner similar to a falling leaf, swaying from side to side as it descends. Modification of the EMS platform to incorporate a stable, streamlined body would allow it to fall in a more predictable manner during uncontrolled drops, as well as enable more reliable mechanisms for controlling it should that path be pursued.

By modeling the descent of the unguided vehicle as a random walk, where $\dot{x} = N(0, \sigma_w^2)$, then the landing error upon reaching the bottom would be represented by:

$$W_t = \int_0^T N(0, \sigma_w^2) dt$$
$$E(W_t) \approx \sigma_w \sqrt{T} \tag{2.2}$$

This random walk behavior is known as the Wiener process (synonymous with Brownian motion) [10]. The expected variance of a random walk over time is proportional to $\sqrt{T}$, so by reducing descent time, the landing error can be reduced. The main disadvantage of increased descent rate is the risk of equipment damage upon reaching the seafloor. While the vehicle's electronics would need to be shock-mounted regardless, any increase in descent speed makes proper safeguards for the vehicle's equipment more important. Figure 2-5 shows a diagram of a streamlined vehicle (in this case a teardrop shape) descending alongside a non-streamlined vehicle for comparison. By descending faster, the streamlined vehicle should be less susceptible to uncertain ocean currents.

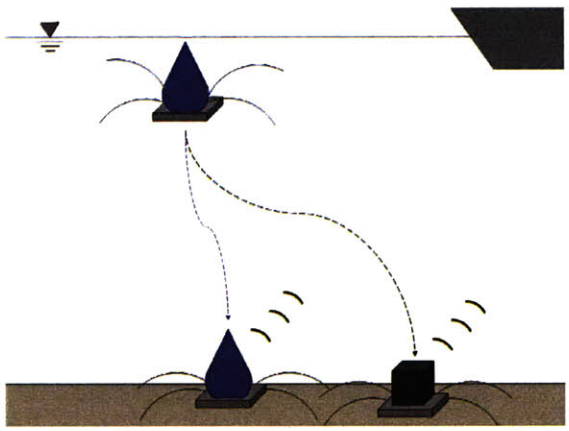

Figure 2-5: Heavier, Streamlined Vehicle

2.2.5 Tracking and Control During Descent

The concepts described thus far in this section have focused largely on providing relatively simple solutions to the problem of unreliable EMS delivery methods. Many are simple software changes, while the streamlined vehicle body requires more effort

and materials. The next step beyond a streamlined vehicle is to introduce an active navigation and control system to the vehicle, which would allow the vehicle to continually update its knowledge of its location, and be able to make course corrections during descent to account for any errors in its path. This method and its associated systems will be the focus of this thesis.

In addition to the increased landing accuracy, an active control system also provides operators with knowledge of underwater currents, assisting in mission planning. Over the span of many missions, deployment time will decrease, as fewer vehicles will miss their mark and need to be retrieved for a second deployment. The main disadvantage to this method is the greatly increased complexity of the delivery apparatus. In addition to the navigation and communications systems, this system would also have more moving parts, which increases the likelihood of equipment failure or damage. Figure 2-6 shows a vehicle descending with active control, using navigation and steerable fins to keep the vehicle on course.

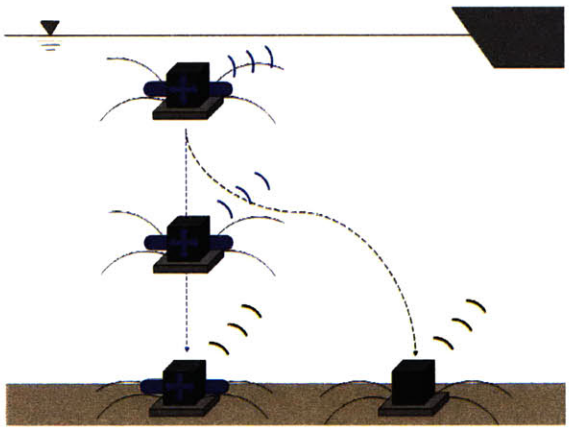

Figure 2-6: Tracking and Control During Descent

2.2.6 Retractable, Active Legs for Streamlining

One of the main concerns for underwater vehicles used in operations near the seafloor is the possibility of contact with the bottom. Many vehicles are equipped with bottom detecting sensors such as altimeters and Doppler velocity loggers (DVL) to avoid collisions with the bottom, which can cause equipment damage and potentially lodge

the vehicle against obstacles, preventing retrieval. For a vertical glider intended to deliver equipment to the seafloor, bottom contact is an necessary part of the mission, but damage and vehicle stability are still concerns. One possible solution to the issue of bottom collision is to include retractable legs to cushion the landing.

Retractable legs also provide the vehicle with a stable landing configuration, so that the vehicle can be maintained in an upright orientation. This is often crucial to the performance of the vehicle once it reaches the seafloor. For sensors that require ground contact, they can also provide a stable contact with the seafloor to take readings from. If the legs are maneuverable or retractable, they can aid in streamlining during the vehicle's descent, or even serve as control surfaces. However, the legs must be designed so that they do not affect the vehicle's stability during descent, or the vehicle may become uncontrollable. Figure 2-7 shows the descent of a vehicle with retractable legs. By keeping the legs folded during descent, the vehicle can be more streamlined. The legs would only for deploy for sensor or stability upon reaching the seafloor.

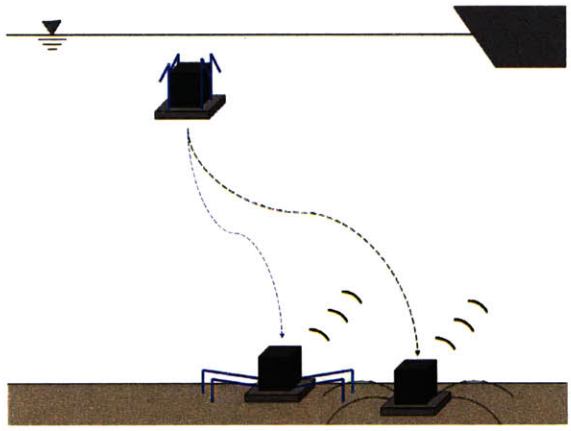

Figure 2-7: Retractable, Active Legs for Streamlining

2.2.7 Combination Solution: Streamlining, Tracking and Control

For a more complete solution to the problem of guiding a payload to the seafloor, several options can be combined into a single platform. The vehicle must be streamlined

to provide predictable and stable dynamics, have a navigation system that allows the vehicle to know where it is during the descent, and have control surfaces with actuation to allow it to correct course mid-flight if it should stray off path. This combination solution develops the vehicle into a form factor more similar to standard AUV's. The most common form of such a vehicle is one with a rounded or pointed nose, slender body, and control fins in the rear to control the pitch and yaw of the vehicle. In this case however, the vehicle is traveling a vertical path, instead of horizontal. This form factor could also include retractable legs to maintain the vehicle's upright position upon landing. Figure 2-8 shows the combination solution implemented on a vehicle, with active navigation, control, and streamlined surfaces.

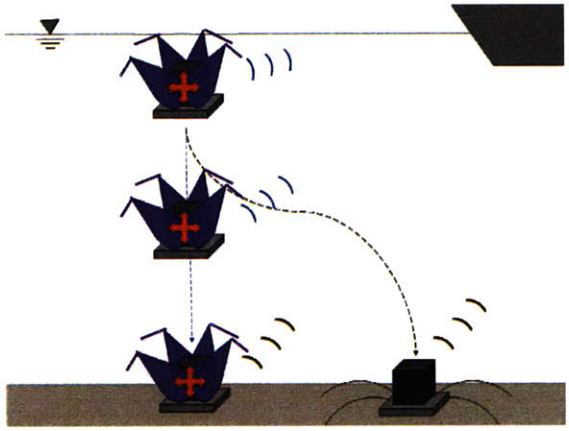

Figure 2-8: Combination Solution: Streamlining, Tracking and Control

2.3 Concept Selection

After developing several concepts that could be used to solve the problem of deep sea package delivery, we move to the task of choosing one to be taken to a prototype level for testing. Table 2.1 shows a concept comparison chart, which includes a description, required level of effort, and advantages and disadvantages to each proposed solution. The first three solutions, which includes tracking during descent, utilizing one-way travel times, and large simultaneous deployments, all employ an operational change to the deployment of the vehicle. They require either only software changes, or changes

in the way the vehicles are used, but require no extra mechanical design or moving parts. For these reasons, these solutions are all relatively simple to implement and are less costly than complete navigation and control systems.

Solutions 4a and 4b both alter the dynamics of the vehicle to improve placement accuracy. Current sensor platforms are designed for operating on the seafloor, but are not designed with the descent in mind, only the portion after landing on the seafloor. A heavier vehicle would enable the package to reach the seafloor in less time, and to reduce the effect of unpredictable cross-currents. However, this also raises the risk of equipment damage when the package impacts the seafloor. The implementation of a stable, streamlined body would enable the vehicle to descend in a more predictable manner.

Solution 5 is the objective studied in this thesis, and focuses on implementing a control and navigation system to effectively guide the vehicle, which we call the vertical glider. Solution 6 is an extension of this, and any full scale design would likely incorporate several of the features included in this method.

	Solution	**Pro**	**Con**	**Changes Required**	**Cost**
1	Tracking during descent (no control)	Up-to-date info on ocean currents	Need to turn on pinger; increased battery usage; requires monitoring by the ship	Operational, software	very low
2	Utilize one-way travel time	Fast survey	Need accurate clock	Software	low
3	Simultaneous Deployment	Increased precision through large number of deployed vehicles	Collision risk, fatigue, limited deck space	Operational	low
4a	Heavier to fall faster	Gets pushed off course less, reduces descent time	Failure from impact, deck safety, assembly	Mechanical	medium
4b	Streamlined to fall straighter	Less uncertainty of landing position, reduces descent time and σ_w	More difficult assembly	Mechanical	medium
5	Tracking & active control	Best way to get good positioning	More complex, most expensive, damage to moving parts	Modem, hardware, software, model of glide path	medium/ high
6	Retractable legs	Improves sensor contact, fall straighter, packability	Moving parts can break, complex, fragile	Mechanical	medium/ high

Table 2.1: Concept Selection Table

Chapter 3

Control Systems and Simulation

3.1 Overview

The field of glider vehicles has seen substantial work toward low power usage and the ability to conduct long term missions over several months. Vertical gliders, while not requiring the fine buoyancy control of horizontal gliders, still stand to benefit from an in-depth analysis of the limitations of navigation systems, vehicle dynamics, and various control strategies. While offering a low cost, low power solution for subsea equipment delivery, the vertical glider suffers from the fact that it is not a fully actuated vehicle. The descent speed is for the most part constant, and the vehicle is maneuvered only by pitch and yaw actuators. This leads to several challenges, most notably the problem of controlling a freely falling vehicle when acoustic navigation information is not always up to date or perfectly accurate. If the vehicle receives a poor navigation update, it would be unable to stop and take better measurements. In this chapter we seek to understand some of the performance characteristics and limitations of a vertical glider, which would aid operators in planning missions within the capabilities of the vehicle.

3.2 Noise Parameters

As with any navigation system, measurements taken by USBL incorporate a degree of uncertainty. The main limitations of USBL navigation are that both the update rate and lateral position error worsen with increasing distances between the surface vessel and the vehicle. These are expressed via the measurement noise, modeled to be Gaussian noise with distribution $N(0, \sigma_\alpha^2)$, and the position update rate. The update rate is defined as the time required for a signal to travel from the underwater vehicle to the surface and back (Δt_U), described below in Equation 3.1. In this equation, c is the speed of sound in seawater, while R is the distance from the vehicle to the surface, in this case synonymous with depth. In simulation, this is measured as the mission depth minus the height from bottom ($D - z$).

$$\begin{aligned} c &= 1500 \text{ m/s} \\ \Delta t_U &= \frac{2R}{c} = \frac{2(D-z)}{c} \end{aligned} \tag{3.1}$$

The second important source of uncertainty in the system is the presence of ocean currents. These cause a side-to-side motion of the vehicle which we have chosen to model by a Gaussian noise with distribution $N(0, \sigma_w^2)$. For the purposes of our simulations, we neglect vertical disturbances.

For our simulations, we will be implementing discrete time approximations of the vehicle's motion. While the measurements arrive at discrete time intervals, the process noise is a continuous phenomenon being translated into a discrete model. The behavior of continuous random variables in discrete time systems is slightly different than those of other system parameters. As noted in Equation 3.2, the process noise scales with $\sqrt{\Delta t}$, not just Δt. This was verified through literature [10], and a Monte Carlo simulation of the Euler approximations of a normally distributed random variable driving a first-order ordinary differential equation.

$$\begin{aligned} \dot{x} &= N(0, \sigma^2) \\ x_{k+1} &= x_k + N(0, \sigma_w^2)\sqrt{\Delta t} \end{aligned} \tag{3.2}$$

3.3 Vehicle Dynamics

The dynamics of underwater vehicles have been well documented, but there is still a need to accurately model the behavior of new designs to help in designing components, control systems, and mission plans. We have developed three distinct models for the vertical glider, each exploring gradually more complex behavior. The basic kinematic model treats the glider as a massless body, falling at a constant rate and able to respond instantly to control input. The extended kinematic model introduces a variable dive rate and maximum lateral velocity, to partially account for the lift and drag forces acting on the vehicle. Additionally, the extended model introduces a low-pass filter to mitigate noisy sensor measurements. Finally, the dynamic model incorporates the effects of gravity, buoyancy, fluid forces and inertia to depict a complete picture of the motion of the vehicle. While the fundamental equations for the dynamic model are presented here, the development of a three degree of freedom simulation is outside the scope of this thesis.

In Figure 3-1, the coordinate system of the vehicle is shown. The angles and dimensions are common to all three models, and are used in determining the position of the vehicle. Table 3.1 shows a description of each variable. In all three models, the parameter being measured by the vehicle is the angle α, which it seeks to minimize.

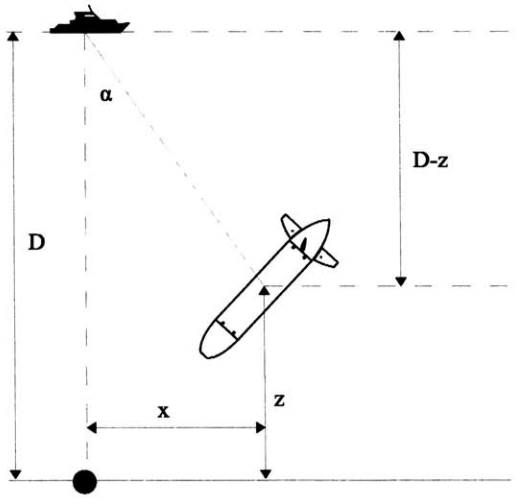

Figure 3-1: Vehicle Coordinate System

Variable	Note	Units
D	Mission depth	meters
D-z	Vehicle depth below surface	meters
x	Horizontal distance from target	meters
z	Vehicle height off bottom	meters
α	Angle from vertical under boat	radians

Table 3.1: Vehicle Coordinates

3.3.1 Basic Kinematic Model

The simplest model of motion for the vertical glider is based in kinematics. That is, it neglects the dynamic forces acting on the vehicle due to inertia , lift and drag. In this model the sideways motion of the vehicle is controlled directly: $\frac{dx}{dt}$ can change at anytime due to control input. The main benefit of this approach is greatly simplified equations of motion, which allows quicker computation time. We predict the kinematic model will yield results similar to the dynamic model. This is because the time constant associated with the turning motion of the vehicle is shorter that the average navigation update delay; the vehicle will have settled into a steady state by the time each new navigation hit arrives.

Equations 3.3, 3.4, 3.5, and 3.6 show the equations of motion for the basic kinematic model. Equation 3.3 describes the intended measurement α, which represents the angle from the surface vessel to the underwater vehicle. Equation 3.4 is the actual measurement recorded by the vehicle (α_m), a combination of α and the measurement noise. Equations 3.5 and 3.6 show the motion of the vehicle, which involves a constant descent rate and the horizontal motion due to control action. In Equation 3.6, K is the control gain.

$$\text{(true) } \alpha_t = \arctan\left(\frac{x}{D-z}\right) \tag{3.3}$$

$$\text{(measured) } \alpha_m = \alpha_t + N(0, \sigma_\alpha^2) \tag{3.4}$$

$$\dot{z} = \text{constant dive rate} \approx 1 \text{ m/s} \tag{3.5}$$

$$\dot{x} = -K\alpha_m \dot{z} + N(0, \sigma_w^2) \tag{3.6}$$

Equation 3.7 shows the discrete forms of the equations of motion, which are implemented in computer simulation. Note the dependence of process noise of $\sqrt{\Delta t}$.

$$\begin{aligned} \alpha_{k+1} &= \arctan\left(\frac{x_k}{D - z_k}\right) \\ \alpha_m &= \alpha_k + N(0, \sigma_\alpha^2) \\ z_{k+1} &= z_k + \dot{z}\Delta t \\ x_{k+1} &= x_k - K\alpha_m \dot{z}\Delta t + N(0, \sigma_w^2)\sqrt{\Delta t} \end{aligned} \tag{3.7}$$

3.3.2 Extended Kinematic Model

The extended kinematic model corrects several of the simplifications used in the basic kinematic model. The additions covered by the extended kinematic model include:

1. Variable dive rate: large lateral velocities reduce vertical velocity
2. Maximum lateral velocity ($\dot{X}_{max}$) enforced
3. Discrete low-pass filter to compensate for measurement noise

The basic kinematic model assumed that the vehicle would fall with a constant rate, and did not take into account the effects of drag or lift encountered by the vehicle as it changed direction or orientation. While a full analysis of the forces on the vehicle is outside the scope of the extended kinematic model, a simple relationship between lateral and vertical velocities can be implemented to represent these effects. Figure 3-2 shows this relationship, modeled by the quadratic function described in Equation 3.8, where $a = -2.4691$, $b = 2.716$, and $c = -0.2469$. This equation was derived by setting the parameters of $\dot{z}_{min} = 0.55$ m/s, $\dot{z}_{max} = 1$ m/s, and $\dot{X}_{max} = 0.5$ m/s.

$$\dot{z} = -\frac{b + \sqrt{b^2 - 4ac + 4a\dot{x}}}{2a} \tag{3.8}$$

The presence of noise in the measurement signal α_m can cause the vehicle to respond to faulty position information, potentially steering it off course. A discrete

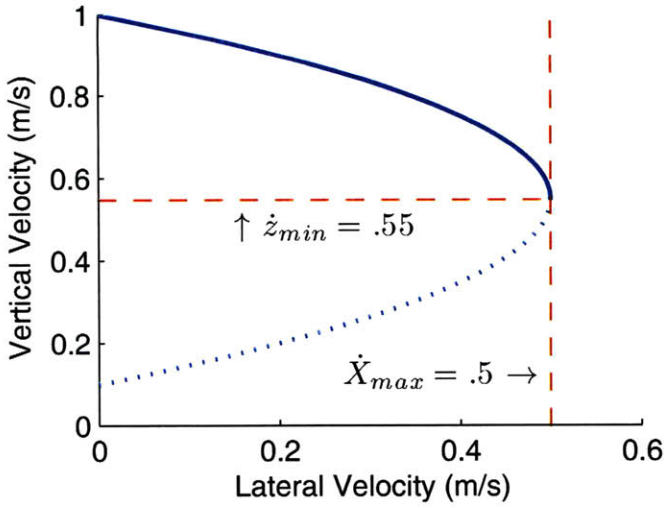

Figure 3-2: Relationship of dive rate and lateral velocity for extended kinematic model

low-pass filter can be used to attenuate the high frequency components associated with the Gaussian noise encountered in the measurement. The discrete low-pass filter creates each new measurement estimate $\tilde{\alpha}_{k+1}$ as a weighted average between the recorded measurement α_m and the previous estimate $\tilde{\alpha}_k$ (Equation 3.9). The filter gain K_f is tuned to provide a first-order time constant, determined by Equation 3.10. In our case, the filter time constant should be several times that of the measurement update interval.

$$\tilde{\alpha}_{k+1} = (1 - K_f)\tilde{\alpha}_k + K_f \alpha_k \tag{3.9}$$

$$\begin{aligned} \tau &= \Delta t \left(\frac{1 - K_f}{K_f} \right) \\ K_f &= \frac{\Delta t}{\tau + \Delta t} \end{aligned} \tag{3.10}$$

For full ocean depths of 4 km, the update interval would be about 5.33 seconds. For our filter design, we will use half of the mission depth of 2 km as our representative distance, which has an update time of 2.66 seconds. By selecting a time constant 10 times this value (26.6 seconds), we can ensure the filter records enough values to be effective. Using this and a simulation time step of 1 second, we can compute the

desired filter gain K_f to be 0.0362. Since the dive rate $\dot{z}$ varies during the mission,we have also used a reference value in computing the horizontal velocity in Equation 3.14, setting $\dot{z}_{ref} = 1$ m/s. Combining these additional behaviors, we get a new model of vehicle motion:

$$\text{(true) } \alpha_t = \arctan\left(\frac{x}{D-z}\right) \tag{3.11}$$

$$\text{(measured) } \alpha_m = \alpha_t + N(0, \sigma_\alpha^2) \tag{3.12}$$

$$\text{(estimate) } \tilde{\alpha} = \left(\frac{1}{1+s\tau}\right)\alpha_m \tag{3.13}$$

$$\dot{x} = -K\tilde{\alpha}\dot{z}_{ref} + N(0, \sigma_w^2), \dot{x} \leq \dot{X}_{max} \tag{3.14}$$

$$\dot{z} = f(\dot{x}) \tag{3.15}$$

The discrete versions of these equations are implemented in computer simulation, as shown in Equation 3.16.

$$\begin{aligned}
\alpha_{k+1} &= \arctan\left(\frac{x_k}{D-z_k}\right) \\
\alpha_m &= \alpha_k + N(0, \sigma_\alpha^2) \\
\tilde{\alpha}_{k+1} &= (1-K_f)\tilde{\alpha}_k + K_f\alpha_m \\
x_{k+1} &= x_k - K\tilde{\alpha}_k\dot{z}\Delta t + N(0, \sigma_w^2)\sqrt{\Delta t} \\
z_{k+1} &= z_k + f(\dot{x})\Delta t
\end{aligned} \tag{3.16}$$

3.3.3 Dynamic Model

A further extension of the basic and extended kinematic models is the dynamic model, which takes into account the gravitational, inertial and fluid forces acting upon the vehicle to determine its motion through the water. As shown in Figure 3-3, the motion of the vehicle is a balance of forces between the downward force of gravity, located at the vehicle's center of mass, and the upward force of buoyancy, located at the vehicle's center of buoyancy. The center of buoyancy is in the same location as the vehicle's center of mass for an equivalent volume of water, which is approximately the

geometric center. These forces are coupled with the lift and drag forces encountered by the vehicle body and rudder fins, as well as the Munk moment (M_m,M_{Rm}), a torque experienced by streamlined bodies when moving through the water at a non-zero angle of attack. The control action in this model is the angle of deflection of the rudder fins δ, which causes a lift force on the fin perpendicular to to the direction of travel (F_{RL}). This force causes the vehicle to rotate, and the resulting lift F_L on the body enables horizontal motion.

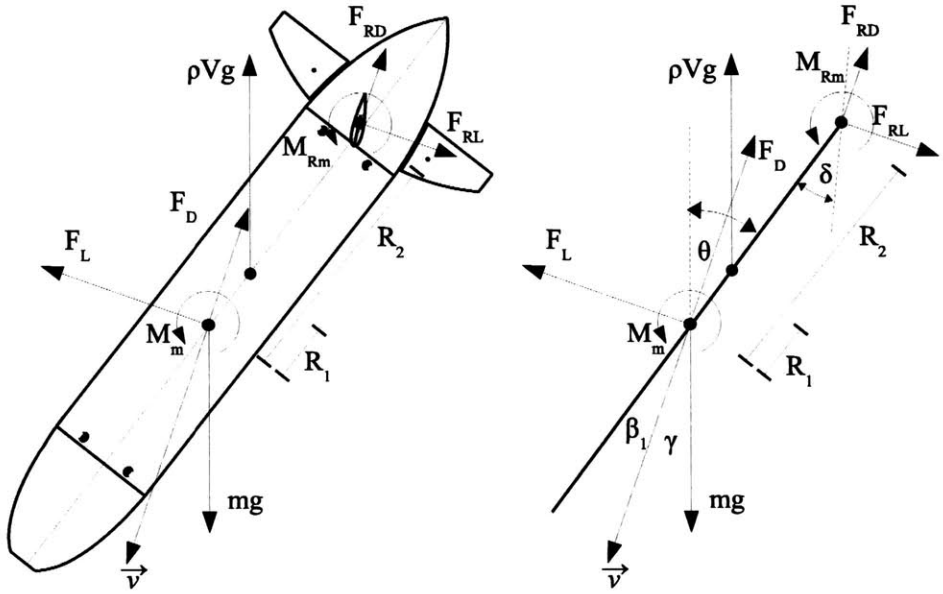

Figure 3-3: Vertical AUV Force Balance

The equations of motion for this model are derived by evaluating the forces in the horizontal and vertical directions, as well as the moments on the vehicle. This force balance gives us differential equations for the components of acceleration $\frac{d^2x}{dt^2}$, $\frac{d^2z}{dt^2}$, and $\frac{d^2\theta}{dt^2}$. The lift force on the body or rudder is always perpendicular to the direction of travel, denoted here by the velocity's angle from the vertical γ. The drag force is always parallel and opposite to the direction of travel. These forces are further broken down to their components in the xy plane, denoted by the sums F_x and F_y in

Variable	Note	Value/Units
ρ	Density of water	1000 kg/m^3
V	Volume of vehicle	.007382 m^3
m	Mass of vehicle	7.234 kg
g	Accel. of gravity	9.8 m/s
R_1	Distance from C_m to C_b	0.05 m
R_2	Distance from C_m to rudder	0.25 m
I	Moment of Inertia	0.23 kg·m^2
F_D	Body drag force	N
F_L	Body lift force	N
F_{RD}	Rudder drag force	N
F_{RL}	Rudder lift force	N
M_m	Munk moment on body	N·m
M_{Rm}	Munk moment on rudder	N·m
$\vec{v}$	Velocity vector	m/s

Table 3.2: Dynamic Model Parameters

Equations 3.17 and 3.18.

$$\begin{aligned}\sum F_x = m\frac{d^2x}{dt^2} &= (F_D + F_L + F_{RD} + F_{RL}) \cdot \hat{x} \\ &= F_D \sin\gamma + F_L \cos\gamma + F_{RD} \sin\gamma + F_{RL} \cos\gamma\end{aligned} \tag{3.17}$$

$$\begin{aligned}\sum F_y = m\frac{d^2y}{dt^2} &= -mg + \rho V g + (F_D + F_L + F_{RD} + F_{RL}) \cdot \hat{y} \\ &= -mg + \rho V g + F_D \cos\gamma + F_L \sin\gamma + F_{RD} \cos\gamma + F_{RL} \sin\gamma\end{aligned} \tag{3.18}$$

$$\begin{aligned}\tau = I\frac{d^2\theta}{dt^2} &= \sum r \times F \\ I\ddot{\theta} &= \rho V g \times R_1 + F_{RD} \times R_2 + F_{RL} \times R_2 + M_m + M_{Rm} \\ \ddot{\theta} &= \frac{1}{I}\left(\rho V g \times R_1 + F_{RD} \times R_2 + F_{RL} \times R_2 + M_m + M_{Rm}\right)\end{aligned} \tag{3.19}$$

The lift and drag forces are dependent on several parameters, including fluid density, fluid velocity, a reference area such as cross-section or planform, and the lift or drag coefficient. The coefficients of lift and drag for solid bodies at non-zero angles

of attack cannot usually be described by a single equation, they must be observed through experimentation, although are approximately linear below stall angles. The relationships for lift and drag force, as well as the Munk moment, are described in Equation 3.20.

$$\begin{aligned} F_D &= -\tfrac{1}{2}\rho A_o v^2 C_D \\ F_L &= \tfrac{1}{2}\rho A_p v^2 C_L \\ M_m &= -\tfrac{1}{2}\left(A_{zz} - A_{xx}\right) v^2 \sin 2\alpha \end{aligned} \tag{3.20}$$

A vehicle described by this dynamic model would require a more complex control system than the one presented for the basic kinematic models. One likely candidate for this would be a state space controller combined with a Kalman filter [25] for position estimation, because our system is already described as a state vector, and state space models are useful for systems with nonlinear behavior. In practice, the vehicle position estimation is accomplished by the hardware and software built into the USBL navigation system, but the equations presented here are instructive for exploring the effects of noise on system performance. For the purposes of this thesis, the dynamic model will not be pursued further beyond this point.

3.4 Dimensional Analysis

Dimensional analysis is an important tool in developing models of dynamic systems, especially when small prototypes are used to explore the behavior of large systems. By transforming system parameters into their non-dimensional counterparts, inferences can be made about system behavior that hold true regardless of the scale of the mission. In the case of the vertical glider, the main metric of system performance is the standard deviation of the landing error σ_E, which can be measured in meters. However, it remains unclear what other parameters this depends on, and whether our notion of performance has the same meaning at different vehicle and mission scales. For example, is a landing error of 10 meters for a mission depth of 20 meters as

meaningful as one with a mission depth of 2000 meters?

In the following sections we will discuss the process by which we develop dimensionless parameters and analyze their effects on system performance. The first step in dimensional analysis is identifying the variables of the system, and deciding which are relevant in determining system performance.

3.4.1 Basic Kinematic Model Variables

The performance of the basic kinematic model as described in Equation 3.7 depends on 6 variables, as shown in Equation 3.21. The effects of some variables are fairly intuitive, such as increasing measurement and process noise causing degraded performance. The effects of other variables, such as control gain K, dive rate, and update rate are less clear. Here we seek to develop dimensionless parameters that represent vehicle performance. The relevant parameters of the basic kinematic model that may have an effect on vehicle performance are defined in Table 3.3.

Our metric of vehicle performance, the standard deviation of landing error σ_E, is expressed in units of meters. To create a dimensionless parameter, this must be divided by a parameters or parameters having units of meters as well.

$$\sigma_E = f\left(K, \sigma_\alpha, \sigma_w, \dot{z}, D, c, \Delta t_u\right) \tag{3.21}$$

Variable	Description	Units
σ_E	Std. Dev. of Landing Error	m
K	Control Gain	1/rad
σ_α	Measurement Noise	rad
σ_w	Process Noise	m/s
$\dot{z}$	Dive Rate	m/s
D	Mission Depth	m
c	Speed of Sound in water	m/s
Δt_u	Position update delay	s

Table 3.3: Relevant Variables for Dimensional Analysis, Basic Kinematic Model

The metric of system performance σ_E can be divided by the mission depth to yield a dimensionless quantity $\frac{\sigma_E}{D}$. Additionally, the process noise σ_w is in units of m/s,

and can be divded by the dive rate to yield a dimensionless quantity $\frac{\sigma_w}{\dot{z}}$. The values of measurement noise σ_α and control gain K are in units of radians and 1/radians. The radian is itself a dimensionless quantity, so these two parameters can remain as they are.

The variable of position update delay (Δt_u) determines how often the vehicle receives new information about its location. However, this is not a single value; rather, it is the collection of all update delays throughout the course of the mission $\{\Delta t_1, \Delta t_2, \Delta t_3, ..., \Delta t_n\}$. To effectively implement this parameter in our simulations, we must derive the dependency of Δt_u as a function of other parameters.

We start with the equation for the update delay as a function of depth. The time it takes to send a signal to the surface and receive a response is the current depth $(D - z_i)$ plus the depth when the response arrives $(D - z_{i+1})$, divided by the speed of sound c.

$$\Delta t_i = \frac{(D - z_i) + (D - z_{i+1})}{c} = \frac{2D - z_i - z_{i+1}}{c} \tag{3.22}$$

Additionally, this holds true for future times, Δt_{i+1}:

$$\Delta t_{i+1} = \frac{(D - z_{i+1}) + (D - z_{i+2})}{c} = \frac{2D - z_{i+1} - z_{i+2}}{c} \tag{3.23}$$

The depth at the next timestep will be the current depth plus its rate of change (dive rate) times the time interval:

$$z_{i+1} = z_i + \dot{z}\Delta t_i \tag{3.24}$$

By plugging the value of Δt_i from Equation 3.22 into Equation 3.24, we get:

$$z_{i+1} = z_i + \dot{z}\left(\frac{2D - z_i - z_{i+1}}{c}\right) \tag{3.25}$$

We now simplify the equation and combine terms having z_i and z_{i+1}, solving for

z_{i+1}:

$$\begin{aligned} cz_{i+1} &= cz_i + 2\dot{z}D - \dot{z}z_i - \dot{z}z_{i+1} \\ (c+\dot{z})z_{i+1} &= (c-\dot{z})z_i + 2\dot{z}D \\ z_{i+1} &= \left(\frac{c-\dot{z}}{c+\dot{z}}\right) z_i + \frac{2\dot{z}D}{c+\dot{z}} \end{aligned} \tag{3.26}$$

The values above also hold true for z_{i+1} and z_{i+2}, as shown below:

$$z_{i+2} = \left(\frac{c-\dot{z}}{c+\dot{z}}\right) z_{i+1} + \frac{2\dot{z}D}{c+\dot{z}} \tag{3.27}$$

We now substitute the values of z_{i+1} and z_{i+2} from equations 3.26 and 3.27 into equation 3.23:

$$\begin{aligned} \Delta l_{i+1} &= \frac{1}{c}\left(2D - \left(\frac{c-\dot{z}}{c+\dot{z}}\right) z_i - \frac{2\dot{z}D}{c+\dot{z}} - \left(\frac{c-\dot{z}}{c+\dot{z}}\right) z_{i+1} - \frac{2\dot{z}D}{c+\dot{z}}\right) \\ &= \frac{2D}{c} - \frac{1}{c}\left(\left(\frac{c-\dot{z}}{c+\dot{z}}\right) z_i + \frac{2\dot{z}D}{c+\dot{z}} + \left(\frac{c-\dot{z}}{c+\dot{z}}\right) z_{i+1} + \frac{2\dot{z}D}{c+\dot{z}}\right) \\ &= \frac{2D}{c} - \frac{1}{c}\left(\left(\frac{c-\dot{z}}{c+\dot{z}}\right)(z_i + z_{i+1}) + \frac{4\dot{z}D}{c+\dot{z}}\right) \\ &= \frac{2D}{c} + \frac{1}{c}\left(\left(\frac{c-\dot{z}}{c+\dot{z}}\right)(2D - z_i - z_{i+1} - 2D) - \frac{4\dot{z}D}{c+\dot{z}}\right) \\ &= \frac{2D}{c} + \left(\frac{c-\dot{z}}{c+\dot{z}}\right)\left(\frac{2D - z_i - z_{i+1}}{c}\right) - \frac{1}{c}\left(\frac{c-\dot{z}}{c+\dot{z}}\right)(2D) - \frac{1}{c}\left(\frac{4\dot{z}D}{c+\dot{z}}\right) \end{aligned} \tag{3.28}$$

We have now isolated a term matching the value of Δt_i from Equation 3.22, and can substitute that into our equation. This allows us to express Δt recursively, solely

in terms of its previous value and constant parameters such as D, $\dot{z}$, and c.

$$
\begin{aligned}
&= \frac{2D}{c} + \left(\frac{c-\dot{z}}{c+\dot{z}}\right)\Delta t_i - \frac{1}{c}\left(\frac{c-\dot{z}}{c+\dot{z}}\right)(2D) - \frac{1}{c}\left(\frac{4\dot{z}D}{c+\dot{z}}\right) \\
&= \frac{2D}{c} + \left(\frac{c-\dot{z}}{c+\dot{z}}\right)\Delta t_i - \frac{2D(c-\dot{z}) + 4\dot{z}D}{c(c+\dot{z})} \\
&= \frac{2D}{c} + \left(\frac{c-\dot{z}}{c+\dot{z}}\right)\Delta t_i - \frac{2cD - 2\dot{z}D + 4\dot{z}D}{c(c+\dot{z})} \\
&= \frac{2D}{c} + \left(\frac{c-\dot{z}}{c+\dot{z}}\right)\Delta t_i - \frac{2cD + 2\dot{z}D}{c(c+\dot{z}}) \\
&= \frac{2D}{c} + \left(\frac{c-\dot{z}}{c+\dot{z}}\right)\Delta t_i - \frac{2D\cancel{(c+\dot{z})}}{c\cancel{(c+\dot{z})}} \\
&= \frac{\cancel{2D}}{\cancel{c}} + \left(\frac{c-\dot{z}}{c+\dot{z}}\right)\Delta t_i - \frac{\cancel{2D}}{\cancel{c}}
\end{aligned}
\tag{3.29}
$$

A good bit of algebra later, we arrive at an equation for Δt_{i+1}, which depends only on the previous update delay Δt_i, the dive rate $\dot{z}$, and the speed of sound c.

$$\Delta t_{i+1} = \left(\frac{c-\dot{z}}{c+\dot{z}}\right)\Delta t_i \tag{3.30}$$

By dividing through by c, we can put the equation in terms of a dimensionless pararameter $\frac{\dot{z}}{c}$.

$$\Delta t_{i+1} = \left(\frac{1-\frac{\dot{z}}{c}}{1+\frac{\dot{z}}{c}}\right)\Delta t_i \tag{3.31}$$

We now have our four dimensionless parameters:

$$\frac{\sigma_E}{D} = f\left(K, \sigma_\alpha, \frac{\sigma_w}{\dot{z}}, \frac{\dot{z}}{c}\right) \tag{3.32}$$

In the following sections we will be using these parameters to look at their effects on vehicle performance.

3.4.2 Extended Kinematic Model Variables

For the extended kinematic model, our equations of motion incorporate several more variables than the basic model, specifically the parameters used to define relationship

between dive rate $\dot{z}$ and horizontal velocity $\dot{x}$. Since the dive rate $\dot{z}$ is no longer constant throughout a mission, it can no longer be applied in the dimensional analysis. However, for the purposes of defining the velocity relationship, we have chosen a reference velocity $\dot{z}_{ref} = 1$ m/s, which is equal to the dive rate when the vehicle is travelling straight down (zero horizontal velocity). This value is equal to the maximum dive rate $\dot{z}_{max}$. The relevant parameters for the extended kinematic model are shown in Equation 3.33 and Table 3.4.

$$\sigma_E = f\left(K, \sigma_\alpha, \sigma_w, \dot{z}_{ref}, D, c, \Delta t_u, \dot{X}_{max}, \dot{z}_{min}, \dot{z}_{max}\right) \tag{3.33}$$

Variable	Description	Units
σ_E	Std. Dev. of Landing Error	m
K	Control Gain	1/rad
σ_α	Measurement Noise	rad
σ_w	Process Noise	m/s
D	Mission Depth	m
c	Speed of Sound in water	m/s
Δt_u	Position update delay	s
$\dot{X}_{max}$	Max Horizontal Velocity	m/s
$\dot{z}_{min}$	Min Vertical Velocity	m/s
$\dot{z}_{max}$	Max Vertical Velocity	m/s

Table 3.4: Relevant Variables for Dimensional Analysis, Extended Kinematic Model

The introduction of several new parameters significantly complicates the task of representing changes in vehicle performance in a meaningful way. The basic kinematic model depended on four input parameters, which can be shown in several groups of contour plots. However, systems with more parameters than this become increasingly difficult to visualize. For this reason we have chosen to fix several of the parameters, to make the dimensional analysis easier to visualize. The variables that define the relationship between horizontal and vertical speed are fixed as shown by Equation 3.8 and Figure 3-2. Once these parameters have been fixed, we are left with three input parameters, as shown in Equation 3.34. This analysis can be represented by a single group of contour plots. We are left with four dimensionless parameters, including the system performance and three parameter to vary during the simulation. These

parameters are expressed in Equation 3.34.

$$\frac{\sigma_E}{D} = f\left(K, \sigma_\alpha, \frac{\sigma_w}{\dot{z}_{ref}}\right) \tag{3.34}$$

3.5 Simulation Setup

The application of the models discussed in Section 3.3 is to perform simulations in MATLAB to estimate the performance of the system with varying input parameters. Each simulation begins with the vehicle placed at the water surface, directly above the target ($x = 0, z = D$), as it would in real world tests. The water depth in these scenarios is 4000 m. The simulated vehicle descends with dive rate $\dot{z}$ and takes measurements of its position at appropriate time steps, according to the update delay described earlier. While the rate of position updates varies with depth, the simulation still computes the actual position of the vehicle at a constant time step, in this case 1 second. For times when the measurement is not updated, the simulation keeps the last current measurement. The main metric of performance for the vertical glider is the standard deviation of the landing error, σ_E. For the basic kinematic model, the varied parameters are the measurement noise σ_α (measured in radians), the process noise σ_w (measured in meters/second), the control gain K (1/radians), and the dive rate $\dot{z}$. The extended kinematic model does not vary the dive rate.

To show the effects on system performance by the variations in multiple parameters, we have broken the results analysis into two sections. First, we look at the effects on performance by varying a single parameter, while keeping others constant, and looking at plots of the vehicle's path during the mission. This allows us to visualize the path the vehicle takes, and observe any interesting behavior. Secondly, we will look at the non-dimensional case, varying all parameters simultaneously, and observing the resulting performance through a colored contour plot. This allows us to visualize performance on a broader scale.

Sample code from the MATLAB simulations discussed here can be found in Appendix B.2.

3.6 Basic Kinematic Model Simulation Results

The basic kinematic model simulations vary each input parameters between 4 different values: $\sigma_w = [0.1, .15, .2, .25]$ m/s, $\sigma_\alpha = [0.4, 0.8, 1.2, 1.6]$ degrees, $K = [2, 4, 6, 8]$ 1/rad. While each parameter is varied, the other two are held constant, to give a clear picture of that parameter's effect on system performance. Figures 3-4, 3-6, and 3-7 show the results of 10 sample trials from each scenario. Table 3.5 shows a sample of vehicle performance from the basic kinematic model with a control gain of $K = 1$, varying both σ_α and σ_w. Tables of additional results with higher gains can be found in Appendix A.1.1.

Each grouping of plots shows us the unique effects that each varying parameter has on the system. In Figure 3-4, we vary the measurement noise σ_α. Here we observe that increasing σ_α causes more erratic side to side motion, which can translate into increased errors as depth increases. Figure 3-5 shows the histograms of landing error from 10^4 trials of the scenario where σ_α is varied. The main observation of this plot is that while the standard deviation of landing error changes when we change parameters, the distribution itself is still approximation Gaussian.

In Figure 3-6, we vary process noise σ_w between four values, and can observe that increasing process noise causes shifts in velocity to gradually push the vehicle off course. Since the plots shown have a control gain of $K = 1$, the ability of the vehicle to counteract this effect is limited. As shown in the contour plots later, higher control gains enable the vehicle to better resist side to side motion from process noise.

Figure 3-7, we vary the control gain K. Here we can observe that the cases with high gains can cause large side to side motion early in the mission. This is because at shallow depths, a given horizontal position error produces a higher angular measurement, and thus a more pronounced side to side motion. We can see that as the vehicle descends, this erratic motion continues to cause high horizontal velocities, but as the vehicle is able to quickly overcome errors, the landing accuracy is not severely affected.

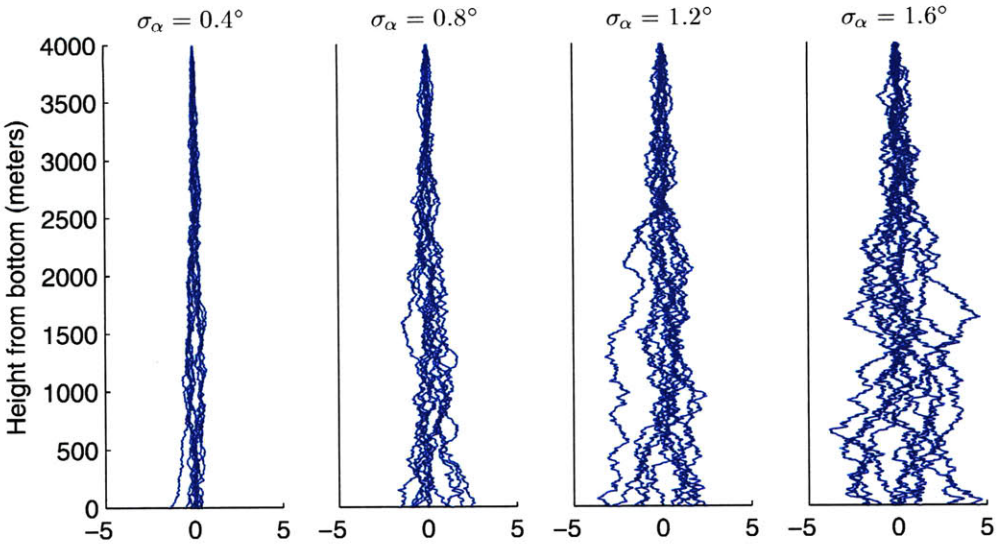

Figure 3-4: Basic Kinematic Model, Plot with varying σ_α while $\sigma_w = 0$, constant gain $K = 1$, 10 trials

	$\sigma_w = 0.00$	$\sigma_w = 0.10$	$\sigma_w = 0.15$	$\sigma_w = 0.20$	$\sigma_w = 0.25$
$\sigma_\alpha = 0.0°$	0.00	3.94	4.70	7.48	9.49
$\sigma_\alpha = 0.4°$	0.55	3.30	5.20	6.76	9.88
$\sigma_\alpha = 0.8°$	1.11	3.77	5.24	7.28	9.07
$\sigma_\alpha = 1.2°$	1.86	3.70	5.66	6.91	10.40
$\sigma_\alpha = 1.6°$	2.24	4.45	5.46	7.53	8.59

Table 3.5: Basic Kinematic Model Vehicle Performance, K = 1, values correspond to standard deviation of landing error over 100 trials, units of meters

3.6.1 Dimensional Analysis Results

In addition to simulations showing the path of the vehicle through the mission while varying a single parameter, it is also helpful to perform large scale simulations showing the relationship of vehicle performance to variations in multiple parameters. Here we have used the non-dimensional forms of our system parameters, $\frac{\sigma_w}{\dot{z}}$, σ_α, K, and $\frac{\dot{z}}{c}$, and the non-dimensional system performance, $\frac{\sigma_E}{D}$. Figures 3-8, 3-9, and 3-10 each show multiple contour plots showing the resulting system performance from varying all four non-dimensional parameters simultaneously. For these simulations, we vary σ_w at 20 steps between 0 and 0.25 m/s, while σ_α is varied between 0 and 1.6 degrees. In each contour plot, $\frac{\sigma_w}{\dot{z}}$ is shown on the X axis and σ_α is on the Y axis. Each plot in the group represents a different value of the control gain K, and each group of

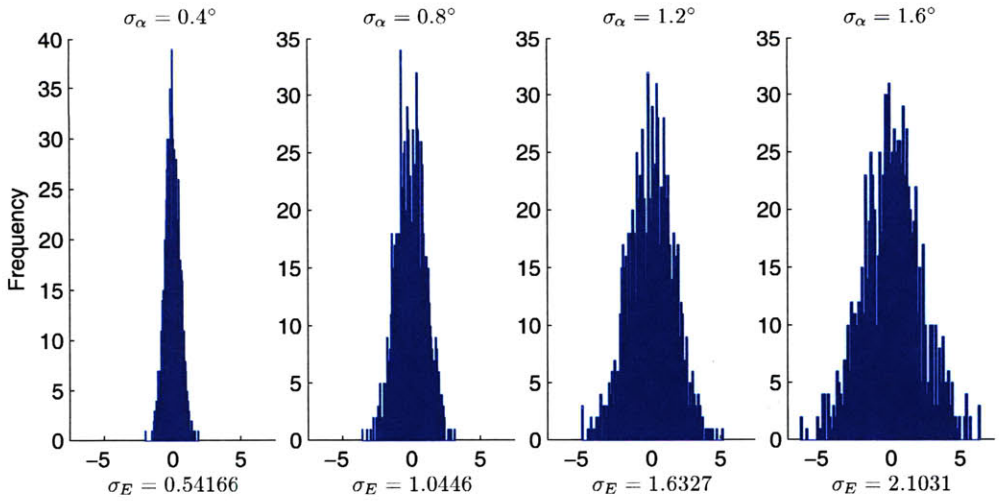

Figure 3-5: Histogram of varying σ_α while $\sigma_w = 0$, constant gain $K = 1$, 10^4 trials

plots is a different value of $\frac{\dot{z}}{c}$. The colors in each plot are the system performance parameter, $\frac{\sigma_E}{D}$, with blue representing low values and red representing high values. To better express the variations in the parameters, $\frac{\sigma_w}{\dot{z}}$, σ_α, and the performance $\frac{\sigma_E}{D}$ are plotted on a $\log_{10}$ scale.

The results shown by this simulation indicate that both σ_w and σ_α have an adverse effect on vehicle performance, although σ_w is dominant unless the system is operating with a high control gain, which would amplify errors in the measurements. At a dive rate of 0.5 m/s and values of σ_α below $10^{-2.5}$ radians (0.18 degrees), the the process noise σ_w will dominate at any value of K. Thus, desiging a system with values of σ_α below $10^{-2.5}$ would be a poor use of resources, because at this point better control will do nothing to improve vehicle performance. However, as shown in Figures 3-9 and 3-10, increasing the dive rate to 1 or 2 m/s reduces the effects of σ_w and can make improvements in σ_α more effective.

3.7 Extended Kinematic Model Simulation Results

Similar to the basic kinematic model, the extended kinematic model simulations vary each input parameters between 4 different values: $\sigma_w = [0.1, .15, .2, .25]$, $\sigma_\alpha = [0.4, 0.8, 1.2, 1.6]$, $K = [2, 4, 6, 8]$. While each parameter is varied, the other two are

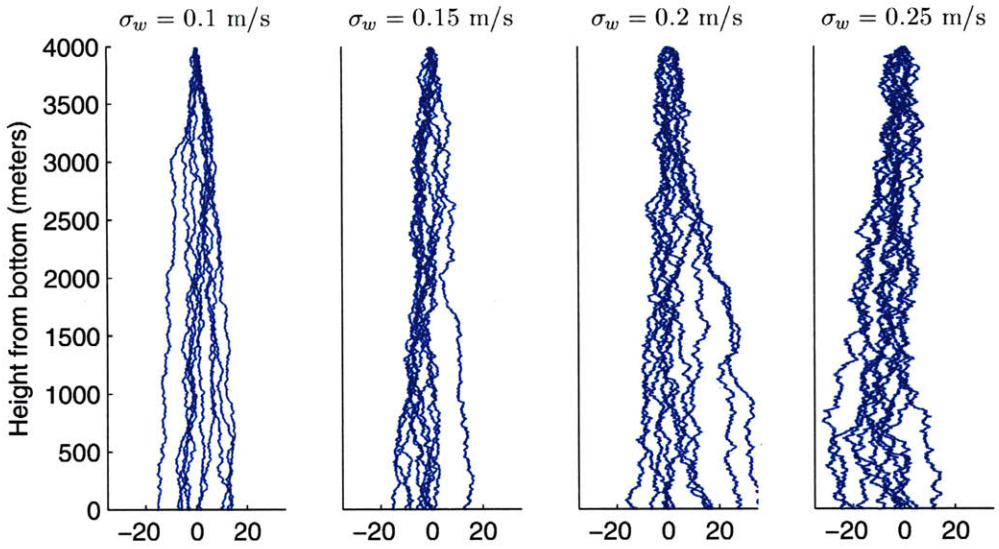

Figure 3-6: Basic Kinematic Model, Plot with varying σ_w while $\sigma_\alpha = 0$, constant gain $K = 1$, 10 trials

held constant, to give a clear picture of that parameter's effect on system performance. Figures 3-11, 3-12, and 3-13 show the results of 10 sample trials from each scenario. We observe from each of these scenarios that while vehicle performance is not drastically different than the basic kinematic model, the motion of the vehicle takes a smoother path due to the low-pass filter implemented on the measurements. In the case of varying the control gain K, the filter also helps to mitigate the erratic behavior caused by measurement errors at very shallow depths. Table 3.6 shows a sample of vehicle performance from the extended kinematic model with a control gain of $K = 1$, varying both σ_α and σ_w. Tables of a additional results with higher gains can be found in Appendix A.1.2.

	$\sigma_w = 0.00$	$\sigma_w = 0.10$	$\sigma_w = 0.15$	$\sigma_w = 0.20$	$\sigma_w = 0.25$
$\sigma_\alpha = 0.0°$	0.00	0.52	1.01	1.54	2.15
$\sigma_\alpha = 0.4°$	3.46	4.08	3.88	3.99	4.19
$\sigma_\alpha = 0.8°$	5.48	5.77	5.85	6.47	6.47
$\sigma_\alpha = 1.2°$	6.18	6.36	7.06	6.83	8.31
$\sigma_\alpha = 1.6°$	10.62	8.44	10.00	10.19	9.28

Table 3.6: Extended Kinematic Model Vehicle Performance, K = 1, values correspond to standard deviation of landing error over 100 trials, units of meters

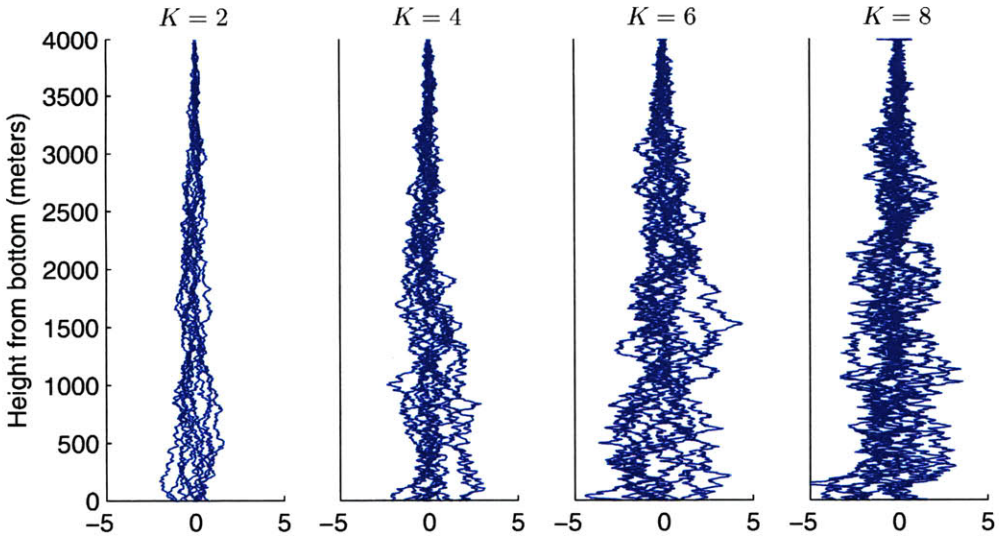

Figure 3-7: Basic Kinematic Model, Plot with varying K while $\sigma_w = 0$, $\sigma_\alpha = 0.4°$, 10 trials

The benefits of the low-pass filter implemented on the extended kinematic model are not immediately apparent from the plots of the vehicle path, other than a smoother path and slightly improved performance. In Figures 3-14 and 3-15, we can observe the vertical and horizontal velocity of a single trial, simulated with the low-pass filter turned on with a time constant $\tau = 26.6$ seconds, and with the filter turned off. Here we observe a much more stable side-to-side motion, and because of their coupled nature, vertical velocity as well. In real world testing, the vehicle would be unable to change velocity as quickly as described by the kinematic model, but filtered measurements would help to reduce wear on the moving parts of the underwater vehicle, preventing it from trying to respond to inaccurate measurements.

3.7.1 Dimensional Analysis Results

The extended kinematic model dimensional analysis results follow suit with the basic kinematic model, varying the parameters $\frac{\sigma_w}{\dot{z}}$, σ_α, and K to observe resulting changes in system performance $\frac{\sigma_E}{D}$. However, in this case because the dive rate is constantly changing throughout the mission, $\frac{\dot{z}}{c}$ is not a parameter that can be fixed. We have chosen to define the relationship between $\dot{x}$ and $\dot{z}$ ahead of time, so the dimensional

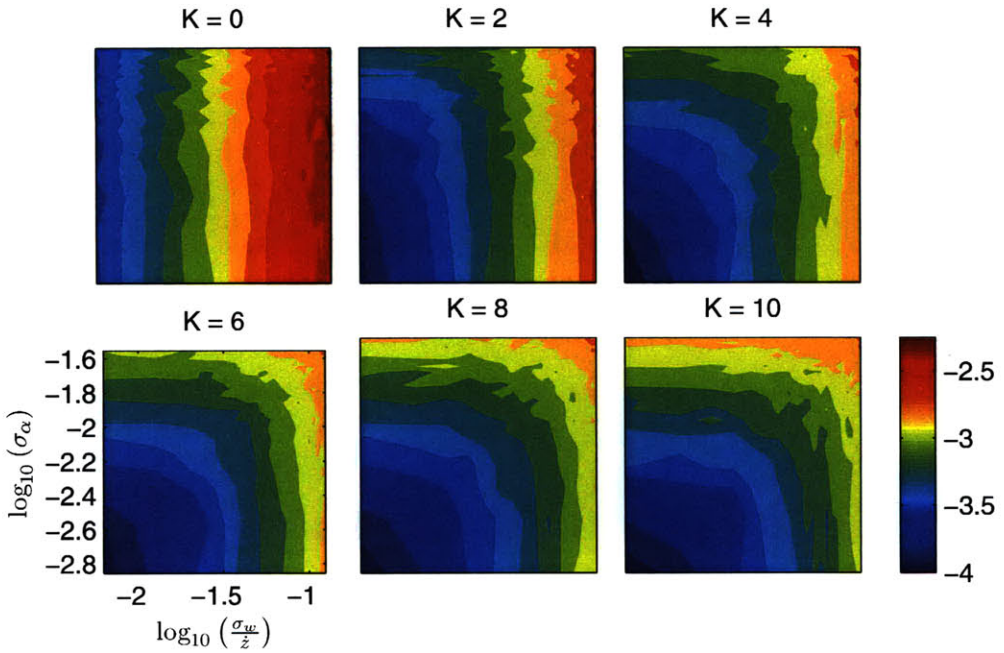

Figure 3-8: Basic Kinematic Model, Non-Dimensional Multiple Contour Plot, $\dot{z} = 0.5$ m/s; Color bar represents non-dimensional landing error $\log_{10}\left(\frac{\sigma_E}{D}\right)$

analysis will consist of only one group of plots, shown in Figure 3-16.

The contour plot for the extended model, Figure 3-16, shows results very similar to the basic kinematic model, with performance on par with the basic model set at a dive rate of 1 m/s, shown in Figure 3-9. The low-pass filter in this case has reduced erratic side-to-side motion to keep the dive rate very close to $\dot{z}_{ref} = 1$ m/s.

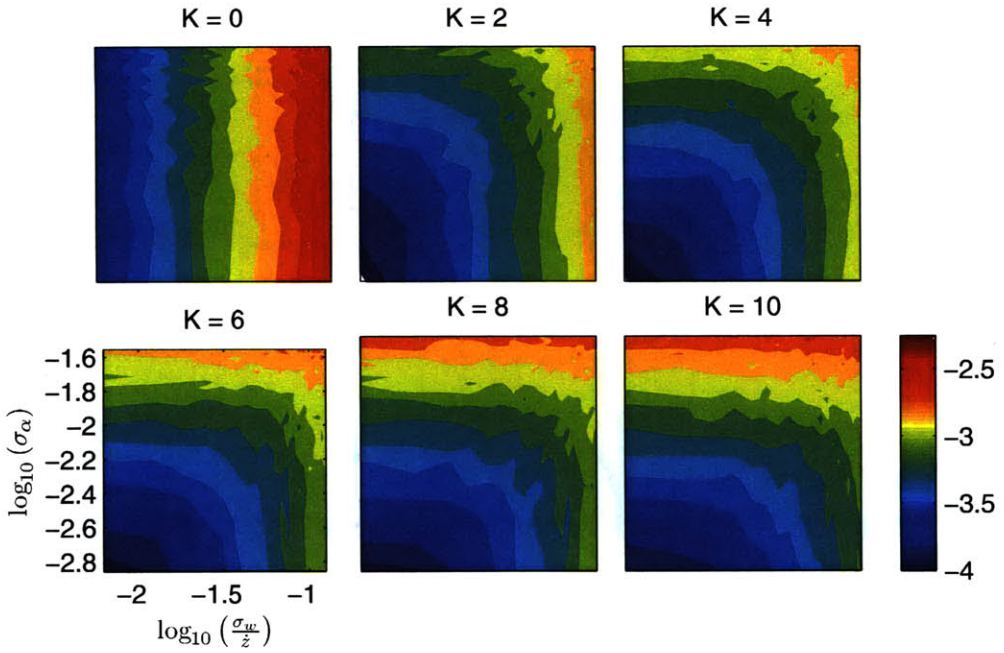

Figure 3-9: Basic Kinematic Model, Non-Dimensional Multiple Contour Plot, $\dot{z} = 1$ m/s; Color bar represents non-dimensional landing error $\log_{10}\left(\frac{\sigma_E}{D}\right)$

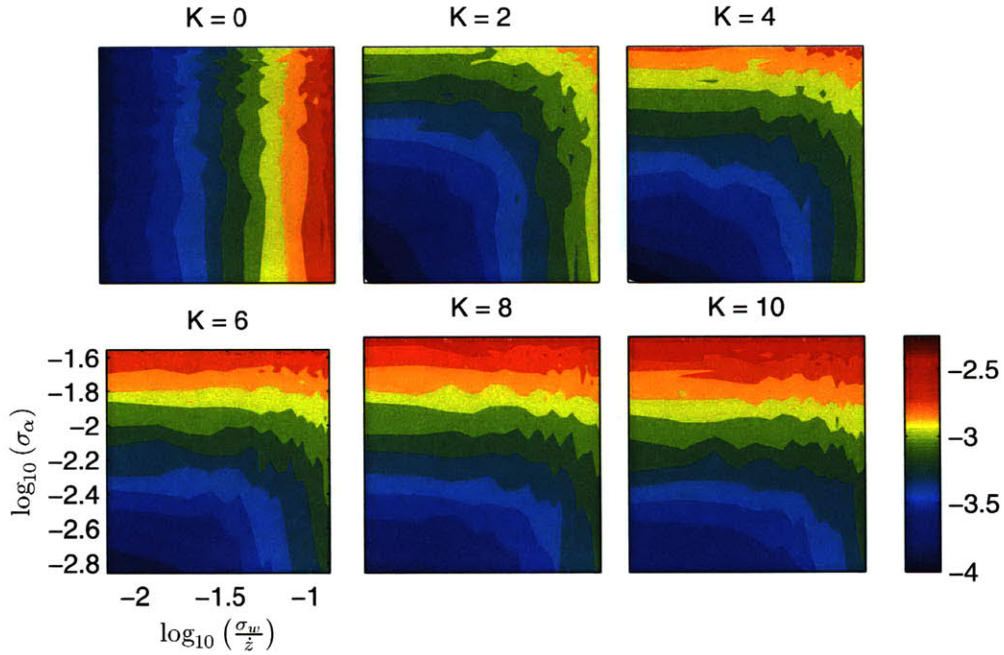

Figure 3-10: Basic Kinematic Model, Non-Dimensional Multiple Contour Plot, $\dot{z} = 2$ m/s; Color bar represents non-dimensional landing error $\log_{10}\left(\frac{\sigma_E}{D}\right)$

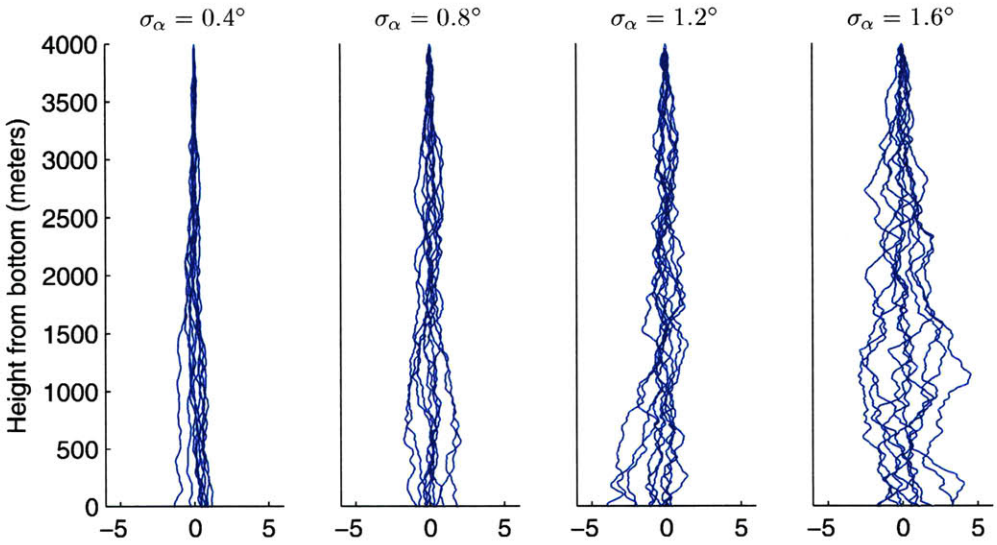

Figure 3-11: Extended Kinematic Model, Plot with varying σ_α while $\sigma_w = 0$, constant gain $K = 1$

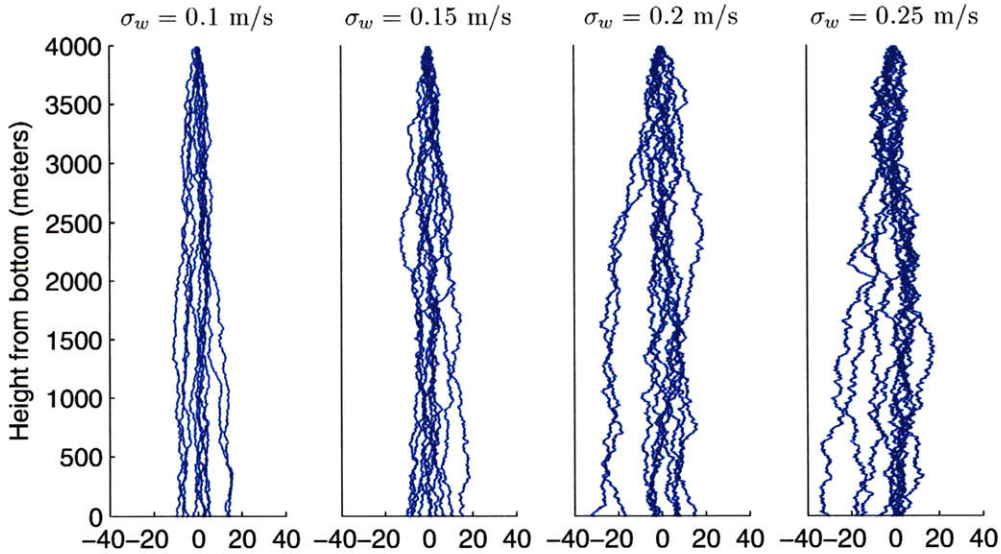

Figure 3-12: Extended Kinematic Model, Plot with varying σ_w while $\sigma_\alpha = 0$, constant gain $K = 1$

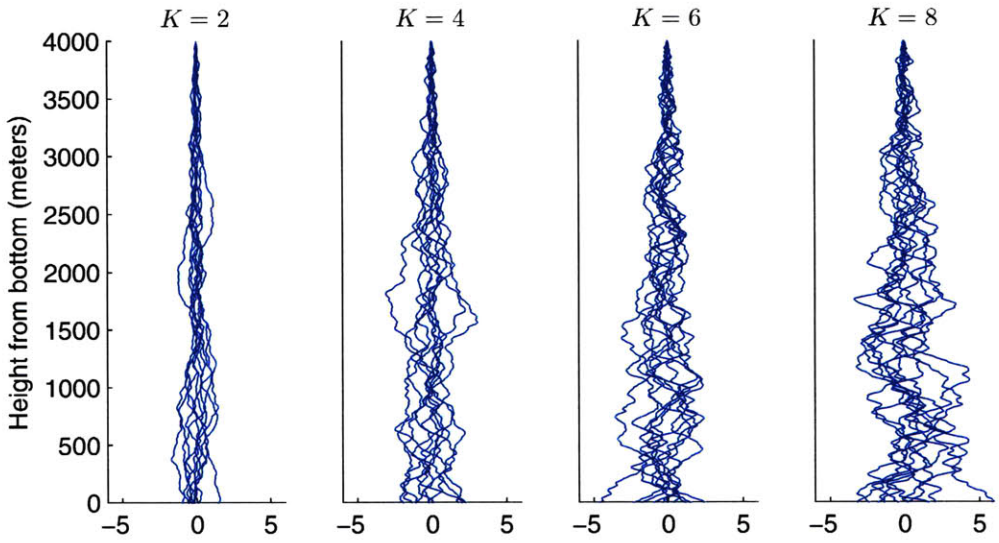

Figure 3-13: Extended Kinematic Model, Plot with varying K while $\sigma_w = 0$, $\sigma_\alpha = 1°$

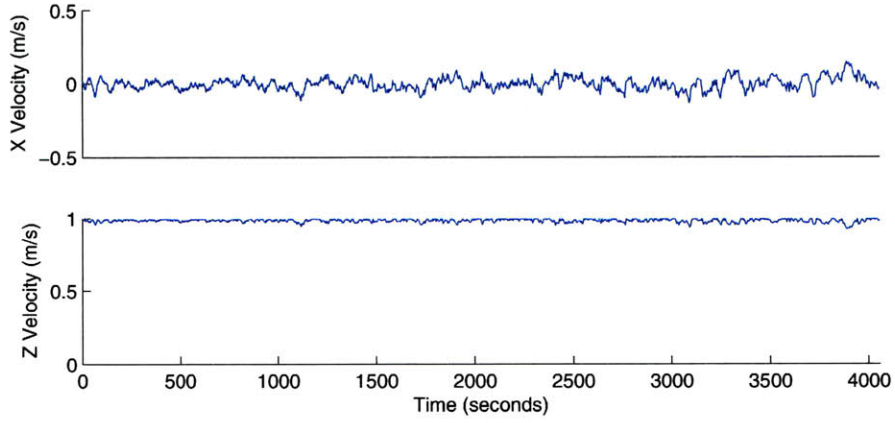

Figure 3-14: Extended Kinematic Model: Horizontal and Vertical Velocity Over Time With Low Pass Filter ($\tau = 26.6$ sec), $K = 10$, $\sigma_w = 0$, $\sigma_\alpha = 1°$

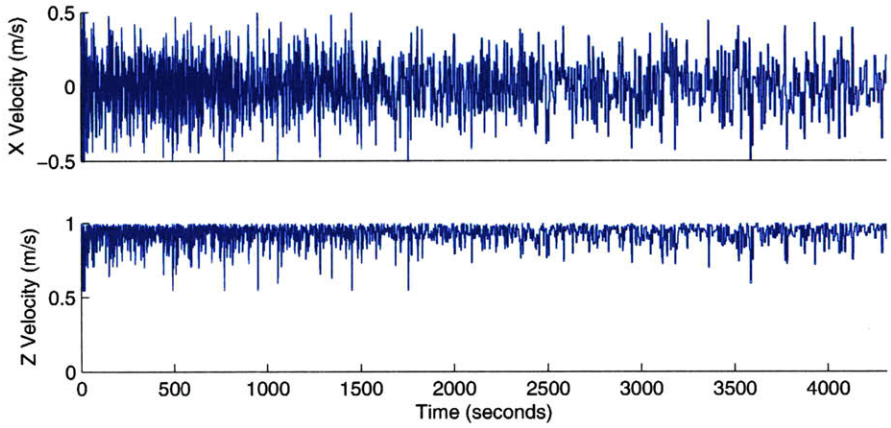

Figure 3-15: Extended Kinematic Model: Horizontal and Vertical Velocity Over Time With No Filtering, $K = 10$, $\sigma_w = 0$, $\sigma_\alpha = 1°$

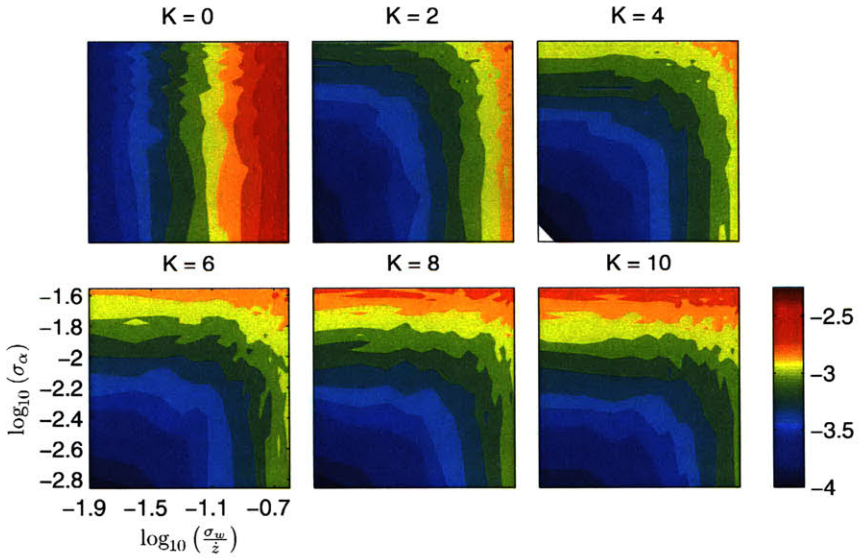

Figure 3-16: Extended Kinematic Model, Non-Dimensional Multiple Contour Plot; Color bar represents non-dimensional landing error $\log_{10}\left(\frac{\sigma_E}{D}\right)$

Chapter 4

Vehicle Design, Fabrication, and Testing

4.1 Overview and Rationale

In addition to modeling the behavior of a vertical glider through simulation, another main goal of the project is to build a prototype vehicle with which to explore the various navigation and control challenges that might be faced. The prototype body helps us understand the behavior of streamlined vehicles while diving, the effectiveness of control fins, and the achievable glide slopes of vehicles with this form factor. The balance of righting moment and control moment helps in designing a vehicle that favors a nose down orientation, but is still easy to control. The performance of the prototype in pool tests helps us determine optimal control strategies for vertical glider vehicles, and the use of an angle-based measurement provides insight into the effectiveness of acoustic navigation measurements in the open ocean. Discussed earlier in simulation, future prototype vehicles will also explore the limitations cause by finite sensor update speeds and noise properties, which limit the effectiveness of the vehicle and help guide the design of vehicles to make efficient use of sensor capabilities.

The vehicle is designed with a simple, streamlined shape, and a blunt, rounded nose to allow for the installation of a camera, used for guidance. Additionally, the vehicle is as self-contained as possible. By keeping power usage and outside connec-

tions to a minimum, the vehicle can remain small and lightweight, and maximize its mobility in the water. The vehicle uses a camera-based guidance system, intended to replicate as effectively as possible the behavior of an acoustic navigation system that would be used by ocean vehicles. We chose the camera system because acoustic systems are ineffective in a closed pool environment, due to signals bouncing off pool walls and the water's surface. We describe the navigation setup for the prototype in depth in subsection 4.3.1. Table 4.1 lists some of the vehicle's physical characteristics.

Length	77.72 cm
Diameter	12.7 cm body , 29.85 cm at tips of fins
Volume	7382.13 cm^3
Weight	7.96 kg
Fin Profile	NACA-0020
Design Dive Rate	40 cm/s
Max Depth	4 meters
Servos	HiTec HS-322HD (x2)
Power Source	8xAA NiMH batteries (1.2 V each, 9.6 V total)

Table 4.1: Vertical Glider Physical Parameters

4.2 Body Description

The vertical glider uses a small negative buoyancy to provide a constant dive rate, along with control fins to steer the vehicle. As noted in the dynamic model discussed in Section 3.3.3, the vertical glider acts like an inverted pendulum: the downward force acting at the center of gravity is counteracted by the upward force at the center of buoyancy. If the center of mass is placed in front of the center of buoyancy, the vehicle will naturally orient into a nose-down position. This pendulum configuration has a motion analagous to a spring-mass-damper system. The weight of the vehicle, along with the space between these two points will determine the "stiffness" of the body, and how much force acting on the fins is required to change the vehicle's orientation.

This righting moment which the control fins must overcome to turn the vehicle

can be determined by:

$$\tau = (\rho V g - mg) \times R_1 = F_{rudder} \times R_2 \tag{4.1}$$

In this equation, R_1 is the distance between the center of mass and center of buoyancy, while R_2 is the distance between the center of mass and the rudder. These dynamics are familiar to the domain of horizontal gliders, and they are equally important to vertical gliders. Without a sufficiently low righting moment, the vehicle will simply fall straight down, and the control fins will have no effect on maneuvering the vehicle.

4.2.1 Nose and Tail

The camera is mounted on a small aluminum frame that is installed in the nose of the vehicle. The end of the nose has been machined out, and a clear acrylic window installed as a view port for the camera.

The tail is made of lightweight machinable urethane foam[1], which has a density of 48 lbs./cu. ft. ($\approx$ 75% that of water), to provide a higher buoyancy to the rear of the vehicle. It was designed with a tapered end, to aid in streamlining the vehicle. The tail cone houses the servo and linkage assembly, which transmits power from two HiTec HS-322HD servos to the rudder and elevator shafts. The servos and fin shafts are connected through four bar linkages, which give a 1:1 ratio between servo rotation and fin rotation. The rudder and elevator fins were designed with the NACA-0020 airfoil profile [4, 11], which signifies a thickness equal to 20% of the chord length. The fins also feature swept leading edges and rounded corners, to prevent the fin from catching on any external obstacles.

The nose and tail cones are attached to the polycarbonate body section with six stainless steel screws on each end, and are sealed against the outside with two rubber O-rings at each end. Because of the air pocket inside of the vehicle, I also installed a small venting hole in the nose cone to allow the escape of excess air while assembling

[1] http://www.mcmaster.com, Item #86595K55

the vehicle. This venting hole prevents the buildup of positive pressure, and is sealed with a nylon screw and Teflon tape. Schematic of the nose and tail sections can be found in Figures C-4, C-5, C-9, and C-10 in Appendix C.

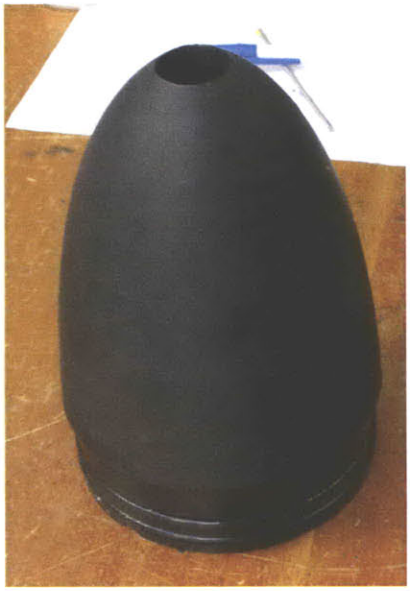

Figure 4-1: Nose Section

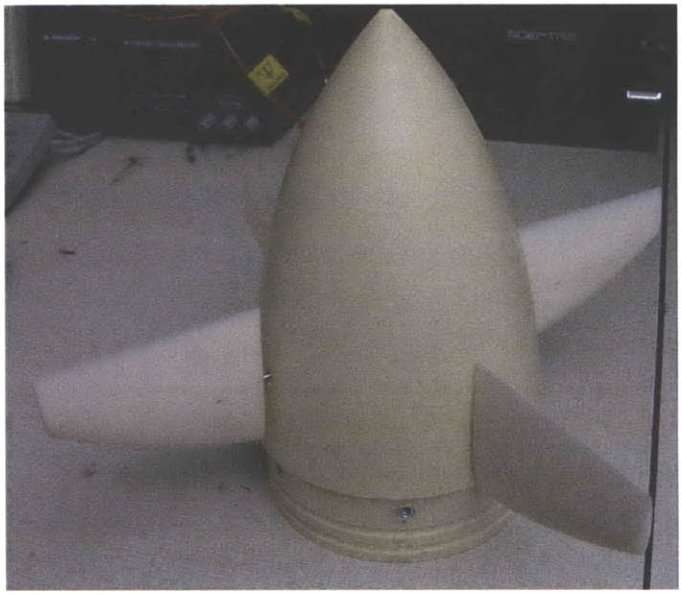

Figure 4-2: Tail Section

4.2.2 Body Section

The vehicle body was designed to house the sensors, electronics, power systems and servos in a compact, streamlined package. The components are mostly plastic to allow for easy machining: the nose is made from ABS plastic, the body from clear polycarbonate tubing, and the tail from machinable urethane foam. It also incorporates an internal frame consisting of stainless steel rods and aluminum supports on which to mount electronics, weights, and the battery pack. Further discussion of the body section components can be found in Section 4.5.1, along with a schematic of the body section in Figure C-3 in Appendix C.

4.2.3 Fins

The vehicle is controlled through servos acting on two pairs of fins in the rear of the vehicle: two elevator fins and two rudder fins. This gives the vehicle the ability to

control itself through pitch and yaw. The main design criteria for the fins is that they be large enough to provide control of the vehicle, while also being lightweight and streamlined, so that they don't catch on obstacles in the water. For the fin cross-section, I chose to use the NACA-0020 wing profile [4]. The NACA-0020 profile is a symmetric airfoil with a thickness at its thickest part equal to 20% of the length of the wing. Equation 4.2 shows the airfoil thickness y_t as a function of chord length x and maximum thickness $t = 0.20$. The equation is normalized about a wing length of 1. Figure 4-3 shows a plot of the shape of the NACA-0020 wing profile.

$$\pm y_t = \frac{t}{0.20}\left(0.29690\sqrt{x} - 0.12600x - 0.35160x^2 + 0.28430x^3 - 0.10150x^4\right) \quad (4.2)$$

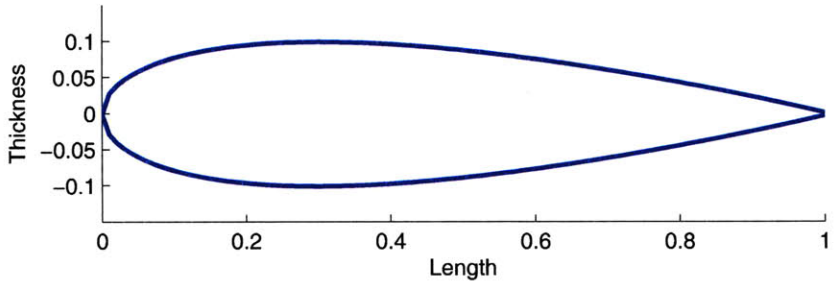

Figure 4-3: NACA-0020 Airfoil Profile, with maximum thickness $t = 0.20$

The rudders and elevators are designed with a sweep angle of 23° and take the form of a standard spade rudder, described in the Principles of Naval Architecture [11]. A schematic of the fins can be found in Figure C-7 in Appendix C.

4.3 Navigation and Sensing

4.3.1 Camera

The vehicle is guided towards the target by a small CMOS camera (352x288 pixel resolution) mounted in the front of the nose cone. The camera used is the CMUcam3 [3], developed by researchers at Carnegie Mellon University. The camera has an

integrated microcontroller board, which using our algorithm is able to perform real-time image processing at approximately 6 frames per second. The current version of the camera's firmware allows a wide variety of programs, allowing software access to exposure, white balance, color, and resolution settings, as well as providing the framework for taking and saving pictures, and accessing individual pixels.

The first implementation of the CMUcam tracking software the location of a dark area on a lighter background. An alternative sensor software design involves tracking a light target instead of a dark one. For the dark target, we used a weighted black plastic disk (7.5 inches diameter), while for the light target, we used a high powered underwater flashlight, with a translucent screen to diffuse the light. Samples of images using each detection algorithm are shown in Figures 4-4 and 4-5, as well as code samples in Listings 4.1 and 4.2 . As shown in Figure 4-5 and observed during pool tests, the method using a flashlight was much more successful at performing in a mixed lighting environment. Discussion from this point on will focus on tracking of a light target.

To track a light target, the camera algorithm computes the center of mass of bright pixels in a recorded image. We only work with the brightness value in a gray-scale image, which can be set with a CMUcam option command. The camera reads the brightness as 8-bit values, with zero corresponding to black and 255 corresponding to white. The camera is capable of taking full color images, but a gray-scale tracking program is simpler to implement and less computationally intensive. For each loop of the program, the CMUcam takes a picture and saves it to the memory buffer. The program then cycles through each pixel in the image, and compares the brightness value of the pixel to a predetermined threshold value. If the pixel value is greater than (lighter) than the threshold, it will include the pixel in a running average of the X and Y positions of all sufficiently bright pixels. The threshold value allows the operator to adjust for varying ambient lighting conditions in the testing tank, as well as for changes in the light output of the flashlight between missions. The code sample in Listing 4.2 shows the program loop used to compute the center of mass of the target. The camera's microcontroller board then uses the location of the target

Listing 4.1: Dark Target CMUcam Image Sampling

```
while (cc3_pixbuf_read_rows (img.pix, 1)){
   for (uint16_t x = 0; x < img.width; x++){
      uint8_t black = ((uint8_t *) img.pix)[x];
      if (black < MIN_BLACK_INTESITY) {
         sum_x += (MIN_BLACK_INTESITY-black)*x;
         sum_y += (MIN_BLACK_INTESITY-black)*y;
         coef += (MIN_BLACK_INTESITY-black);
         counter++;
      }
   }
   y++;
}
xc = (sum_x/coef) - X_CENTER;
yc = (sum_y/coef) - Y_CENTER;
```

Listing 4.2: Light Target CMUcam Image Sampling

```
while (cc3_pixbuf_read_rows (img.pix, 1)){
   for (uint16_t x = 0; x < img.width; x++){
      uint8_t white = ((uint8_t *) img.pix)[x];
    if (white > MIN_WHITE_INTENSITY) {
         sum_x += x;
         sum_y += y;
         coef += 1;
         counter++;
      }
   }
  y++;
}
xc = (sum_x/coef) - X_CENTER;
yc = (sum_y/coef) - Y_CENTER;
```

in its field of view to compute the command that will be sent to the control fins. The initial coding and camera testing were accomplished by Michael Fertin, a graduate exchange student assisting on the project. Source code for the CMUcam tracking program can be found in Appendix B.1.

4.4 Power, Storage and Communications

Early in the design process, we considered whether the vertical glider should be completely self-contained, or operate with a tether. The advantages of a self-contained vehicle include the lack of communications or electrical ports which can cause leakage, greater freedom of movement, and quicker operation due to short data connections. The benefits of a tethered vehicle include higher power availability, easier data retrieval and communications, and the ability to use the tether as a support for the

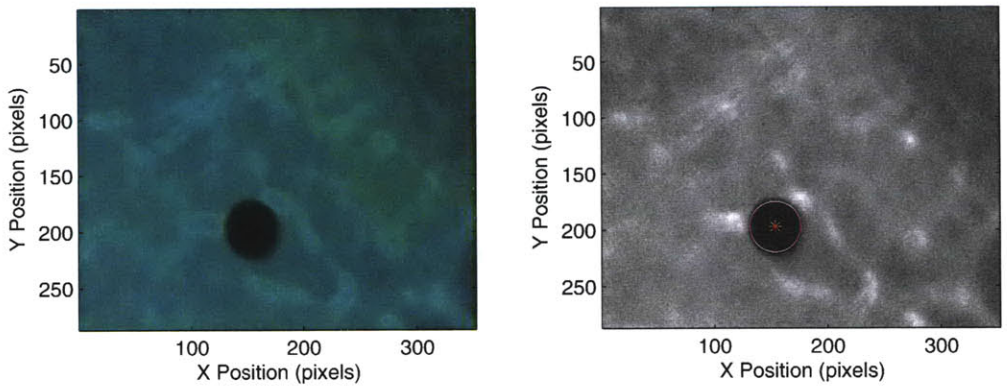

Figure 4-4: CMUcam Image Processing Sample; left: original color image, right: grayscale image with target location and size

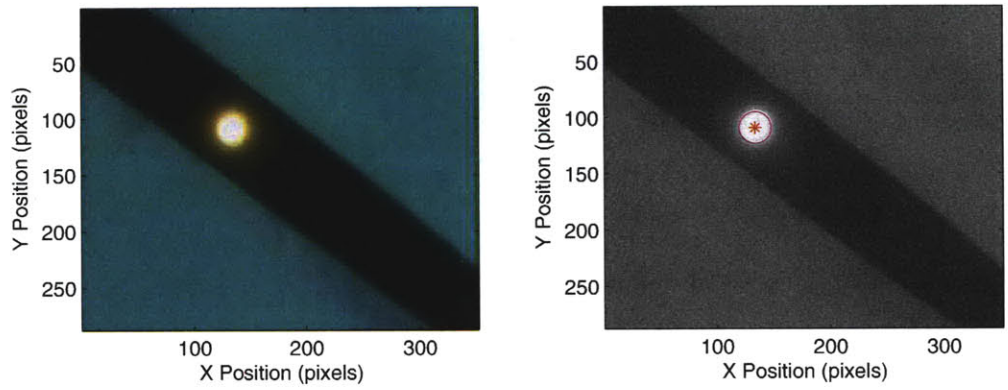

Figure 4-5: CMUcam Revised Image Processing Sample; left: original color image, right: grayscale image with target location and size

vehicle when retrieving it from the bottom. Ultimately, we decided to develop a self contained vehicle, since the power usage was generally low and connector interfaces have the potential to form leaks and cause damage to the electronics.

We compared several types of batteries, including Lithium Ion, Nickel Cadmium (NiCad), Nickel Metal Hydride (NiMH), and standard alkaline batteries. The lithium and NiMH batteries were the most likely options, as they have a high power density, are rechargeable, and do not suffer the "memory" issues commonly encountered in NiCad batteries, which require them to be fully discharged. We ultimately chose NiMH batteries for their low cost, and their availability in standard AA size, which are easy to install and replace. Each AA NiMH battery outputs 1.2 V, with a 2000 mAh lifespan. To achieve the voltages required for our electronic components, we

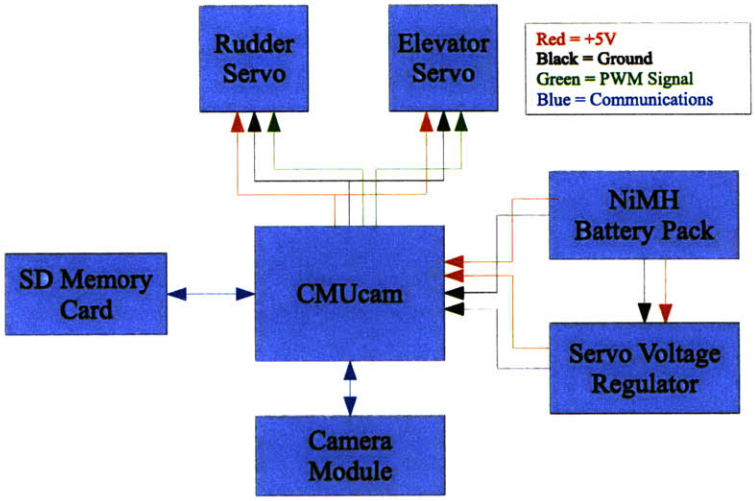

Figure 4-6: VGR Power & Communications Diagram

connect eight of these batteries in series for a total output of 9.6 V. As shown in Table 4.2, the vehicle's electronics draw approximately 450 mA during operation, which would allow the vehicle to operate for 4 hours between charges. This allows several testing sessions to be run per charge, but it is generally safest to have the batteries fully charged at the start of each session.

Component	**Voltage**	**Current Draw**
CMUcam	5 V	130 mA
Servos (x2)	5 V	160 mA each
Total		450 mA

Table 4.2: Power Consumption

An important consideration for the power distribution is making sure all components receive properly controlled voltages. Figure 4-6 shows a diagram of the power and communications connections between the various components of the vertical glider. While the battery pack may provide 9.6 V when fully charged, this begins to decrease as the battery drains. The CMUcam includes an integrated voltage regulator to provide the board with a constant 5 V supply, but needs 7-12 V input to operate properly. The CMUcam also has circuitry to provide regulated power to the servos, but the high current draw of the servos is too much to be handled by the

CMUcam's regulator. A momentary spike in current draw by the servos could cause the microcontroller to reset. For this reason we provided the servos with a separate voltage regulator, and only passed the servo control signals through the CMUcam. The CMUcam has the ability to pass through the separately controlled voltage by setting a jumper pin, which instructs the board not to feed the servo power through its own voltage regulator.

The data from all missions are stored on an SD memory card integrated into the microcontroller on the CMUcam. This allows the vehicle to keep a record of all sensor measurements taken during a mission, and also to save JPEG formatted images taken by the camera. Once a mission is completed the card can be removed and read by any computer with a memory card reader. Additionally the CMUcam features a serial port that can be used for programming as well as data retrieval. Future designs will make use of wireless protocols to retrieve data without disassembling the vehicle.

4.5 Vehicle Fabrication and Assembly

The prototype vehicle was constructed at the MIT Edgerton student shop and Laboratory for Manufacturing and Productivity (LMP) using a variety of machine tools, discussed in depth in this section. To ensure durability in the underwater environment, most components were fabricated using plastics or corrosion resistant metals such as stainless steel, except where machining methods required alternate materials. The vehicle body is divided into three sections: body, nose, and tail. The body section houses ballast weights, batteries, power regulation circuitry. The nose contains the CMUcam and its associated microcontroller board. The tail section houses the control fins, servos, and gear linkages. A schematic of the assembled vehicle can be found in Figure C-1, along with an exploded view of the assembly in Figure C-2, both in Appendix C.

Figure 4-7: Assembled Vertical Glider Prototype

4.5.1 Body Components

The pressure vessel for the vehicle is an 1/8 inch thick polycarbonate plastic tubing, with a diameter of 5 inches. It connects to the nose and tail sections through double O-ring seals, and is fastened in place with six screws on each end. The fastening screws are placed on the wet side of the O-rings to ensure proper sealing. The polycarbonate body also has a chamfer cut into the inside edges to facilitate smooth assembly of the O-rings.

The body section was designed to contain a rigid frame on which all main components could be mounted, to ensure that the plastic body section does not bear the load of the weights or electronics. The frame consists of four stainless steel threaded rods, connected by several aluminum divider plates. The use of threaded rods, combined with push-button quick threading nuts allows the quick assembly and adjustment of balancing weights, as well as electronic components. The adjustable nature of the body section components is most useful in positioning the ballast weights to ensure an optimal righting moment for the vehicle.

The body section also includes several mounting brackets for electronics, including the batteries and future sensor installations. These brackets were cut from 1/4" and 1/8" acrylic sheets using a laser cutter. The laser cutter allows very precise geometry in thin two-dimensional plastic pieces. Because of their weight, the bracket

for the batteries was cut using 1/4" acrylic. The battery mount also includes a switch for powering the entire vehicle on and off. This switch is mounted close to the polycarbonate tubing, and has a strong neodymium magnet attached to it. This allows power to be switched during testing while the vehicle is sealed, without the need for a waterproof interface.

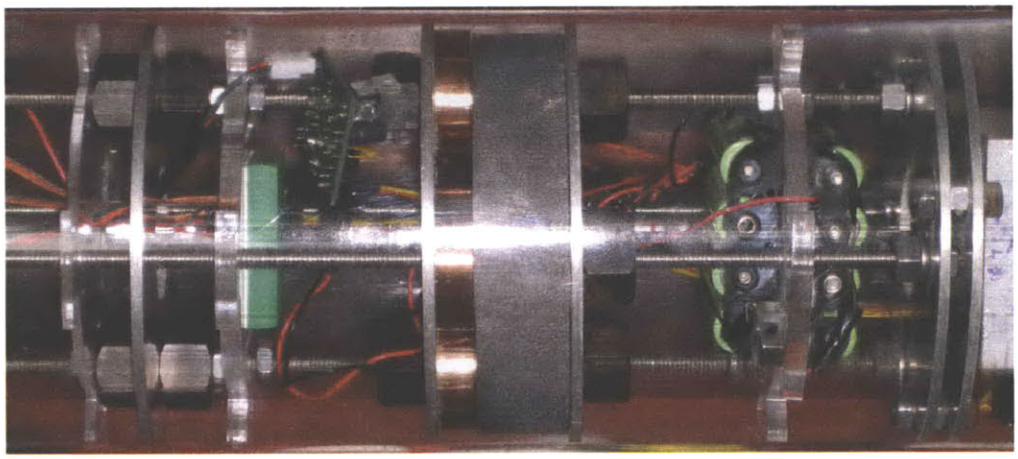

Figure 4-8: Vehicle Body Section

4.5.2 Nose Components

The nose section consists of the nose cone, viewport window, and camera mount. The nose cone was constructed from ABS plastic on a CNC lathe at the MIT Edgerton machine shop. By using the computer controlled lathe, I was able to use the CAD design of the nose to machine a precise nose profile. The viewport, necessary for the CMUcam to see the target, is a small disc of 1/8" acrylic, cut using the laser cutter. It was glued and sealed in place using RTV (room temperature vulcanization) silicone adhesive.

The camera mount contains the lens module and CMUcam controller board, which handle sensing, data storage, and servo control. The camera mount is meant to keep the lens module in a stable position pointing directly forward, so that the camera readings will provide an accurate picture of the vehicle's orientation. It is constructed out of aluminum parts machined using a water jet cutter. The CMUcam is sold

with the lens module and controller board connected directly, but for the purposes of mounting the camera in the front of the vehicle's nose, it became necessary to separate the two and connect them via a flexible ribbon cable. Early stage designs called for the controller board to be positioned in the vehicle main body, to allow easy access to the memory card and connection ports. However, this required using a ribbon cable approximately 6-8 inches in length, and this caused an inconsistent connection between the camera and board when pictures were recorded. Often, the image returned would be garbled or completely corrupted. It was discovered that the only reliable way to prevent this was to minimize the cable length, keeping it under 3 inches. This requires mounting the controller board in the nose along with the lens module. Once the lens module and controller board are connected, the entire camera mount assembly is installed in the nose cone, and is fastened in place with a machine screw.

4.5.3 Tail Components

The tail section contains the fins, servos and gear linkages to connect the two together. The fins were fabricated from ABS plastic using a Dimension 3D printer, according to the CAD design described earlier. While 3D printed parts are generally only for demonstration purposes, the parts made from thermoplastic 3D printer are strong enough to be used directly in the vehicle. However, for higher production volumes, 3D printing becomes infeasible, and a process such as injection molding would be far more efficient for plastic parts.

The servos and fins are connected via four-bar linkages, which transmit the rotation of the servo to the rudder and elevator axles with a 1:1 ratio. The linkages are constructed from aluminum pieces cut using a water jet cutter, and assembled with locking nuts and Teflon washers to facilitate smooth motion. The fins are mounted on stainless steel shafts and fixed in place with set screws. Since parts of the axles will be exposed to the water, I chose stainless steel for its strength and excellent corrosion resistance. A schematic of the servo linkages can be found in Figure C-6 in Appendix C.

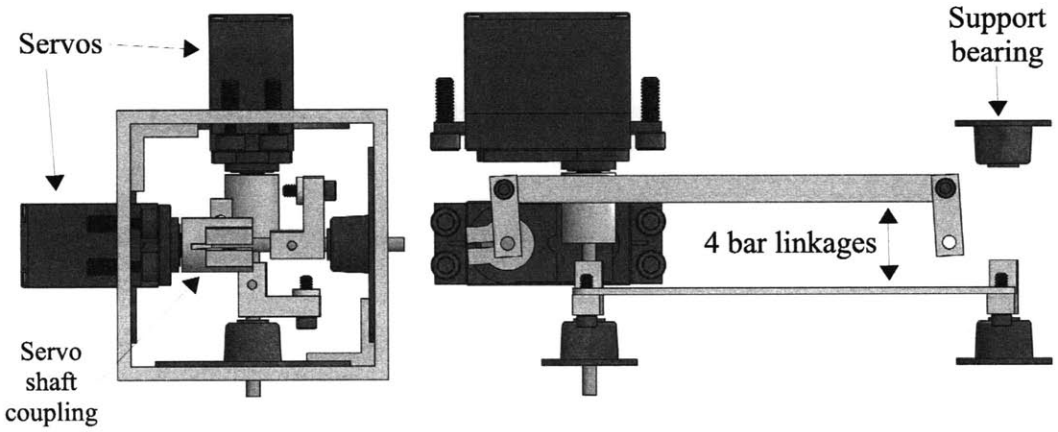

Figure 4-9: Servo Mount Assembly

4.6 Prototype Control System

The control system implemented on the prototype vehicle is conceptually different than that of a full size vehicle utilizing acoustic navigation, and so warrants discussion. As shown in Figure 4-10, the nature of the camera guidance system yields a new set of variables which the vehicle's sensor systems measure. θ represents vehicle pitch, while ζ is the the angular error of the vehicle. Thus, when the vehicle is pointing directly at the target, ζ should equal zero. Finally, δ represents the fin angle, which is the control input for the system.

The simplest control system for the vertical glider is one in which the software attempts only to keep the target in the center of its field of view. We call this "angle only" control because the only control input is the error angle between the vehicle axis and the target, shown in Figure 4-10 as the angle ζ. As explained previously, this angle error is read by the camera as a pixel offset, although the conversion to angular error is easily accomplished with knowledge of the camera's fields of view in the horizontal and vertical directions. The current prototype model uses this angle only method of control, with a proportional controller translating commands to the servos. This was chosen because of its computational simplicity, enabling faster updates to

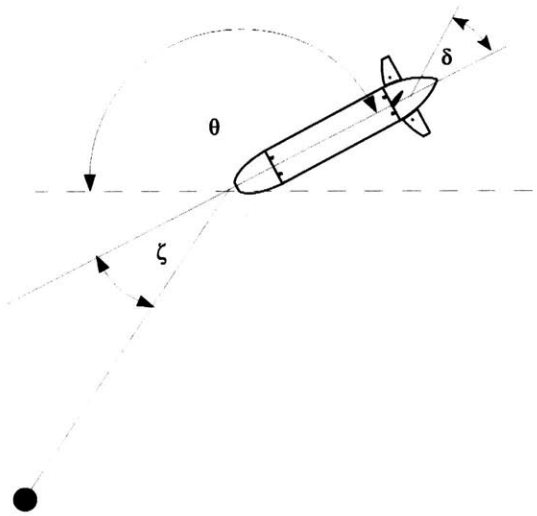

Figure 4-10: Vertical Glider Coordinate System

the fins, and because it facilitates easy calibration of the camera and servos.

$$\begin{aligned} \dot{z} &\approx 40 \text{ cm/s} \\ \delta &= K_1 \zeta \\ \theta &= K_2 \delta \\ \dot{x} &= \sin\theta \end{aligned} \tag{4.3}$$

4.6.1 Angle Only With Offset

An alternative method for controlling the vertical glider involves integrating camera measurements with depth and roll/pitch measurements, to enable the "angle-only" system to follow a more complex path. The system would aim to position the vehicle directly above the target as soon as possible, and then continue on a near vertical path to the bottom. This is accomplished by introducing an offset to the camera's measurement, such that the vehicle drives itself at a shallower pitch than normal. The value of this offset would be determined by the depth and pitch measurements, enabling the vehicle to drive at a shallower angle early in the mission. This would also ensure that if the vehicle was directly over the target pointing straight down, the pitch measurement would reflect this and minimize the offset to drive the vehicle

straight down. A diagram of the offset angle is shown in Figure 4-11, where ζ_o depicts the altered measurement. This method has not yet been implemented on the current platform, but is planned for future testing.

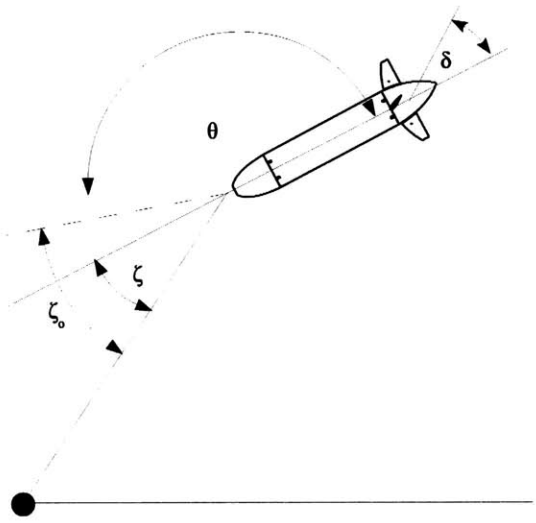

Figure 4-11: Angle-Only Control with Offset

4.6.2 Noise Simulation

While not yet implemented in hardware, future prototypes will also apply noise terms to the test vehicle's measurements to allow us to recreate the limitations of a USBL navigation system. This is accomplished by modifying the camera data before it is sent to the servos for actuation. The camera records the position of the target in its field of view, and outputs the results in terms of the target's distance from the center in pixels. Before the measurement is sent to the servos, we add an additional random number using a random number generator from an on-board microcontroller, such as the Arduino.

One important consideration is that random number generators in embedded systems generally produce uniform random variables, while for our purposes of generating Gaussian noise, we require a normally distributed random number. To accomplish this, we use the Box-Muller transformation [5], which can convert between the two

distributions. For two uniformly distributed random variables, u and v:

$$s = u^2 + v^2 \tag{4.4}$$

$$z_0 = u\sqrt{\frac{-2\ln s}{s}} \tag{4.5}$$

$$z_1 = v\sqrt{\frac{-2\ln s}{s}} \tag{4.6}$$

This yields two normally distributed random variables, z_0 and z_1 with a mean of zero and standard deviation of one. This is the polar form of the Box-Muller method. It is ideally suited to implementation on small robotics platforms, since unlike the basic form of the Box-Muller transformation, it does not require the use of trigonometric functions, which are computationally costly.

4.7 Prototype Testing Results

The first step in verifying the correct operation of the physical system is to test out the components in a laboratory setting. This consisted of calibrating the center points and ranges of motion of the servos, and verifying the camera algorithm's ability to track a moving target in real time.

4.7.1 Servo Configuration

Initial tests were conducted in the lab to verify the mechanical linkages connecting the servos to the fins. To ensure that the CMUcam sends the correct commands to the servos, it was necessary to set the neutral point and ranges for each servo. The neutral point is the servo value (0-255) at which the fins would point straight, resulting in zero net torque on the vehicle. Figure 4-12 show the relationship between measured target location and servo commands in both the vertical and horizontal directions. The servos have upper and lower saturation limits to prevent the linkages from moving beyond their capabilities.

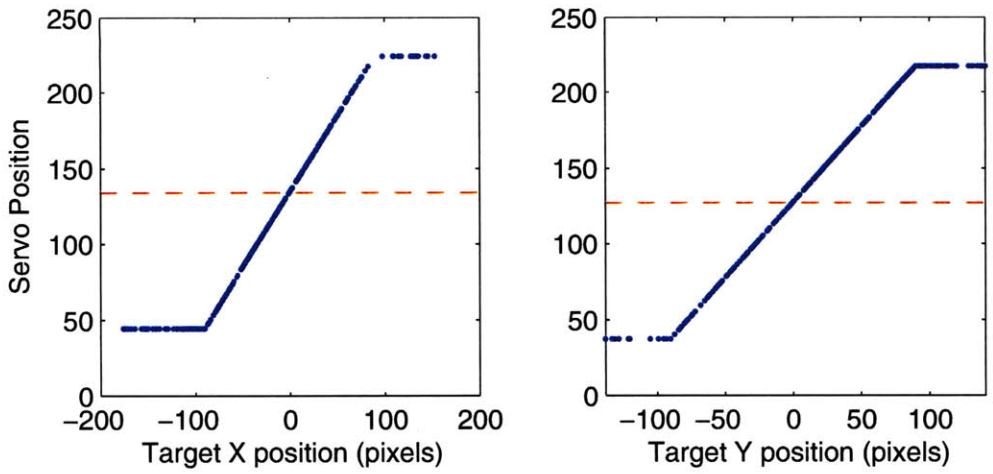

Figure 4-12: Plot of CMUcam measurement data vs. servo commands for X and Y axes; servos receive command from 0-255, have been set to saturate at ±90 of their neutral point, shown by dashed red line

4.7.2 Pool Testing

After preliminary lab tests of the vehicle, we began pool testing in the Alumni Pool at the MIT main campus. The testing pool has a depth of 13 feet (3.96 meters). The pool has a white tile floor, with a series of 12 inch wide black lane lines running the length of the pool on the bottom. Our pool testing was divided into several stages, meant to verify correct operation of the vehicle before conducting target tracking experiments.

1. Verify the integrity of watertight seals at full pool depths
2. Calibrate lead and copper weights to achieve desired dive rate
3. Calibrate camera sensor threshold value
4. Verify proper motion of control fins
5. Conduct target tracking experiments

By taking these steps ahead of time, we can minimize difficulties during the tracking experiments. The first step was to make sure the O-ring seals on the nose and tail cones prevented water from leaking into the vehicle. While this had been tested

in shallow water in the lab, it is also important to test the seals at full pool depths, as the pressures encountered will be higher at the bottom of the pool. In addition to the O-ring seals, we installed moisture absorbing desiccant packs to remove any residual moisture from the inside of the vehicle. The next step in the process was to position weights inside the vehicle to provide the proper amount of balance during tests. While mostly a trial and error process, proper configuration of the weights allowed the vehicle to dive with enough velocity to provide sufficient flow over the control fins, while not diving so fast as to cause damage to the vehicle upon landing.

After checking the seals and installing the weights, we performed several checks of the camera system to calibrate the threshold value to light levels common to the pool environment. This consisted of running a camera program that saves images to memory repeatedly to observe the light levels at the surface, the bottom, in shadow and light, and the contrast between darkened lane lines and the light background. As discussed in subsection 4.3.1, earlier software designs aimed to track a dark target on a light background. Through several rounds of calibration and camera testing, we observed that the system was unable to reliably distinguish between the target and lane lines or areas darkened by shadow. We attempted to cover the lane lines with white plastic sheeting, but the variability of lighting conditions and the presence of shadows still presented difficulties for the camera's tracking program, so we switched to using the underwater flashlight as a target.

After switching the camera tracking algorithm, we performed the camera and servo calibration tests again, and proceeded with pool trials of the vehicle. The goal of these experiments was to observe the dynamics of the vehicle, and to test the tracking capabilities of the system. We began by dropping the vehicle from directly over the target, and gradually moved the starting point further away from the target, to observe the maximum horizontal velocity of the vehicle. Initial tests have confirmed that the vehicle is effective at tracking the target and guiding itself toward it from a variety of starting points. Figures 4-13 and 4-14 show groups of images taken with an underwater video camera during the tests. Each shows the vehicle along its path to the bottom, maintaining a constant glide slope aimed at the target.

We have also observed that testing conditions seem to have a minor effect on vehicle stability. Initial pool trials involved releasing the vehicle with zero initial vertical velocity. As expected, the vehicle accelerated for a brief period before reaching a steady dive rate. This initial period of acceleration can result in oscillations of the vehicle, most likely because the control fins are not exerting enough force. For the second round of tests, we gave the vehicle a slight push downwards to help stabilize the trajectory earlier in the descent. We also reduced the control gain from 1 to .5, which helped further reduce oscillations of the vehicle.

Throughout all of these tests, the CMUcam recorded data for elapsed time, X and Y target position, and target size. While not used in the control loop, the value of target size is helpful in determining the distance to the target in the absence of other sensors. Figures 4-15, 4-17, and 4-19 show the X position, Y position, and target size versus elapsed time for three missions. The target size is shown on a logarithmic scale ($\log_{10}$). The oscillations observed during testing are apparent in Figure 4-15, with the oscillations in both the X and Y directions having a period of about 3.5 seconds.

Figures 4-16, 4-18, and 4-20 show scatter plots of X position versus Y position over the course of a mission. At each data point, a circle is overlayed with an area proportional to the recorded target size. While the target position does not appear to converge, this is because the recorded target position is an angular measurement, and small perturbations in the vehicle's orientation will cause larger changes in target location with decreasing separation between them. For this reason we have also included an "adjusted position" plot, which shows X and Y target locations divided by a scaling factor according to target size. These provide a clearer picture of the vehicle's ability to lock on the target at a variety of distances, and the convergence towards the center as the vehicle approaches the target.

Mission 2, shown in Figures 4-17 and 4-18, shows a one second delay in data collection at the 32.5 second mark. This is caused by the camera program saving an image to memory, during which the tracking program pauses for approximately one second. After the image is saved, the program resumes, but the vehicle will usually oscillate slightly because the target has strayed from the center of the field of view.

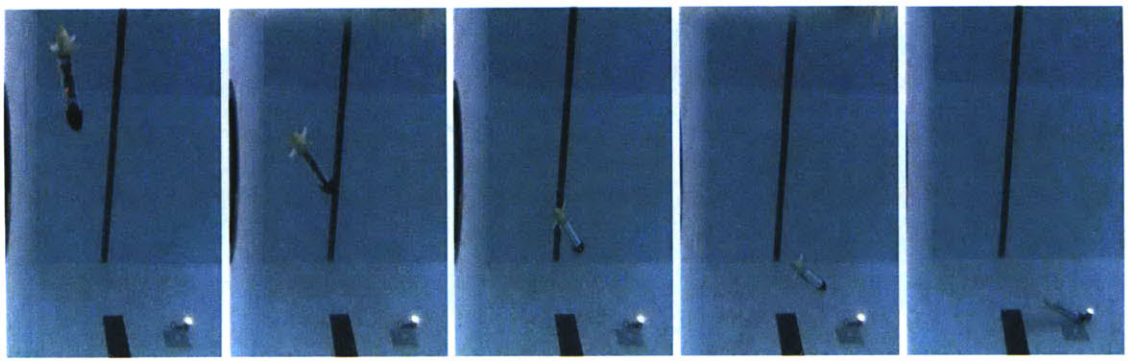

Figure 4-13: Underwater Images of Test Vehicle in Flight 1

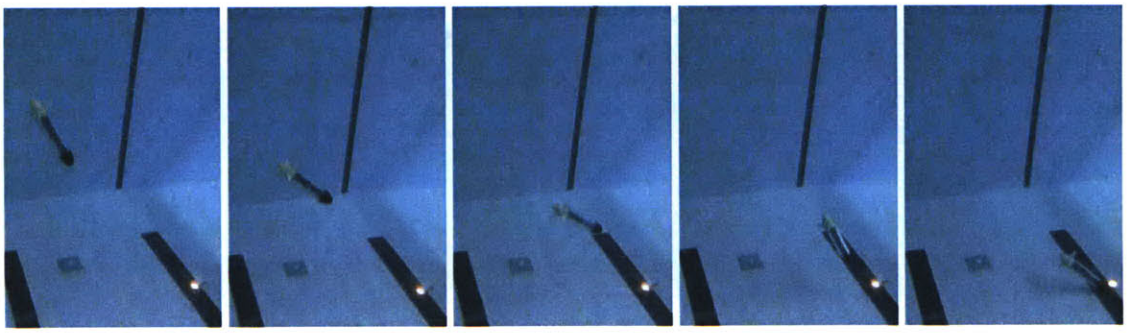

Figure 4-14: Underwater Images of Test Vehicle in Flight 2

The data from several more missions are included in Appendix A.2. Mission 4, shown in Figures A-1 and A-2, shows the characteristic oscillations of the target location, as well as the delay cause by writing the image to memory at the 88.1 second mark. After the image write, the vehicle reacquires the target and begins to oscillate back towards the center. This mission also shows an anomaly at the 89.6 second mark, where the vehicle momentarily loses the target, and computes the target size as zero. However, after one missed data point, the target is reacquired within 1/3 second, and the mission continues. Mission 5, shown in Figures A-3 and A-4, is a relatively fast mission, completing in approximately 5 seconds. As mentioned, later missions began with a higher initial velocity, to improve the vehicle's stability. Mission 5 does not suffer from any delays due to image writing, and the vehicle is able to keep track of the target for the duration. Beginning at the 127 second mark, the vehicle begins to oscillate slightly, but is able to stay on course.

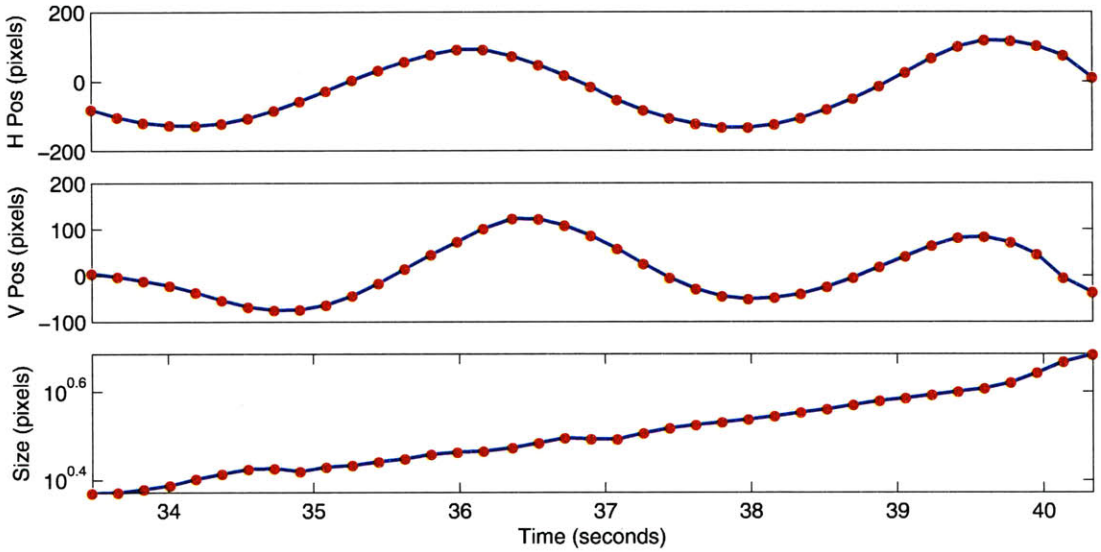

Figure 4-15: Mission 1: Plot of X position, Y position, and target area vs time; red dots denote discrete data points, time is printed on log scale

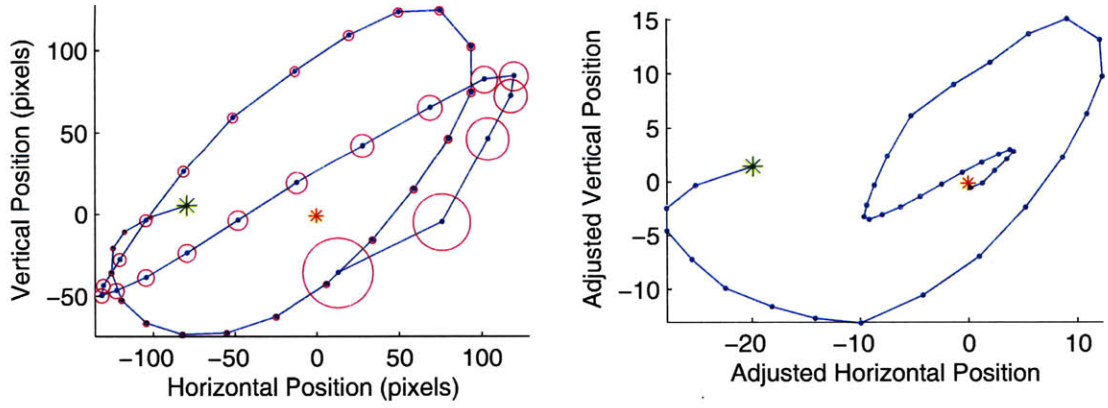

Figure 4-16: Mission 1: Scatter plot of X vs. Y position, magenta circles show target area, growing progressively larger as vehicle nears the target, red star shows field of view center, green star shows target starting location

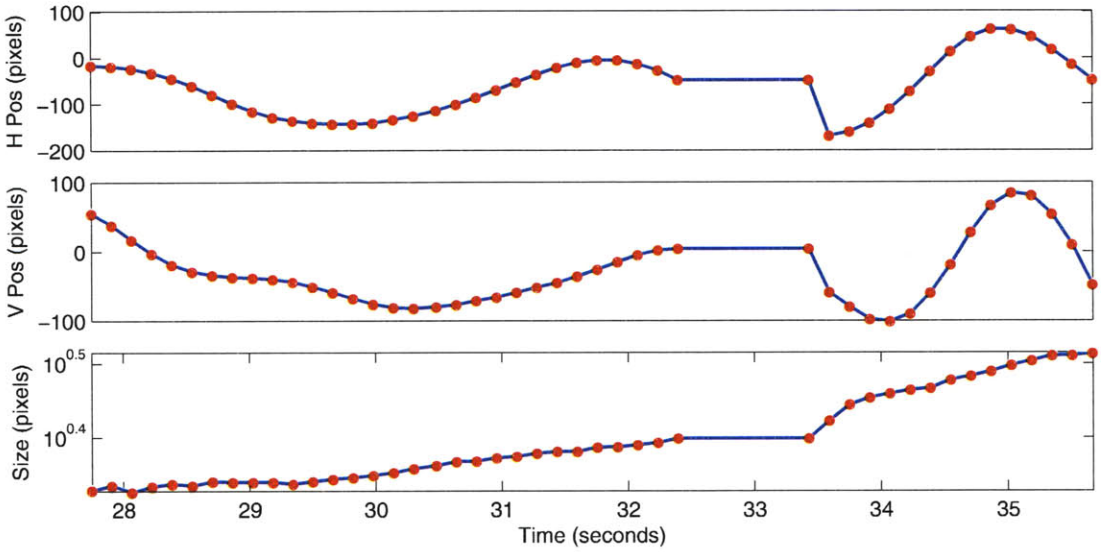

Figure 4-17: Mission 2: Plot of X position, Y position, and target area vs time; red dots denote discrete data points, time is printed on log scale

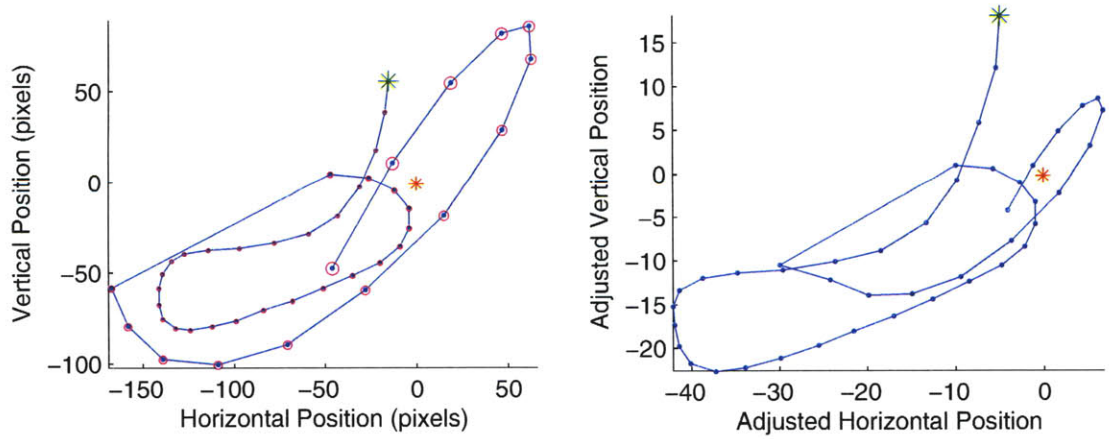

Figure 4-18: Mission 2: Scatter plot of X vs. Y position, magenta circles show target area, growing progressively larger as vehicle nears the target, red star shows field of view center, green star shows target starting location

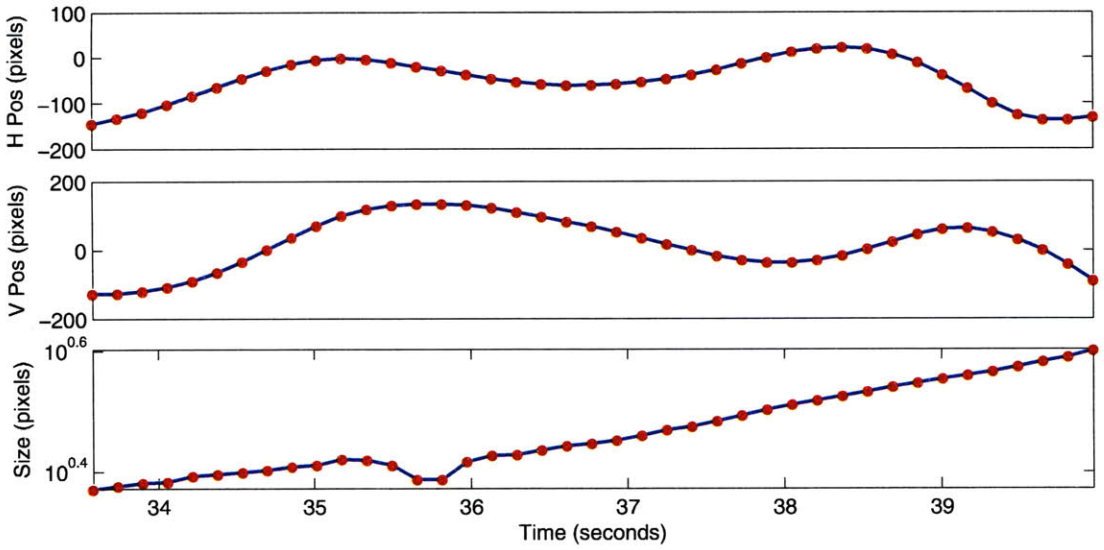

Figure 4-19: Mission 3: Plot of X position, Y position, and target area vs time; red dots denote discrete data points, time is printed on log scale

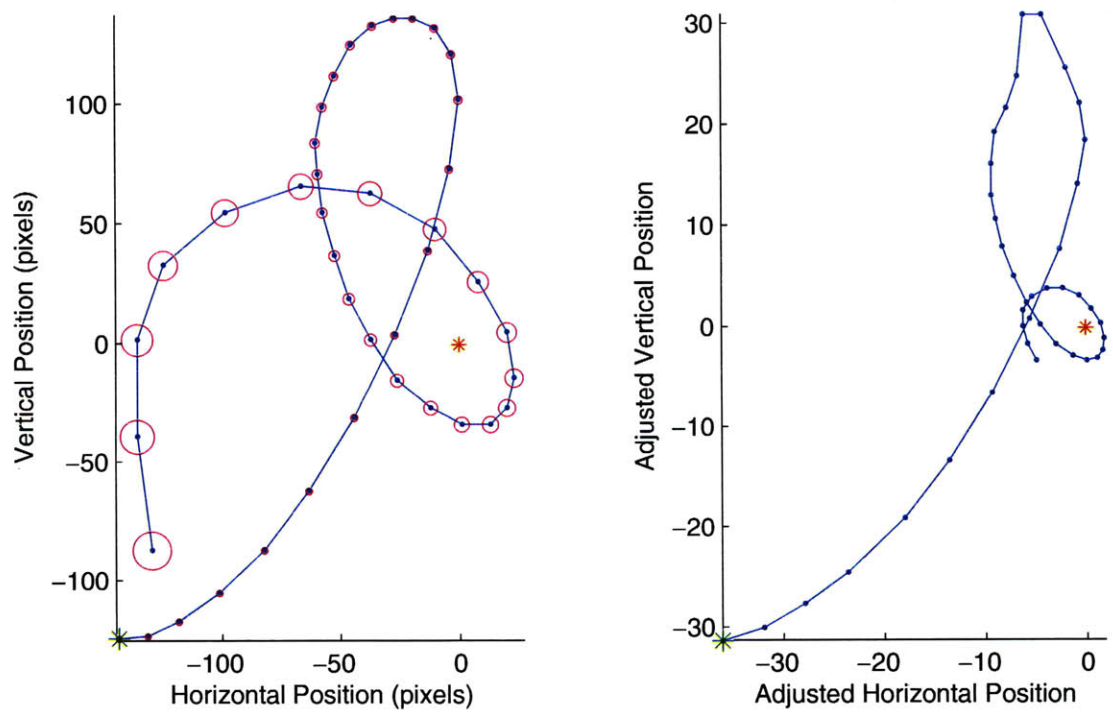

Figure 4-20: Mission 3: Scatter plot of X vs. Y position, magenta circles show target area, growing progressively larger as vehicle nears the target, red star shows field of view center, green star shows target starting location

Chapter 5

Conclusions

This thesis has sought to explore the challenges faced by subsea equipment delivery missions, namely how to effectively guide a payload to a seafloor target using the natural descent of the vehicle. This mission faces many challenges, including uncertain waves and current conditions, sensor noise, and delays in position updates. The vertical glider platform seeks to accomplish this mission by enabling a vehicle to guide itself using the USBL acoustic navigation system already installed on many marine research vessels. We have demonstrated the concept of target tracking in a diving vehicle to be effective in pool tests, and computer simulations have shown the relationship between placement accuracy and navigation system parameters, environmental conditions, and control gains.

5.1 Summary of Work

In Chapter 1 we discussed the current body of research in underwater vehicles, gliders, navigation and near-seafloor operations. While current research with autonomous underwater vehicles focuses mostly on self-propelled vehicles and horizontal gliders, there is great potential in vertically oriented vehicles for underwater missions, as vertical gliders would provide the same low power usage and simplicity as horizontal gliders.

In Chapter 2 we explored several design concepts for accomplishing the task of

improved equipment delivery to the seafloor. Each concept was evaluated for its advantages, disadvantages, cost, and level of complexity. Some concepts involved only software or operational changes, while others required major mechanical and electronic alterations to current subsea delivery platforms. The solution chosen for further study was a streamlined vehicle with active control surfaces, described in further detail in Chapter 4. This concept was chosen because it allows us to learn the most about vertical glider behavior, while remaining simple enough to design, build, and test within a reasonable timeframe.

In Chapter 3 we developed several models for vehicle performance and analyzed the effects of various system parameters on delivery accuracy. The basic kinematic, extended kinematic, and dynamic models each propose a progressively more complex paradigm for vehicle motion. The basic kinematic model proposes a direct connection between control input and horizontal motion. That is, the vehicle can change direction instantly in response to input. The extended kinematic model adds to this a relationship between vertical and horizontal speeds, owing to the drag induced by a change in orientation. Additionally, we began to look at the benefits of filtering sensor data to improve the quality of position information. The dynamic model takes into account the effects of gravity, buoyancy, inertia, lift, and drag on the motion of the vehicle, to present a complete picture of vehicle motion. The kinematic models were integrated into MATLAB simulations of a test vehicle deployment, to observe the effects of varying parameters on vehicle performance.

In Chapter 4 we explored the design and development of a prototype vertical glider to test out control and navigation strategies. The main goal for this vehicle was a small, self-contained vehicle that could be tested in a pool environment. We chose to use a visual navigation system, powered by the CMUcam, a small digital camera combined with a microcontroller, which performs real-time target tracking. The vehicle has yaw and pitch actuation through the use of control fins in the rear, connected to hobby servos. The vehicle itself was contructed from plastic, aluminum, and stainless steel, to limit the effects of corrosion from moisture. Once assembled the vehicle was used in a series of pool tests to observe its ability to track a target on

the pool bottom and guide itself toward that target.

5.2 Discussion of Results

5.2.1 Simulation Analysis

Through computer simulation, we have sought to gain a clearer picture of how vehicle performance is affected by changing system parameters. It is fairly obvious that increasing measurement and process noise (σ_α and σ_w) will cause degradations in landing accuracy (σ_E). However, less clear is which factor is more important, as well as the effect of control gain on the system.

Through analysis of the basic and extended models, we observed that the process noise σ_w, which is an approximation of the side-to-side motion caused by ocean currents, appears to have the most effect on vehicle performance. The errors associated with σ_w cause a spreading out of the vehicle's average position as it descends in the water, while errors associated with σ_α or K remain approximately constant over the course of a mission. Additionally, measurement error only severely affects vehicle performance at higher control gains. Operators who are aware of conditions that may increase measurement noise can configure the vehicle with a lower control gain to minimize this impact.

While the basic kinematic model provided us with a straightforward view of the effects of error on vehicle performance, the extended kinematic model allowed us to observe slightly less obvious behavior. Despite providing performance very similar to that of the basic kinematic model, the filter implemented on the extended kinematic model was able to greatly reduce the horizontal velocity, providing much more stable motion. The basic and extended kinematic models have served as stepping stones towards more detailed simulations of the vertical glider, and will assist in the design of more complex computer models, as well as the design of more advanced prototype vehicles.

5.2.2 Vehicle Design Analysis

The vehicle design phase focused on developing a vehicle that could be easily assembled and operated by 1-2 people in a closed pool environment. Throughout the design process, I encountered several challenges in hardware, software, and fabrication that required changing the design plans for the vehicle.

One of the main challenges faced in the design phase was the mechanical system connecting the control fins to the servos that operate them. Initial designs involved the use of bevel gears to connect the servo shafts and fin axles at right angles to each other. This would allow optimal placement of the servos, while minimizing the space they took up. However, in the fabrication process it became evident that the precision required for operation of the bevel gears was higher than what could be accomplished with our available manufacturing techniques. Additionally, the use of bevel gears would make the servo assembly a closed apparatus, and once assembled we would be unable to see inside for troubleshooting purposes. After working through a few other designs in CAD software, I settled on using a four-bar linkage system to drive the fins. Each fin axle is connected to a long thin aluminum piece, which is in turn connected to an axle mounted on the servo. This results in a 1:1 gear ratio, allowing servo angular movement to translate directly to movement of the fins. The linkage required no bearings for movement, and was assembled using socket screws and teflon washers to allow the rotating linkage arms to move freely.

Another challenge faced was the task of placing electronics within the vehicle body to optimize assembling and disassembling the vehicle. The CMUcam consists of two parts: the camera module and the microcontroller. These components are connected by a 32-pin connector, and are designed to be mated directly to one another. However, the space limitations caused by mounting the camera module in the front of the nose of the vehicle necessitate an extension cable to connect the two. Original designs placed the microcontroller board in the body section to allow easier connections to the power circuits and servo cables. Lab testing revealed that this extension cable can cause losses in the signal, resulting in a corrupted image being sent to the microcontroller

board. After experimenting with several different connector cables, we determined that the cable needs to be no more than 1-2 inches long to ensure a good quality connection. This required placing the CMUcam microcontroller board in the nose compartment as well. This step also involved lengthening the servo cables, but as the servo cables carry only an analog PWM signal, they are less susceptible to noise losses. While making assembly slightly more difficult, this new design approach helps to ensure proper operation of the camera during pool tests.

The research and selection of electronic components to power the vehicle was one of balancing computational power with simplicity, ease of use, and power consumption. Early designs planned on a full featured vehicle with a suite of sensors, all communicating through a TattleTale microcontroller, which afforded multiple communications channels and data logging capability on a CompactFlash memory card. However, it soon became clear that the TattleTale was both physically too large to be accommodated inside the vehicle and too difficult to write effective control code for. Development has largely ceased on it as a platform, so there are few peripherals that communicate well with it. We chose to use the CMUcam alone for processing and data storage in early missions. While not having the same capabilities as the TattleTale, it has a much simpler interface, and could accomplish our control goals on its own without needing to communicate with external peripherals. The Arduino microcontroller is a promising platform once the project's complexity outgrows the CMUcam. It has an open community of hobby robotics enthusiasts, and development is ongoing, leading to more powerful products being developed regularly.

5.2.3 Pool Testing Analysis

As discussed in section 4.7, the prototype vehicle is currently undergoing testing at the Alumni pool facility on the MIT campus. Early tests have been successful, as the vehicle is able to track the target from a variety of locations at the surface, and is stable enough to keep a lock on the target during the descent.

As discussed earlier, I originally intended to implement a camera algorithm that tracked a dark target on a light background. This seemed a logical choice, since

the majority of the pool surface was white tile, and early lab experiments showed the camera was effective at tracking black spots on white sheets of paper. However, we soon learned during pool tests that varying lighting conditions caused extremely erratic measurements, often resulting in the camera losing the target entirely. Several solutions were tried, including placing white sheets of plastic on the pool bottom to provide a homogeneous background for the target, as well as operating in shallower waters. After these attempts produced mixed results, we decided to alter the camera algorithm to track a flashlight placed at the bottom of the pool. It was a relatively simple software change, but greatly increased the effectiveness of the algorithm, as well its robustness to changing lighting conditions and objects on the pool bottom. With this new algorithm, the vehicle is currently capable of hitting the target consistently from a variety of starting locations. Further testing is planned, to include upgraded sensors and a more powerful microcontroller based on the Arduino.

5.3 Future Work

The vehicle has performed well in initial tests, but it should be possible to tune the performance of the vehicle and record more complete data. Moving forward, we have set out several goals for improving the capabilities of the prototype vehicle, including improved sensors and electronics, mechanical design changes, and more challenging mission profiles.

5.3.1 Short Term Goals

The next immediate goal for the project is to begin adding components to the prototype vehicle to improve its sensing and processing capability. These new components include:

- Compass, roll, and pitch sensor: these sensors are integrated into a single device, which enables the compass measurements to be compensated for orientation. Additional inertial sensors can provide information about acceleration in any

direction.

- Depth sensor: a simple piezoelectric pressure transducer, this sensor would be mounted on the hull of the vehicle. It enables measurements of the vehicle's depth in the water, and thus can aid in estimating the vehicle's distance from the target.

- Arduino Microcontroller: The Arduino is an open-source microcontroller platform popular with hobby robotics. The board provides capability for analog sensors, multiple serial communications channels, wireless programming, and data logging.

- Wireless communications: the ZigBee protocol is a low power wireless protocol that allows communication with serial port enabled microcontrollers over a range of distances. By installing a wireless chip such as the ZigBee, we could remotely program the vehicle and retrieve data without disassembling the vehicle.

5.3.2 Mid-Term Goals

In the mid term, there are several upgrades to the current generation vehicle that can be implemented to more closely model the behavior of the full vehicle and its associated USBL navigation system. First and foremost, the move from a vehicle mounted camera to a surface mounted camera would allow us to have a navigation system whose performance mimicked the USBL, in that accuracy deteriorates with increased depth, as opposed to improving at the vehicle gets closer to the target. With a surface mounted camera system, tests can still be conducted in a pool environment, but enable quicker computations and more advanced processing since the controller no longer needs to be contained in the vehicle.

Another design change that could be implemented on the current vehicle are structures to improve the lift characteristics of the vehicle, to maximize the horizontal range it can travel effectively. One possible method for achieving this is the addition of

mid-body or bow mounted wings, to improve lift. However, the addition of these wings would alter the hydrodynamic behavior of the vehicle, possibly reducing stability, so care must be taken in the design. It could also be possible to make these wings retractable, to enable the vehicle to operate in two modes, depending on mission parameters.

5.3.3 Long Term Goals

In the longer term, we hope to develop the vehicle into a full featured, ocean capable platform for subsea equipment delivery. This would involve significant upgrades to the electronics and processing capabilities of the vehicle, as well as a major redesign of the mechanical structure. A full scale ocean vehicle would need an acoustic navigation and communications system to allow for the kind of control described in this paper. The mechanical structure of the vehicle would also need to be configured to allow for attachment to a sensor or equipment platform, preferably utilizing existing connection points. This would potentially require a shift away from a slender streamlined body to one that could integrate a variety of payloads. An important component of this redesign would be the development of a complete model of the vehicle dynamics, including the effects of various payloads on system behavior. This would not only aid in designing effective control systems, but also in choosing the size and shape of the vehicle to accommodate payloads most likely to be used.

The vertical glider platform could also be extended to the concept of multiple deployment missions. By utilizing guided vehicles, a surface vessel can deploy many vehicles in succession, allowing each to glide down to its destination. By allowing the surface vessel to remain in one spot, this method could significantly reduce mission times and costs for large scale sensor deployment.

5.4 Final Thoughts

The vertical glider prototype and computer simulations described in this thesis are a first step towards developing a robust solution to the challenge of subsea equipment

delivery. We have demonstrated the feasibility of an unpowered vehicle guiding towards a fixed target, and explored through computer simulation the parameters that have an affect on the performance of such a vehicle. These results will hopefully guide the next stage of development of vertical gliders.

Appendix A

Additional Results

A.1 Simulation Results

A.1.1 Basic Kinematic Model Results Tables

	$\sigma_w = 0.00$	$\sigma_w = 0.10$	$\sigma_w = 0.15$	$\sigma_w = 0.20$	$\sigma_w = 0.25$
$\sigma_\alpha = 0.0°$	0.00	3.94	4.70	7.48	9.49
$\sigma_\alpha = 0.4°$	0.55	3.30	5.20	6.76	9.88
$\sigma_\alpha = 0.8°$	1.11	3.77	5.24	7.28	9.07
$\sigma_\alpha = 1.2°$	1.86	3.70	5.66	6.91	10.40
$\sigma_\alpha = 1.6°$	2.24	4.45	5.46	7.53	8.59

Table A.1: Basic Kinematic Model Vehicle Performance, K = 1, values correspond to standard deviation of landing error over 100 trials, units of meters

	$\sigma_w = 0.00$	$\sigma_w = 0.10$	$\sigma_w = 0.15$	$\sigma_w = 0.20$	$\sigma_w = 0.25$
$\sigma_\alpha = 0.0°$	0.00	2.97	4.63	5.86	7.23
$\sigma_\alpha = 0.4°$	0.85	2.81	4.19	6.02	6.81
$\sigma_\alpha = 0.8°$	1.67	3.43	4.16	5.72	6.40
$\sigma_\alpha = 1.2°$	2.57	4.47	5.39	6.25	7.82
$\sigma_\alpha = 1.6°$	3.38	4.79	6.10	6.23	7.51

Table A.2: Basic Kinematic Model Vehicle Performance, K = 2, values correspond to standard deviation of landing error over 100 trials, units of meters

	$\sigma_w = 0.00$	$\sigma_w = 0.10$	$\sigma_w = 0.15$	$\sigma_w = 0.20$	$\sigma_w = 0.25$
$\sigma_\alpha = 0.0°$	0.00	2.01	3.26	4.46	5.50
$\sigma_\alpha = 0.4°$	1.22	2.72	3.25	4.62	5.12
$\sigma_\alpha = 0.8°$	2.62	3.59	4.07	5.00	5.57
$\sigma_\alpha = 1.2°$	4.41	4.56	4.77	5.72	6.75
$\sigma_\alpha = 1.6°$	5.54	6.40	7.00	7.09	7.59

Table A.3: Basic Kinematic Model Vehicle Performance, K = 4, values correspond to standard deviation of landing error over 100 trials, units of meters

	$\sigma_w = 0.00$	$\sigma_w = 0.10$	$\sigma_w = 0.15$	$\sigma_w = 0.20$	$\sigma_w = 0.25$
$\sigma_\alpha = 0.0°$	0.00	1.50	2.33	3.02	4.08
$\sigma_\alpha = 0.4°$	1.94	2.72	2.91	3.18	3.78
$\sigma_\alpha = 0.8°$	3.60	4.14	4.63	5.35	6.13
$\sigma_\alpha = 1.2°$	5.67	6.45	6.59	7.42	7.73
$\sigma_\alpha = 1.6°$	7.55	7.70	8.89	8.87	8.08

Table A.4: Basic Kinematic Model Vehicle Performance, K = 8, values correspond to standard deviation of landing error over 100 trials, units of meters

A.1.2 Extended Kinematic Model Results Tables

	$\sigma_w = 0.00$	$\sigma_w = 0.10$	$\sigma_w = 0.15$	$\sigma_w = 0.20$	$\sigma_w = 0.25$
$\sigma_\alpha = 0.0°$	0.00	0.52	1.01	1.54	2.15
$\sigma_\alpha = 0.4°$	3.46	4.08	3.88	3.99	4.19
$\sigma_\alpha = 0.8°$	5.48	5.77	5.85	6.47	6.47
$\sigma_\alpha = 1.2°$	6.18	6.36	7.06	6.83	8.31
$\sigma_\alpha = 1.6°$	10.62	8.44	10.00	10.19	9.28

Table A.5: Extended Kinematic Model Vehicle Performance, K = 1, values correspond to standard deviation of landing error over 100 trials, units of meters

	$\sigma_w = 0.00$	$\sigma_w = 0.10$	$\sigma_w = 0.15$	$\sigma_w = 0.20$	$\sigma_w = 0.25$
$\sigma_\alpha = 0.0°$	0.00	0.83	1.90	2.93	2.81
$\sigma_\alpha = 0.4°$	3.03	2.58	3.44	4.04	4.45
$\sigma_\alpha = 0.8°$	4.45	4.70	4.76	5.24	5.70
$\sigma_\alpha = 1.2°$	5.08	5.31	6.07	6.87	6.06
$\sigma_\alpha = 1.6°$	7.29	8.00	7.71	8.06	8.27

Table A.6: Extended Kinematic Model Vehicle Performance, K = 2, values correspond to standard deviation of landing error over 100 trials, units of meters

	$\sigma_w = 0.00$	$\sigma_w = 0.10$	$\sigma_w = 0.15$	$\sigma_w = 0.20$	$\sigma_w = 0.25$
$\sigma_\alpha = 0.0°$	0.00	1.26	2.47	3.99	6.27
$\sigma_\alpha = 0.4°$	2.16	2.51	3.50	4.95	6.17
$\sigma_\alpha = 0.8°$	3.29	3.35	4.74	5.17	7.13
$\sigma_\alpha = 1.2°$	4.02	4.62	5.11	5.75	7.11
$\sigma_\alpha = 1.6°$	5.62	6.38	6.29	6.94	7.73

Table A.7: Extended Kinematic Model Vehicle Performance, K = 4, values correspond to standard deviation of landing error over 100 trials, units of meters

	$\sigma_w = 0.00$	$\sigma_w = 0.10$	$\sigma_w = 0.15$	$\sigma_w = 0.20$	$\sigma_w = 0.25$
$\sigma_\alpha = 0.0°$	0.00	1.98	3.93	5.76	8.32
$\sigma_\alpha = 0.4°$	1.43	2.79	4.56	5.37	7.84
$\sigma_\alpha = 0.8°$	2.56	3.07	4.85	6.35	8.18
$\sigma_\alpha = 1.2°$	3.13	3.58	3.92	6.16	9.31
$\sigma_\alpha = 1.6°$	3.61	4.29	5.57	7.15	8.15

Table A.8: Extended Kinematic Model Vehicle Performance, K = 8, values correspond to standard deviation of landing error over 100 trials, units of meters

A.2 Pool Testing Results

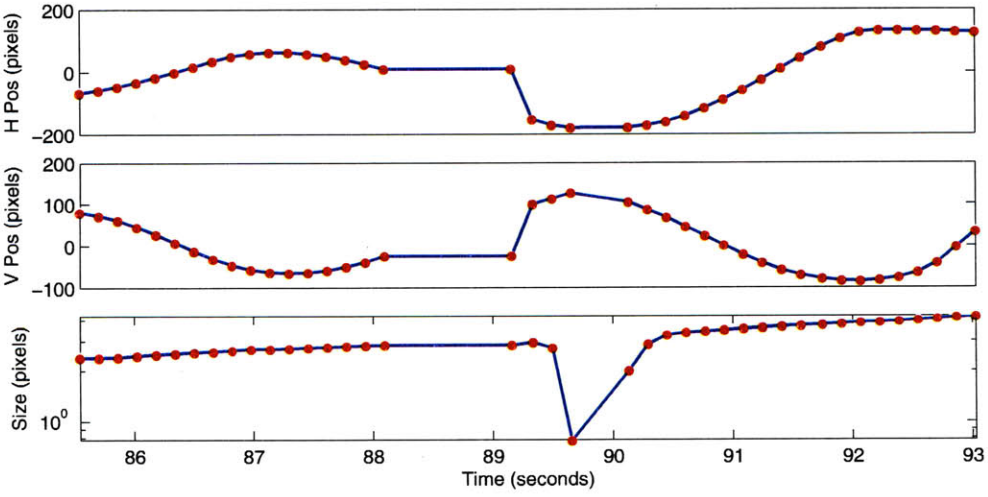

Figure A-1: Mission 4: Plot of X position, Y position, and target area vs time; red dots denote discrete data points, time is printed on log scale

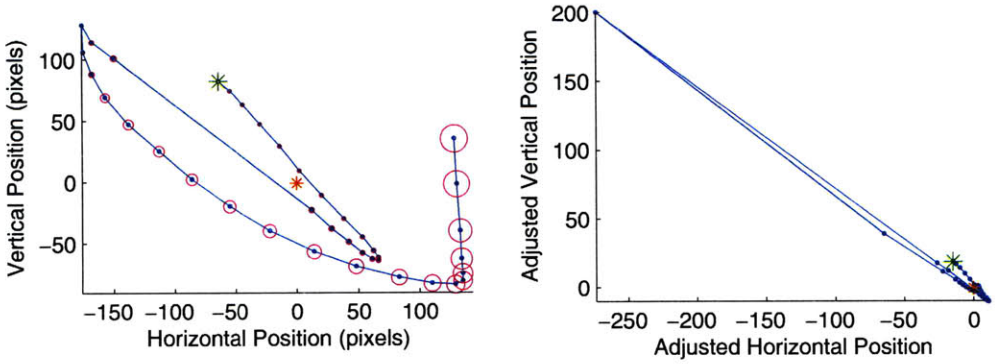

Figure A-2: Mission 4: Scatter plot of X vs. Y position, magenta circles show target area, growing progressively larger as vehicle nears the target, red star shows field of view center, green star shows target starting location

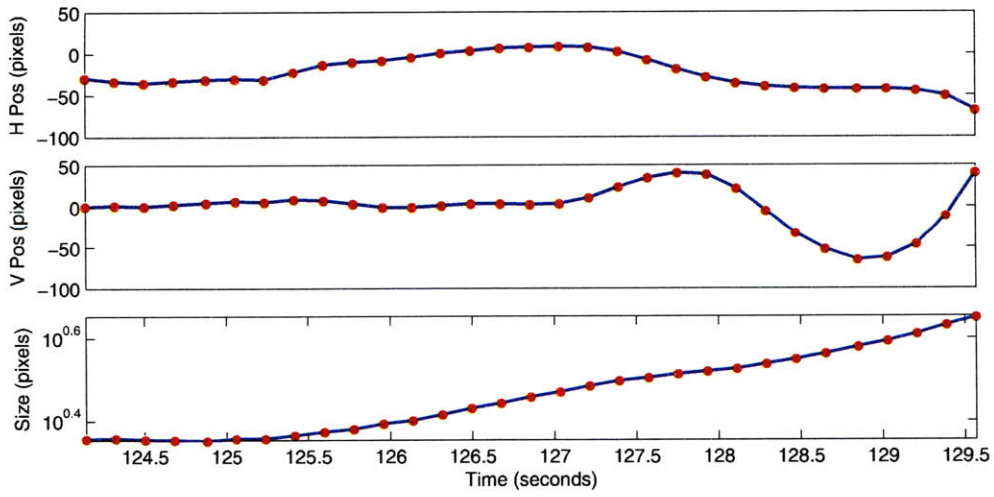

Figure A-3: Mission 5: Plot of X position, Y position, and target area vs time; red dots denote discrete data points, time is printed on log scale

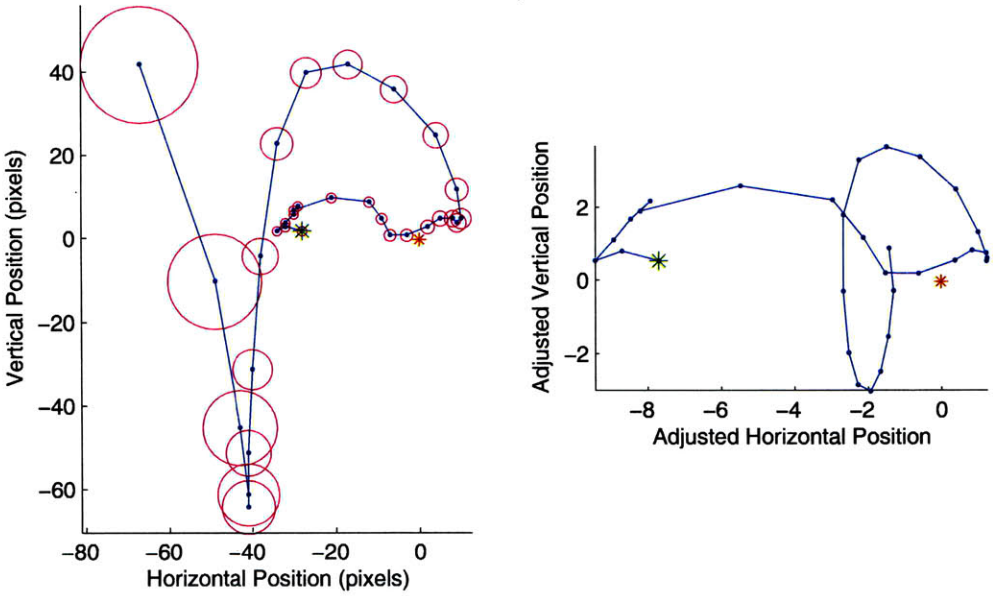

Figure A-4: Mission 5: Scatter plot of X vs. Y position, magenta circles show target area, growing progressively larger as vehicle nears the target, red star shows field of view center, green star shows target starting location

Appendix B

Source Code

Listing B.1: CMUcam Source Code, Camera Only

```
/* CSEM control software, only uses CMUcam, with JPEG save*/

// algorithm internal parameters
#define X_CENTER 176 // center of the CMUCAM picture, X axis
#define Y_CENTER 144 // center of the CMUCAM picture, Y axis

// Servo Calibration
#define X_SERVO_MIDDLE 135 // X Servo middle position
#define Y_SERVO_MIDDLE 128 // Y Servo middle position

// Control parameters
#define GAIN_X 1  // gain of the proportionnal controller, axis X
#define GAIN_Y 1  // gain of the proportionnal controller, axis Y
#define X_SERVO_STEP 90 // saturation boundaries for X_servo
#define Y_SERVO_STEP 90 // saturation boundaries for Y_servo

// Compass Parameters
#define COMPASS_SIZ 20

// control parameters
#define MIN_WHITE_INTESITY 210   // brightness a pixel must have to be taken into account by the algorithm
#define IMG_INTERVAL 10000    // Time to wait in milliseconds in between saving an image to memory card

void set_target(int* p_x, int* p_y, long* p_coef, int* p_count); // acquires target
void set_target_save(int* p_x, int* p_y, long* p_coef, int* p_file); // acquires target and saves image
void compute_servos(int x_error, int y_error, int* p_x_servo, int* p_y_servo); // computes servo commands from target position

int main (void)
{
   // VARIABLES DECLATATION
   int x_f; // X target position
   int y_f; // Y target position
   long coef = 0; // target size coefficient
   int count; // count of dark or light pixels
   int x_servo = X_SERVO_MIDDLE; // Initializes Rudder servo to center position
   int y_servo = Y_SERVO_MIDDLE; // Initializes Elevator servo to center position
   int imgsave = 0;
   int fcount;
```

```
    // TEST : length of the process
    int start_time; // Start time of the program in milliseconds
    int last_time; // Last recorded time in milliseconds

    last_time = cc3_timer_get_current_ms(); // Initializes variable to current CC3
        timer
    start_time = cc3_timer_get_current_ms(); // Initializes variable to current CC3
        timer

    while (1)
    {
       if((cc3_timer_get_current_ms() - last_time) > IMG_INTERVAL)  //test for wait
           interval, save JPEG snapshot
       {
          set_target_save(&x_f,&y_f,&coef,&fcount); // Acquire target position and
              save
          printf("img_save\n"); // Note that an image was saved
          last_time = cc3_timer_get_current_ms(); // Note the time, for next wait
              interval
          imgsave = 1;
       }
       else
       {
          set_target(&x_f,&y_f,&coef,&count); // Acquire target position
          imgsave = 0;
       }

       if(coef == 0) // meaning : if no black pixel is found
       {
          printf("No Target...\n");     // this is the error message sent
          cc3_led_set_state (2,true);      // lights the second LED on the CMUCam
       }

       else  // meaning : there is at least one black pixel found
       {
          compute_servos(x_f, y_f, &x_servo, &y_servo); // Computer servo commands
          printf("(%d,%d);(%d,%d),%d\n",x_f,y_f,x_servo,y_servo,count); // Print info
              to command line
       }

       cc3_gpio_set_servo_position(0,x_servo); // Send command to rudder servo
       cc3_gpio_set_servo_position(1,y_servo); // Send command to elevator servo
    }
    return (0);
}

void set_target(int* p_x, int* p_y, long* p_coef, int* p_count)
{
    cc3_image_t img;
    int y = 0;
    long coef = 0;
    long sum_x = 0;
    long sum_y = 0;
    int xc;
    int yc;
    int counter = 0;

    cc3_pixbuf_load ();  // Take a picture with the camera and load it into the
        internal pixbuf.
    img.channels = 1; // we use one single channel
    img.width = cc3_g_pixbuf_frame.width; // set frame width
    img.height = 1;    // image will hold just 1 row for scanline processing
    img.pix = cc3_malloc_rows (1);

    // This tells the camera to grab a new frame into the fifo and reset
    // any internal location information.
    cc3_pixbuf_frame_set_coi(CC3_CHANNEL_SINGLE);
```

```
100     while (cc3_pixbuf_read_rows (img.pix, 1)) // read a row into the image picture
            memory from the camera
        {
           for (uint16_t x = 0; x < img.width; x++) // Cycle through pixels in the row
           {
              uint8_t white = ((uint8_t *) img.pix)[x]; // record the brightness value
105           if (white > MIN_WHITE_INTENSITY) { // if brightnees is higher than threshold
                sum_x += x; // sum of recorded x positions
                   sum_y += y; // sum of recorded y positions
                   coef += 1; // coefficent sum
                counter++; // count of recorded pixels
110           }
           }
           y++;
        }

115     xc = (sum_x/coef) - X_CENTER; // average of x positions, adjusted to center
        yc = (sum_y/coef) - Y_CENTER; // average of y positions, adjusted to center
        *p_coef = coef;

        *p_x = xc ;
120     *p_y = yc;
        *p_count = counter;

        free (img.pix);
    }
125
    void compute_servos(int x_error, int y_error, int* p_x_servo, int* p_y_servo)
    {
        // Variable declaration
        int x_servo_simple;  // expected X_servo position without saturation limits
130     int y_servo_simple;     // expected Y_servo position without saturation limits
        int x_servo;         // expected X_servo position WITH saturation limits
        int y_servo;         // expected Y_servo position WITH saturation limits

        // sets X_servo position
135     x_servo_simple = GAIN_X*x_error;

        if (x_servo_simple <= X_SERVO_STEP && x_servo_simple >= -X_SERVO_STEP) // ensure
            servo command is within bounds
        {
           x_servo = X_SERVO_MIDDLE+x_servo_simple;
140     }
        else if (x_servo_simple < X_SERVO_STEP) // ensure servo command is within bounds
        {
           x_servo = X_SERVO_MIDDLE-X_SERVO_STEP;
        }
145     else  // ensure servo command is within bounds
        {
           x_servo =X_SERVO_MIDDLE+X_SERVO_STEP;
        }

150     // sets Y_servo position
        y_servo_simple = GAIN_Y*y_error;

        if (y_servo_simple <= Y_SERVO_STEP && y_servo_simple >= -Y_SERVO_STEP) // ensure
            servo command is within bounds
        {
155        y_servo = Y_SERVO_MIDDLE+y_servo_simple;
        }
        else if (y_servo_simple < Y_SERVO_STEP) // ensure servo command is within bounds
        {
           y_servo = Y_SERVO_MIDDLE-Y_SERVO_STEP;
160     }
        else // ensure servo command is within bounds
        {
           y_servo =Y_SERVO_MIDDLE+Y_SERVO_STEP;
```

```
        }

        // sets values
        *p_x_servo = x_servo;
        *p_y_servo = y_servo;
    }
```

Listing B.2: MATLAB Simulation Sample Code

```
% Sample MATLAB Code, Common Parameters

trials = 100;    % Number of trials to conduct
dt = 1;          % Simulation time step, in seconds
D = 4000;        % Max depth of mission, in meters
cs = 1500;       % Speed of sound in water,
Sa = [];         % Measurement Error StdDev
Sw = [];         % Process Error StdDev
K = [];          % Control Gain

% Basic Kinematic Model Parameters

z_dot = -1;      % Vertical rate of descent, in meters per second
x = [];          % X position
x_dot = [];      % X velocity
z = [];          % Height from bottom
t = [];          % Elapsed time in seconds
alpha = [];      % measurement
i=1;

% Initial Conditions
x(i) = 0;        % Vehicle starts directly over target
z(i) = D;        % Initial height from bottom
t(i) = 0;        % Initial time
alpha(i)=0;      % Initial angle measurement

nUpdate = 0.0001; % Initialize the update delay so that the vehicle will take a
    measurement

while (z(i)>0)
    t(i+1) = t(i) + dt; % increment time by dt
    z(i+1) = z(i) + z_dot*dt; % increment vertical position by z_dot*dt
    x(i+1) = x(i) + K*alpha(i)*z_dot*dt + Sw*randn*sqrt(dt); % move sideways
        according to measurement alpha and noise Sw

    if (nUpdate>t(i)) % if delay for new position update has not passed
        alpha(i+1)=alpha(i); % keep measurement the same
    else % if delay for new position update has passed
        nUpdate = t(i)+2*(D-z(i))/cs; % reset delay according to two-way travel time
            to surface
        alpha(i+1) = atan2(x(i),D-z(i))+ Sa*randn; % take new measurement
    end

    i=i+1;
end

% Extended Kinematic Model Parameters

tau = 26.6;              % Filter time constant in seconds
Kf = dt/(tau + dt);      % Filter gain
Xmax = .5;               % Max horizontal velocity
a=2/-.81;                %
b=-1.1*a;                % Parameters for velocity relationship
c=.1*a;                  %

x = [];          % x position
x_dot = [];      % x velocity
z = [];          % height from bottom
z_dot = [];      % dive rate
t = [];          % elapsed time in seconds
alpha = [];      % measurement
alpha_f = [];    % filtered measurement

i=1;

% Initial Conditions
```

```
x(i) = 0;        % Initial x position, directly over target
x_dot(i) = 0;    % Zero initial x velocity
z(i) = D;        % Initial height from bottom
z_dot(i) = 1;    % Initial dive rate of 1 m/s
t(i) = 0;        % Initial time
alpha(i)=0;      % Initial measurement
alpha_f(i)=0;    % Initial measurement estimate

nUpdate = 0.0001;

while (z(i)>0)
    t(i+1) = t(i) + dt; % increment time by dt
    x_dot(i+1) = -K(1)*alpha_f(i); % intended x velocity
    x(i+1) = x(i) - K(1)*alpha_f(i)*dt; % intended increment of x

    if (x_dot(i+1) > Xmax) % check to see if x_dot is over the maximum
        x_dot(i+1) = Xmax;
        x(i+1) = x(i) + Xmax*dt;
    elseif (x_dot(i+1) < -Xmax) % check the negative of x_dot as well
        x_dot(i+1) = -Xmax;
        x(i+1) = x(i) - Xmax*dt;
    end

    % z_dot below has a quadratic relationship with x_dot
    z_dot(i+1) = -(b + (b^2 - 4*a*c + 4*a*abs(x_dot(i+1)))^(1/2))/(2*a);

    rw = randn; % normally distributed random number
    x_dot(i+1) = x_dot(i+1) + Sw*rw; % add process noise to x_dot
    x(i+1) = x(i+1) + Sw*rw*sqrt(dt); % increment x with process noise

    z(i+1) = z(i) - z_dot(i)*dt; % increment z position

    if (nUpdate>t(i)) % check to see if update interval has passed
        alpha(i+1) = alpha(i); % keep measurement the same as previous
        alpha_f(i+1) = (1-Kf)*alpha_f(i) + Kf*alpha(i); % low-pass filter calculation
    else % if its time for an update
        nUpdate = t(i)+2*(D-z(i))/cs; % reset update interval
        alpha(i+1) = atan2(x(i),D-z(i))+ Sa*randn; % take new measurement
        alpha_f(i+1) = (1-Kf)*alpha_f(i) + Kf*alpha(i); % low-pass filter calculation
    end

    i=i+1;
end
```

Appendix C

CAD Drawings

Figures

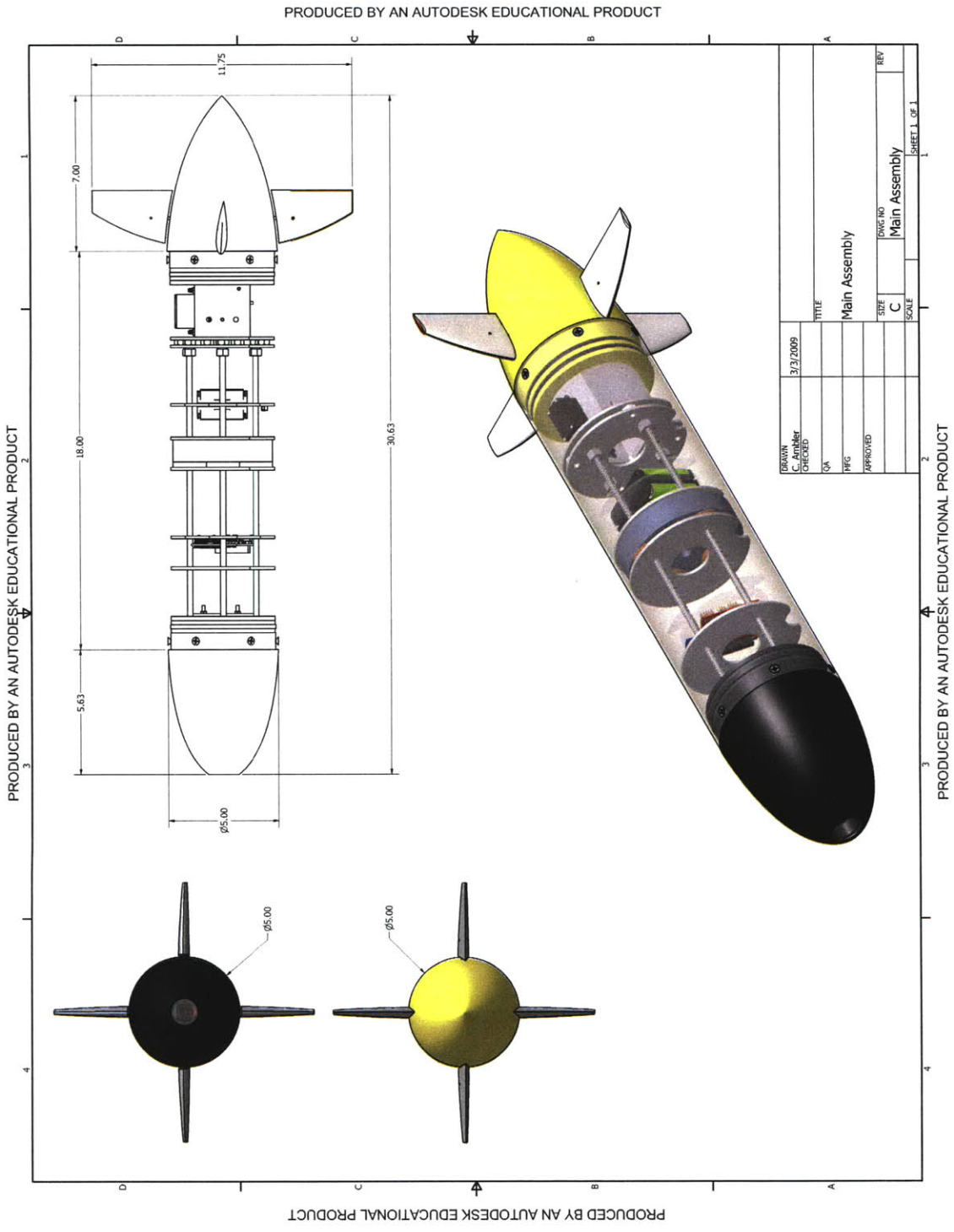

Figure C-1: Main Vehicle Assembly

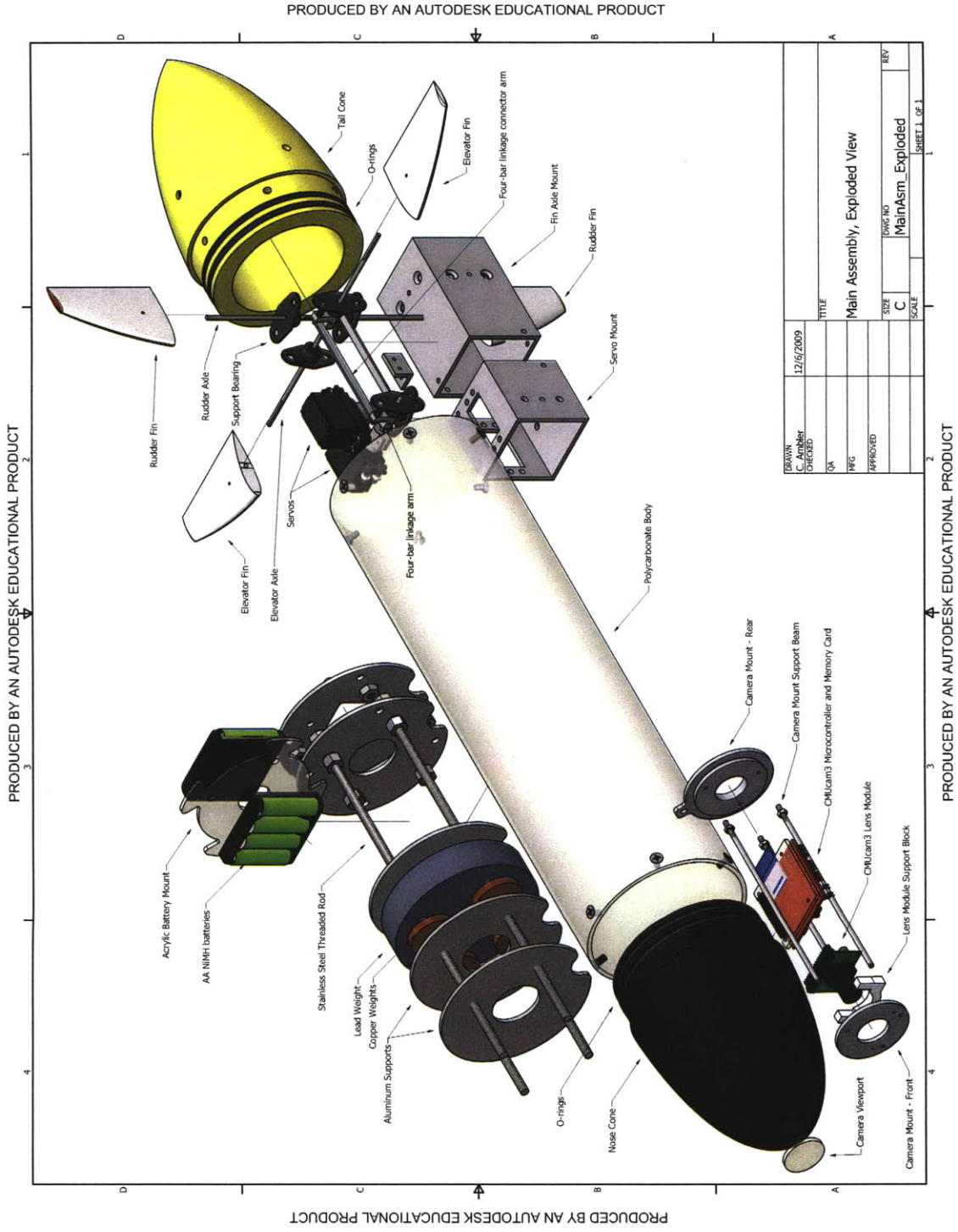

Figure C-2: Main Vehicle Assembly, Exploded View

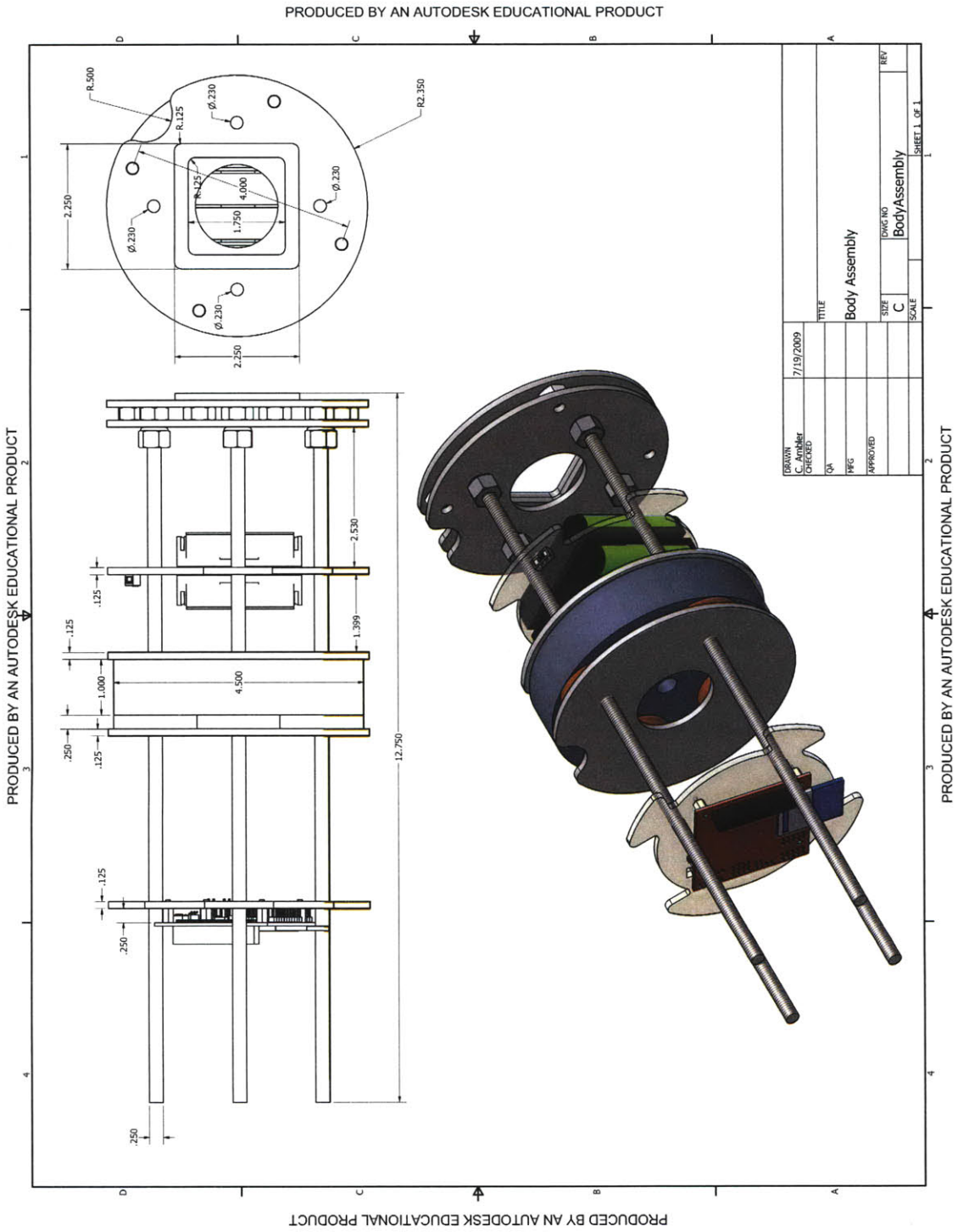

Figure C-3: Body Assembly

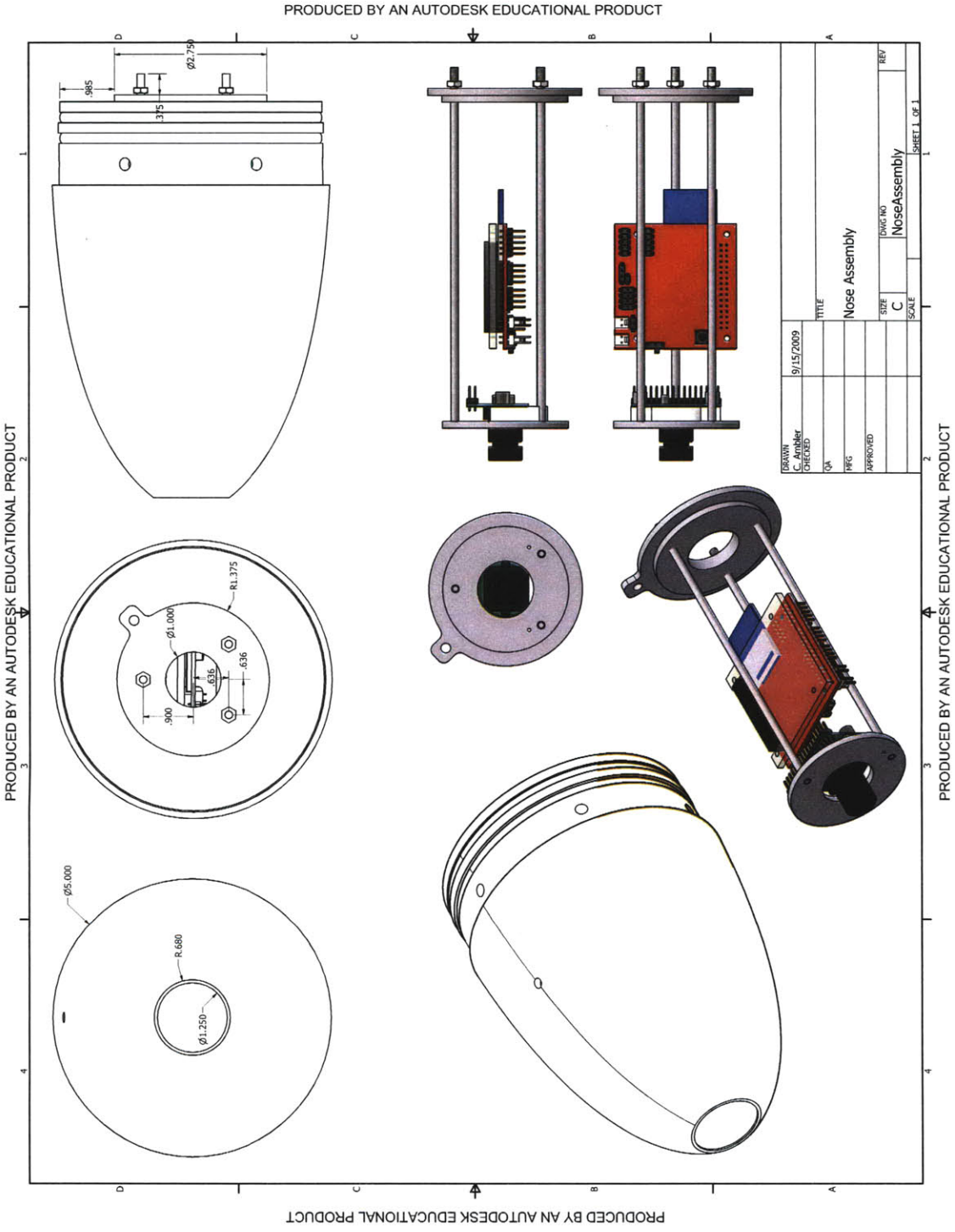

Figure C-4: Nose and Camera Assemblies

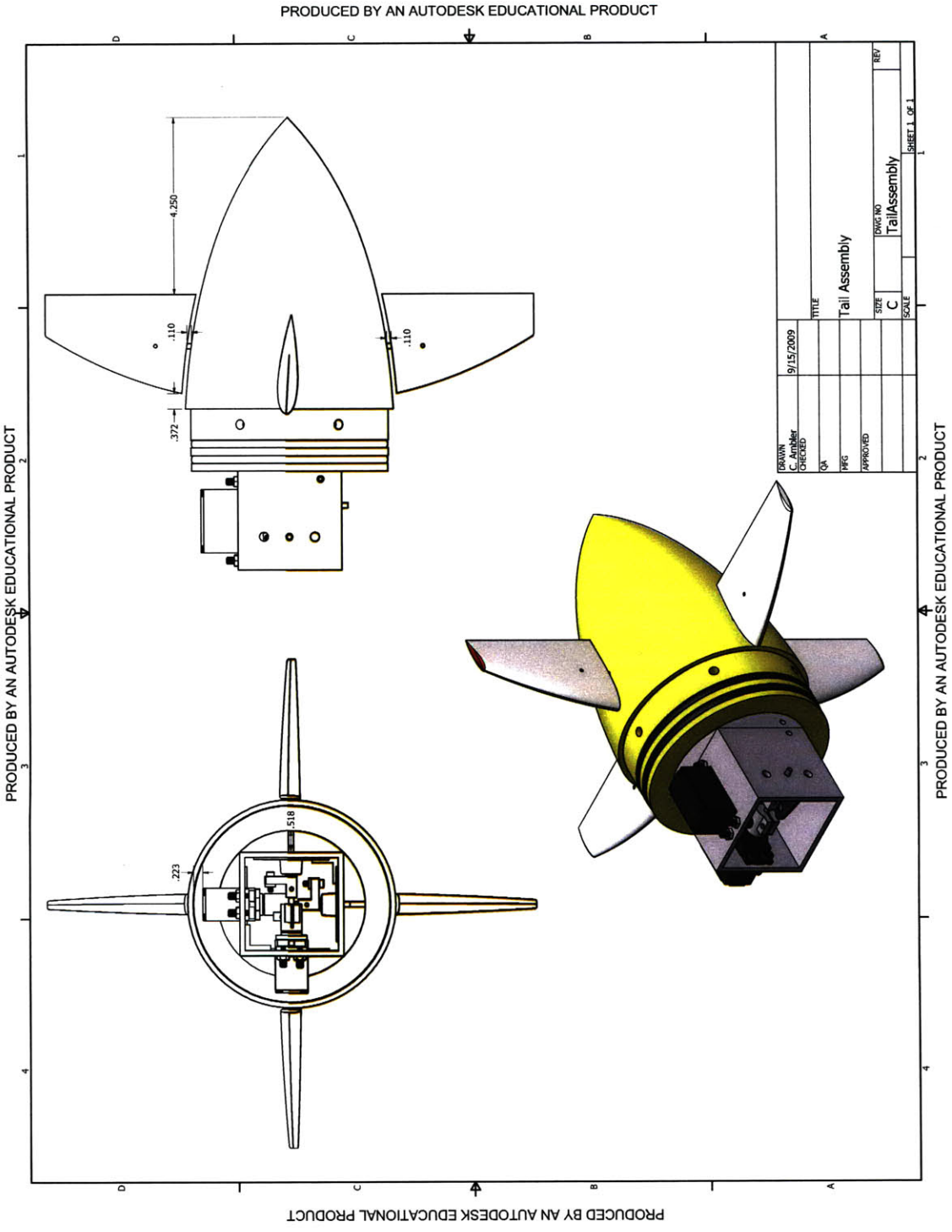

Figure C-5: Tail Assembly

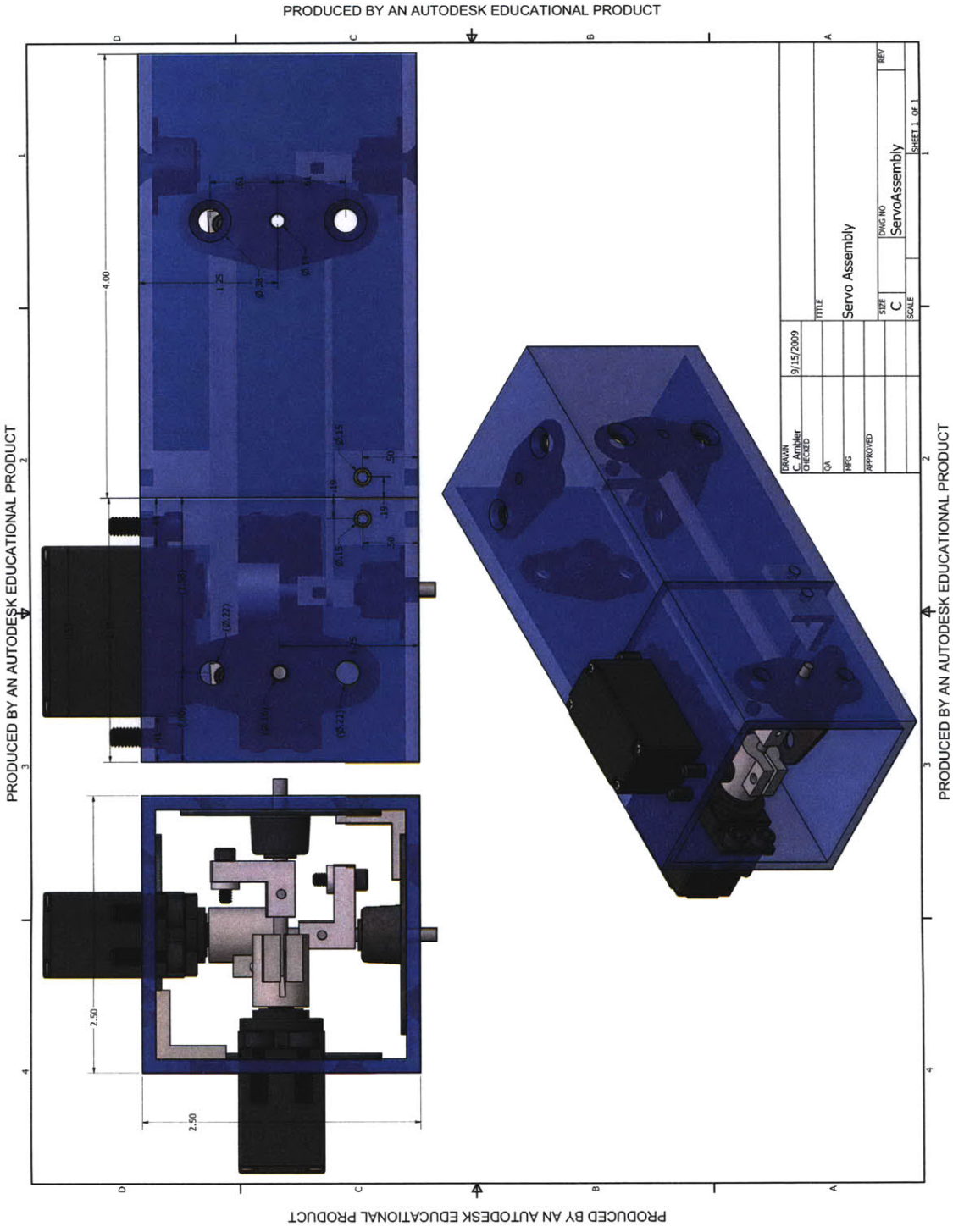

Figure C-6: Servo Assembly

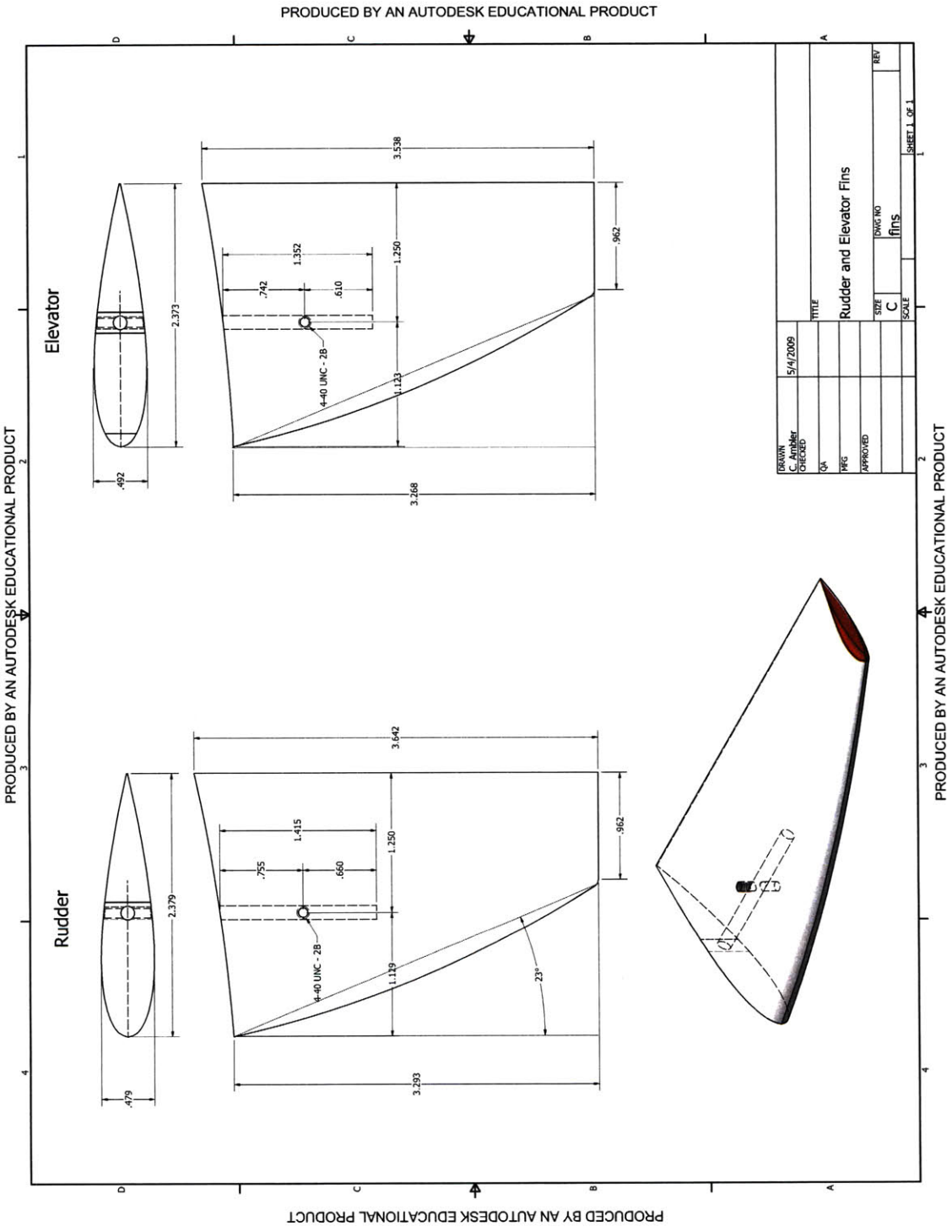

Figure C-7: Rudder and Elevator Fins

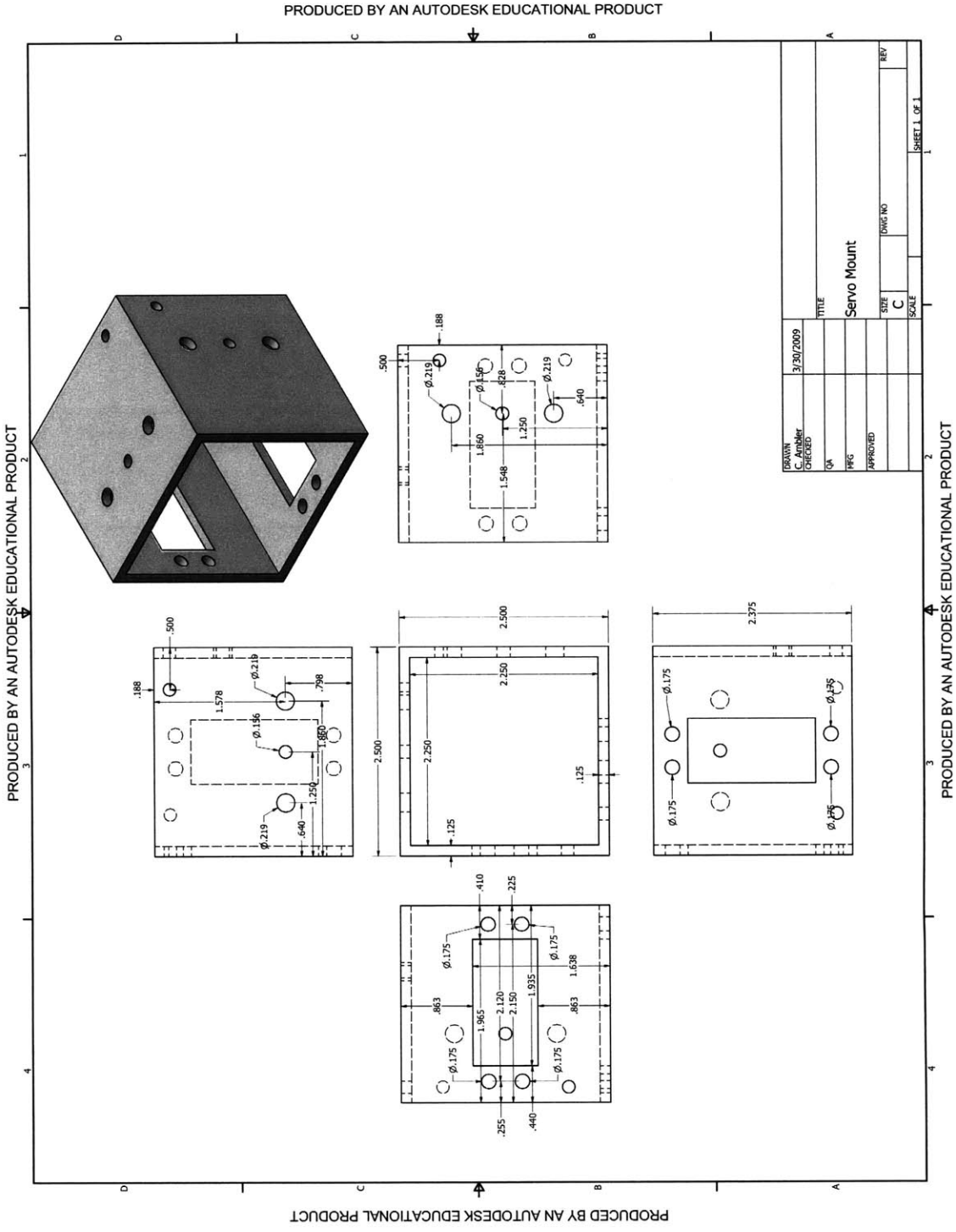

Figure C-8: Servo Mount

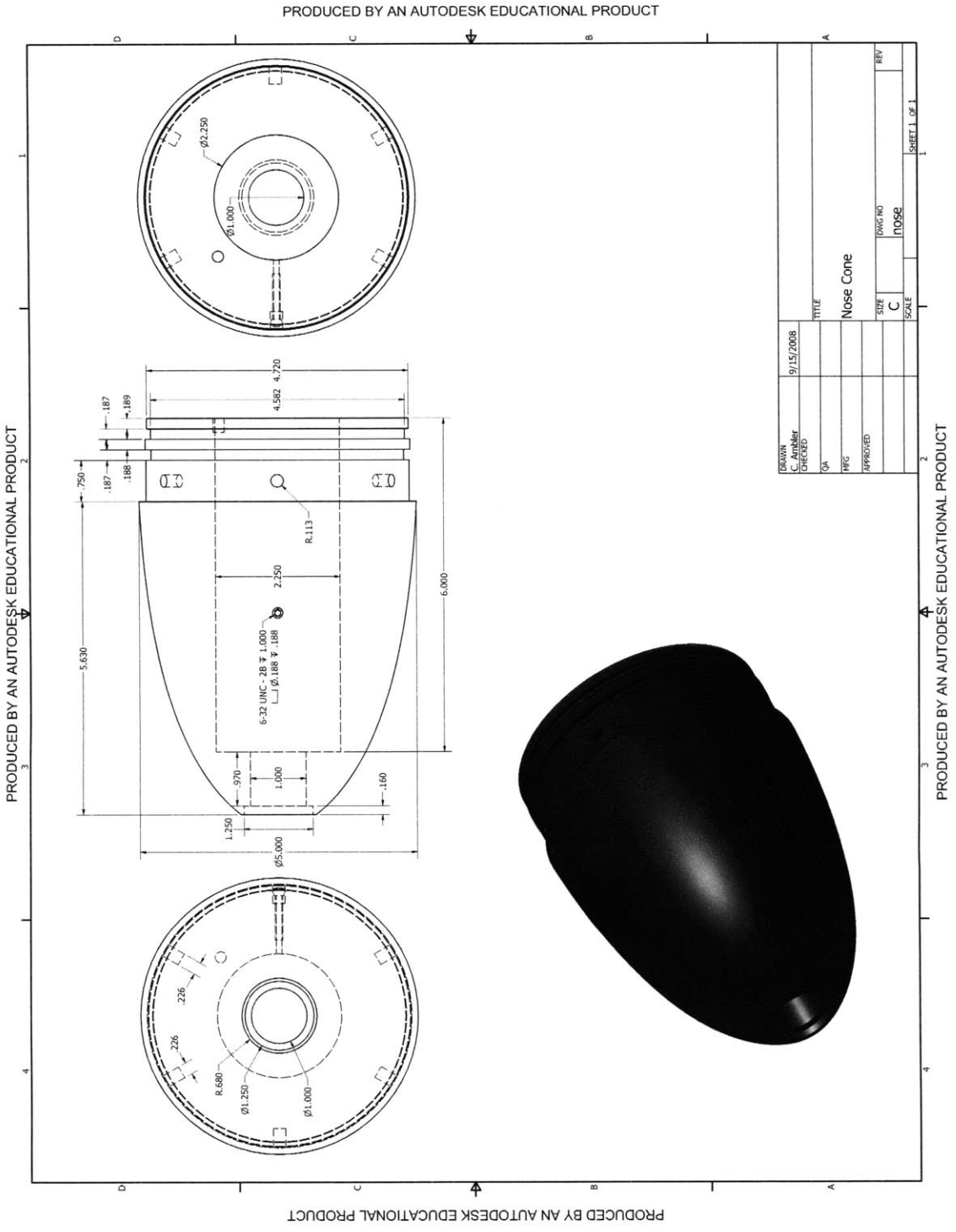

Figure C-9: Nose Cone

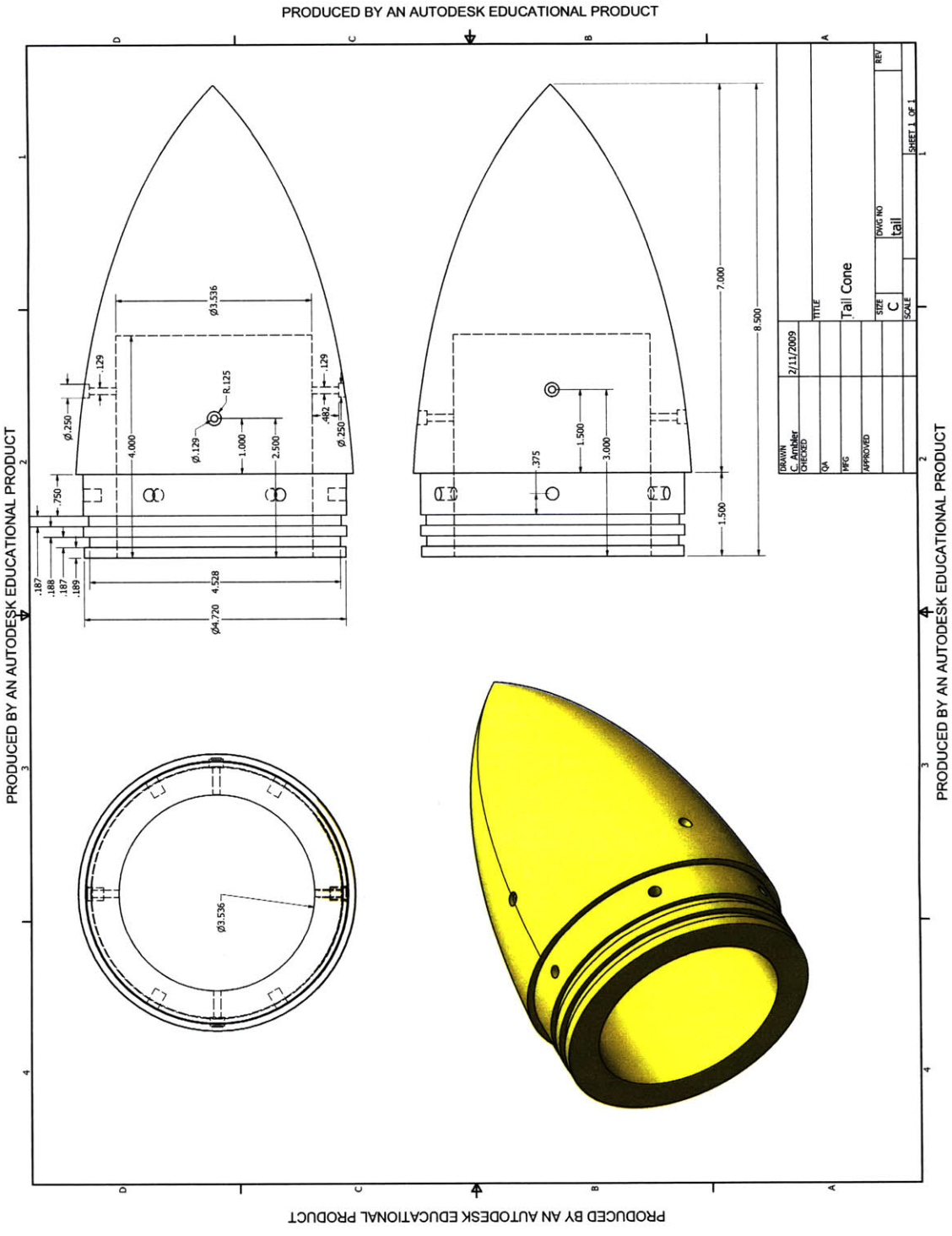

Figure C-10: Tail Cone

Bibliography

[1] IXSEA subsea positioning systems. http://www.ixsea.com/en/subsea_positioning/.

[2] Kongsberg HiPAP acoustic underwater positioning and navigation systems. http://www.km.kongsberg.com.

[3] CMUcam website. http://www.cmucam.org, 2009.

[4] I.H. Abbott and A.E. Von Doenhoff. *Theory of Wing Sections.* Dover Publications, 1959.

[5] GEP Box and M.E. Muller. A note on the generation of random normal deviates. *The Annals of Mathematical Statistics*, pages 610 611, 1958.

[6] J. Byron and R. Tyce. Designing a vertical / horizontal AUV for deep ocean sampling. In *Oceans 2007*, pages 1 10, 2007.

[7] C.C. Eriksen, T.J. Osse, R.D. Light, T. Wen, T.W. Lehman, P.L. Sabin, J.W. Ballard, and A.M. Chiodi. Seaglider: a long-range autonomous underwater vehicle for oceanographic research. *Oceanic Engineering, IEEE Journal of*, 26(4):424–436, Oct 2001.

[8] R.M. Eustice, L.L. Whitcomb, H. Singh, and M. Grund. Experimental results in synchronous-clock one-way-travel-time acoustic navigation for autonomous underwater vehicles. In *Proceedings of the IEEE International Conference on Robotics and Automation, Rome, Italy*, pages 4257–4264. Citeseer, 2007.

[9] James C. Kinsey, Ryan M. Eustice, and Louis L. Whitcomb. A survey of underwater vehicle navigation: Recent advances and new challenges. In *IFAC Conference of Manoeuvering and Control of Marine Craft*, Lisbon, Portugal, September 2006. Invited paper.

[10] P.E. Kloeden and E. Platen. *Numerical solution of stochastic differential equations.* Springer, 1992.

[11] E.V. Lewis. *Principles of Naval Architecture.* Society of Naval Architects and Marine Engineers, 1989.

[12] Lonny Lippsett and Amy E. Nevala. Nereus soars to the ocean's deepest trench. *Oceanus*, 2009. http://www.whoi.edu/oceanus/viewArticle.do?id=57606&sectionid=1000.

[13] L. Molnar, E. Omerdic, and D. Toal. Design of an advanced auv for deployment close to the seabed and other hazards. In *Oceans 2005 - Europe*, volume 2, pages 975–980 Vol. 2, 2005.

[14] O. Pfannkuche and P. Linke. GEOMAR landers as long-term deep-sea observatories. *Sea Technology*, 44(9):50–55, 2003.

[15] Timothy Prestero. Verification of a six-degree of freedom simulation model for the REMUS autonomous underwater vehicle. Master's thesis, Massachusetts Institute of Technology, 2001.

[16] I.G. Priede, S. Addison, S. Bradley, P.M. Bagley, P. Gray, C. Yau, J.-F. Rolin, J. Blandin, J. Legrand, A. Cremer, U. Witte, O. Pfannkuche, A. Tengberg, S. Hulth, W. Helder, and T. Van Weering. Autonomous deep-ocean lander vehicles; modular approaches to design and operation. In *OCEANS '98 Conference Proceedings*, volume 3, pages 1238 1244 vol.3, 1998.

[17] IG Priede, PM Bagley, A. Smith, S. Creasey, and NR Merrett. Scavenging deep demersal fishes of the Porcupine Seabight, North-east Atlantic: observations by baited camera, trap and trawl. *Journal of the Marine Biological Association of the UK*, 74(03):481 498, 2009.

[18] P. Rigby, O. Pizarro, and S.B. Williams. Towards Geo-Referenced AUV navigation through fusion of USBL and DVL measurements. In *OCEANS 2006*, pages 1 6, 2006.

[19] J. Sherman, RE Davis, WB Owens, and J. Valdes. The autonomous underwater glider "Spray". *IEEE Journal of Oceanic Engineering*, 26(4):437 446, 2001.

[20] H. Singh, J. Catipovic, R. Eastwood, L. Freitag, H. Henriksen, F. Hover, D. Yoerger, J.G. Bellingham, and B.A. Moran. An integrated approach to multiple auv communications, navigation and docking. In *OCEANS '96. MTS/IEEE. 'Prospects for the 21st Century'. Conference Proceedings*, volume 1, pages 59 64 vol.1, Sep 1996.

[21] M. Stojanovic. Recent advances in high-speed underwater acoustic communications. *Oceanic Engineering, IEEE Journal of*, 21(2):125 136, 1996.

[22] R. Stokey, M. Purcell, N. Forrester, T. Austin, R. Goldsborough, B. Allen, and C. von Alt. A docking system for remus, an autonomous underwater vehicle. In *OCEANS '97. MTS/IEEE Conference Proceedings*, volume 2, pages 1132–1136 vol.2, Oct 1997.

[23] K. Vickery. Acoustic positioning systems. a practical overview of current systems. In *Autonomous Underwater Vehicles, 1998. AUV'98. Proceedings Of The 1998 Workshop on*, pages 5–17, Aug 1998.

[24] D.C. Webb, P.J. Simonetti, and C.P. Jones. SLOCUM: an underwater glider propelled by environmental energy. *Oceanic Engineering, IEEE Journal of*, 26(4):447–452, Oct 2001.

[25] G. Welch and G. Bishop. An introduction to the Kalman filter. *University of North Carolina at Chapel Hill, Chapel Hill, NC*, 1995.

[26] L. L. Whitcomb, D. R. Yoerger, and H. Singh. Combined Doppler/LBL based navigation of underwater vehicles. In *Proceedings of the 11th International Symposium on Unmanned Untethered Submersible Technology*, 1999.

[27] L.L. Whitcomb. Underwater robotics: out of the research laboratory and into the field. In *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, volume 1, pages 709–716 vol.1, 2000.