FABRICATION AND AUTOMATION OF A POWER-CONSERVING USV IN MOVING WATER

by

Joseph Wichlinski

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Master of Science in Mechanical Engineering

School of Mechanical Engineering

West Lafayette, Indiana

August 2021

THE PURDUE UNIVERSITY GRADUATE SCHOOL

STATEMENT OF COMMITTEE APPROVAL

Dr. Eric Nauman, Chair

School of Mechanical Engineering

Dr. Charles Krousgrill

School of Mechanical Engineering

Dr. Peter Meckl

School of Mechanical Engineering

Approved by:

Dr. Nicole Key

Dedicated to my supportive family, friends, and colleagues

ACKNOWLEDGMENTS

I would like to thank committee members Dr. Eric Nauman, Dr. Charles Krousgrill, and Dr. Peter Meckl for their continuous support. Also, thank you to the Purdue Mechanical Engineering department for funding my graduate schoolwork. I also want to thank the HIRRT Lab, especially Dr. Taylor Lee, Michael Dziekan, Nick Leiva, Breana Cappuccilli, Justin Markel, Kevin McIver, and Jacob McGough, for helping me throughout many different stages of this project. Lastly, I want to thank my personal colleagues Michael Williams and David Faseur for providing access to helpful tools and resources needed to complete this project.

TABLE OF CONTENTS

LIST OF TABLES 6
LIST OF FIGURES 7
ABSTRACT 11
1. INTRODUCTION 12
2. METHODS 20
2.1 Preliminary Design 20
2.1.1 Thruster Specifications 21
2.1.2 Battery Specifications 22
2.2 USV Structure Fabrication 23
2.3 Electronics and Circuit Design 26
2.4 Sensors and Communication Systems 29
2.5 Summary of Final Prototype 31
2.6 Programming Automation Logic 36
2.7 Field Experiments 42
3. RESULTS 45
3.1 Lake Harner 45
3.2 Wabash River 47
4. CONCLUSION / FUTURE WORK 54
REFERENCES 56

LIST OF TABLES

Table 2.1. BlueRobotics T200 current draw and thrust output at 16V nominal voltage 22

LIST OF FIGURES

Figure 1.1. Gliding Robotic Fish diving underwater to collect water column. The prototype utilized only GPS to navigate [8]. 13

Figure 1.2. HydroNet ASV floating in Livorno harbor, Italy. The catamaran prototype included a GPS, compass, laser scanner, sonar, and altimeter sensors [9]. 14

Figure 1.3. NUSwan driving in a local reservoir in Singapore. The prototype utilized only GPS to navigate [10]. 14

Figure 1.4. Roboat floating in a calm body of water. The prototype included GPS, 3D LIDAR, camera, and IMU sensors [11]. 15

Figure 1.5. Remote controlled catamaran floating in Cegielnia pond in Gliwice, Poland. Select probes attached to the back analyzed water quality [13]. 15

Figure 1.6. HydroNet ASV deploying water sampling collection mechanism. The onboard winch controlled the depth of the deployed water sampler [14]. 16

Figure 1.7. Schematic diagram for soil sample collecting USV. The prototype was fit to a full-size boat for testing [15]. 17

Figure 1.8. Two ASVs deployed in a local reservoir in Durango, Colorado. One is equipped with water quality sensors and the other with a water sampling apparatus [16]. 17

Figure 1.9. Sample scenario of three USVs acting as recharge stations for AUVs. Simulations and field experiments are performed to test algorithms [17]. 18

Figure 2.1. A 2D model of boat driving against river current. The forces in the y direction are ignored, since we are focused on moving in the x-direction. Assuming the boat weighs 15 lbs. and accelerates from rest to 5 mph in 5 seconds, the force required to accelerate is significantly less than the force required to overcome drag. Therefore, we can neglect the force required to accelerate. The resulting equation can be seen in Equation 2.2. 20

Figure 2.2. Required thrust to drive a boat at varying velocities in a 1.7 mph water current, or about the average speed of the Amazon River. Assume steady state conditions, meaning the boat is driving at constant velocity, only required to overcome drag force. Also assume the boat has the shape of a round capped cylinder. A reasonable output thrust to be achieved will be in the range of 10 – 25 lbf. 21

Figure 2.3. Breakdown model of USV structure for 3D printing. The structure is printed in 7 sections due to the print size limitations. The back fin holds the back left and right pieces together. The remaining sections were press fit together using wooden dowels. All connection points were adhered, and water sealed with rubber. The design was modeled around the thrusters while maintaining a fluid dynamic shape. 24

Figure 2.4. Final 3D printed boat. An extra coat of waterproof, rubber sealant paint was used to prevent structural decay of the PLA. The final weight of the structure alone was 9 lbs., likely due to the added waterproofing materials. The thrusters were attached in through holes with machine

screws. Note: the thermocouple at the end of the fin was placed temporarily to practice via measuring and storing data. The final dimensions were 18 x 17 x 7 inches (LxWxH). 25

Figure 2.5. Electric circuit in boat. The 16V battery supplies power to the thrusters and microcontroller. The ESCs regulate the duty cycle of the thrusters, which is set by the microcontroller. A voltage regulator is used to maintain an input voltage of 9V to power the microcontroller. The microcontroller powers all other 5V sensors used. Parallel resistors and capacitors are used to measure the DC voltage across the battery. Fuses are placed throughout the circuit to prevent any damage. 27

Figure 2.6. 4S Battery capacity vs battery voltage [22]. The voltage of the battery corresponds to the amount of available power left in the battery. These known values are plotted and given a polynomial relationship to estimate power consumption. This specific set of data is only valid for 4S LiPo batteries, which are used in this prototype. 28

Figure 2.7. Path planning logic using GPS and compass sensors. The desired angle is calculated from 0º North increasing clockwise, same as the onboard compass measurements. The goal of the USV will be to drive from point A to B, B to C, and C back to A. These locations will be predetermined and input as GPS coordinates by the user. 30

Figure 2.8. Picture of final working prototype. The final USV weighs 14.2 lbs with all including components. The thrusters and sonar sensor were placed below the USV structure, while the battery and all other electronics were placed in waterproof containers on top. A waterproof antenna for the GPS was placed outside the container for a better signal. 31

Figure 2.9. Electronics container including GPS, microcontroller, microSD card reader, and voltage divider PCB board. The power distribution (bottom left) is connected to the 14.8V battery, supplying power to the system. The microcontroller with attached Xbee module (bottom right) controls and powers all 5V sensors from the PCB board (top right). Also attached to the PCB board is the GPS module. The ESC and microSD card reader (not seen), are placed under all seen electronics. Any exposed metal/pins are insulated with electric tape. 32

Figure 2.10. Isolated compass placed in separate container on top of electronics container. All connected wires are run to the electronics container below to connect to the microcontroller. A sheet of aluminum foil was placed below for electric shielding. This setup isolates the compass from high current wires inducing electric fields, providing more accurate measurements for the Earth's magnetic field. 33

Figure 2.11. Water splash shield and rope frame. The splash shield was made from re-sealable plastic bags, providing a press fit cover and easy access to the electronics container. While the splash shield is not waterproof, it significantly reduced the amount of water entering the upper cavity. The rope frame was tightened around the boat and acted as an attachment point for the retrieval tether. 34

Figure 2.12. Final prototype floating in Lake Harner. The USV floats at the desired level, such that the thrusters are completely submerged in the water. The retrieval tether seen is a 0.5-inch polypropylene rope, used to pull the USV to shore in case of failure during testing. 35

Figure 2.13. Phase breakdown of programming logic. The calibration and orientation phases have simple steps and can be explained in order. The auto-drive phase is more complex and described

in Figure 2.14. The orientation phase and auto-drive phase repeat for each driving path described in Figure 2.7. After the last path is complete, the program terminates and waits for the next user-input command..... 36

Figure 2.14. Detailed overview of Auto-drive logic. The top left (dark blue) updates measurements every 1 Hz. Each path diverting from the top left executes on a timer. The turning logic (yellow) updates at 3.3Hz, and the speed logic (blue) updates at 1Hz. The angle Φ is the difference between the desired angle and measured compass angle, used for turning logic. The speed logic has limits to input PWM values to prevent thruster damage..... 37

Figure 2.15. Basic turning logic example. The first case (left) shows the resulting Φ is positive, indicating a right turn. The second case (right) shows the resulting Φ is negative, indicating a left turn. 38

Figure 2.16. Fully defined conditions for turning logic. The first special case (top right) shows the desired and measured angles being on opposite sides of the 0° – 360° limits, resulting in a positive Φ, but with a magnitude greater than 180°. This indicates a left turn, opposite of simple logic in Figure 2.15. The second special case (case 2) creates a similar issue, however both turning directions can be determined by knowing the sign and magnitude of Φ. 39

Figure 2.17. BlueRobotics T200 thruster current draw vs input PWM [23]. An input PWM of 1500 µs is off for the thrusters. Below 1500 µs to 1100µs is reverse. Above 1500 µs to 1900 µs is forward. Focusing on the 1500-1900 µs range corresponds to a 0-100% duty cycle..... 41

Figure 2.18. Lake Harner, Indiana [24]. The marked point indicates the start location of the tested route, with North pointing up. Only a small portion of the lake was used for testing. The total driving route was about 200 ft extending from the shore, whereas the scale shows 500ft. The full route can be seen in Figure 3.2. 42

Figure 2.19. Wabash River at Davis Ferry Park, Indiana [25]. The marked point indicates the start location, with North pointing up. The river current is estimated to be about 1.7 mph flowing West. The full route tested can be seen in Figure 3.4. 43

Figure 2.20. Example of data saved on microSD card. All data is manually transferred from the microSD card to a computer, where it is analyzed in MATLAB. Data is sampled at 3.3Hz. This is the maximum frequency allowed for collecting data on the used Arduino Mega 2560. Higher sampling rates result in backlog errors. 44

Figure 3.1. Depth of water in Lake Harner measured by sonar sensor (Top). Depth of Lake Harner with 5 ft contours reported in 2017 by Indiana Department of Natural Resources (Bottom) [26]. The depth measured by the sonar sensor closely matches the contour map, indicating the measurements are accurate. A 2D map of the USV driving route can be seen in Figure 3.2..... 45

Figure 3.2. Path driven by SVU on Lake Harner. The USV reaches within 12ft of the desired locations. As the USV deviates from the desired straight path, it corrects itself using the turning logic described in Figure 2.14. The black arrows indicate the direction traveled by the USV, and the letters indicate the desired location. The scale is modified to 0 ft as the starting location and increasing North and East. 46

Figure 3.3. Depth of water in Wabash River. Although there is no official recorded depth of this location, the accuracy tested in Lake Harner indicates these results are also accurate. A 2D map of the USV driving route can be seen in Figure 3.4. 47

Figure 3.4. Path Driven by USV on Wabash River. The USV is manually skipped to C after struggling to make it to point B. The dotted circle attached to B indicates where the path was manually terminated. The river current arrows indicate a stronger current at the center of the river. The desired path structure was an "L" shape, however, drag from the retrieval tether made it too difficult to test the auto-drive phase up-stream. 48

Figure 3.5. Measured USV speed and input steady state PWM. Each break/color indicates a new driving phase, after reaching each desired location. The path moving against the river current (red) was the path manually terminated. The speed and thruster duty cycle are compared to show the successful speed logic. 50

Figure 3.6. Measured Voltage across battery and input steady state PWM. Each break/color indicates a new driving phase, after reaching each desired location. The path moving against the river current (red) was the path manually terminated. The battery voltage and thruster duty cycle are compared to show the relationship between power consumption and required thrust. 52

ABSTRACT

Water pollution in drinking water is a major concern in rural areas that depend on local surface and ground water supplies. The Amazon river, for example, has 800 thousand rural inhabitants, many of whom do not have access to treated water. Reaching the Amazon River to collect these water samples is already a complicated task. With constantly changing floodplains, and therefore water quality, the ability to collect water samples remotely and autonomously can help rural areas monitor their drinking water. There have been several studies investigating different unmanned surface vehicle (USV) prototypes and data collection methods. However, none have specifically made a compact USV to maneuver in rivers, while aiming to conserve energy to drive longer distances. This paper describes an in-depth design, fabrication, and automation process for a USV to drive in the Wabash River. The USV monitors its own location, speed, and battery voltage for power consumption analysis. As proof of concept, the USV measures water depth during field studies performed in Lake Harner, Indiana and the Wabash River. These field studies yield affirming results for the controls logic and power conservation of the designed USV.

1. INTRODUCTION

Water pollution is an increasingly important issue. Just in the past 30 years, the NOAA has recovered $10.4 billon USD from accountable parties to restore 306 different settlements damaged by pollution [1]. Even more concerning, most drinking water comes from groundwater and surface water sources [2], which are susceptible to chemical contaminants and other pollutants like agricultural runoff and overflowing wastewater [3]. A 2011 study was done on the Amazon River to analyze the floodplains and resulting pollutants due to water level variation and overflow. The Amazon River floodplain is home to over 1.5 million people, 54% of which live in rural states at the time. Within the Amazon floodplain, 32% of the total water volume from the floodplains is untreated. On top of that, 56% of the rural households do not have indoor plumbing, and 93% do not have a sewage system [4]. Therefore, monitoring water quality is essential for these rural populations. The study found that there are different times throughout the hydrological year that the water becomes dangerous to consume. Particularly during the receding state, there is extreme sedimentation and the floodplains are generally only safe for navigation and landscaping. Water sampling is required to measure the presence of foreign bodies but accessing these areas can be challenging. Creating an autonomous vehicle to collect these water samples on a needed basis can greatly increase frequency of water samples taken, increase the number of samples taken, and provide overall benefit for the health of the nearby inhabitants.

The Wabash River in Lafayette, Indiana has a similar problem due to overflow. When Lafayette's combined sewer and storm water overflows due to heavy rains or melting snow, it is redirected into the Wabash River. While the city is working to install tanks to collect this overflow [5], it is still a polluted river. Although this is not as critical of a safety issue as the Amazon floodplains, creating a prototype of an unmanned surface vehicle (USV) to collect data from the Wabash River will create a proof of concept for deploying similar vehicles in these more critical areas.

Automation in robotics has advanced over the last 30 years [6]. Recent advancements in self-driving cars with the goal of complete autonomy may ultimately become a reality. However, two important issues arise with sensor implementation and power management. Increasingly large quantities of sensor data need to be analyzed in real-time, demanding more processing power [6][7]. USVs are a type of self-driving vehicle and experience the same challenges. Part of managing this

data is creating a USV to conserve and monitor power consumption, allowing more energy to be utilized for sample acquisition and analysis.

It is important to investigate all the existing USVs that were created with a goal to monitor water quality. First focusing on physical design and implemented navigation sensors, one existing prototype is a gliding robotic fish created in 2015. Figure 1.1 shows this robot was designed to submerge in a spiraling motion to collect water samples at a certain depth, then surface and drive to its next location. This process was stated to take about 5 minutes for each location. An attached algae sensor was utilized to analyze the number of algae in each water column collected. This robot used only GPS to navigate, looking at the difference between its current and past GPS locations. It was stated a compass could not be used due to the robot constantly oscillating its orientation due to the designed swimming motion. A Digi Xbee mesh network was used for communication between the user and robot, and all tests were performed in Wintergreen Lake, Michigan [8].

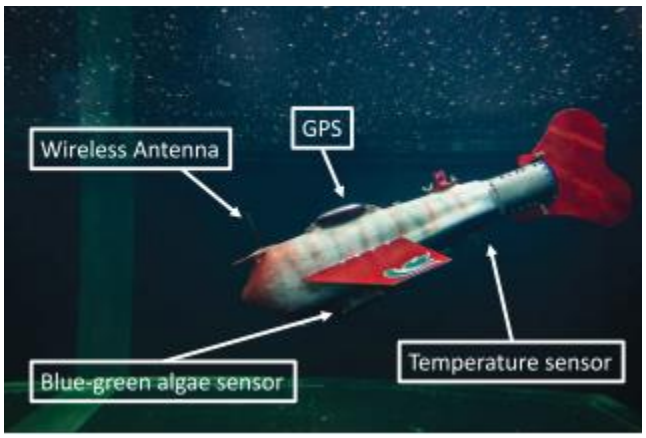

Figure 1.1. Gliding Robotic Fish diving underwater to collect water column. The prototype utilized only GPS to navigate [8].

Another USV from 2015 was an automated catamaran, called the HydroNet ASV, used for long distance navigation and water sampling, seen in Figure 1.2. It utilized GPS, compass, laser scanning, sonar, and an altimeter for path planning navigation. Weighing 183 lbs. with an attached 60,000 mAh battery, results showed it could drive up to 12.4 miles at 3 mph. All tests were performed in Livorno harbor, Italy [9].

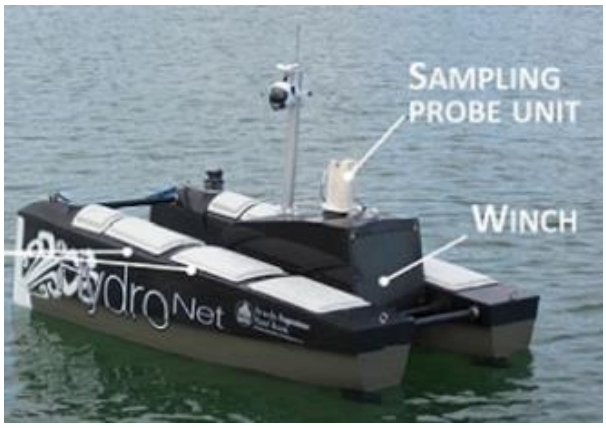

Figure 1.2. HydroNet ASV floating in Livorno harbor, Italy. The catamaran prototype included a GPS, compass, laser scanner, sonar, and altimeter sensors [9].

A 2017 USV developed in Singapore to collect water samples focused on a more minimalist design to conserve battery power, seen in Figure 1.3. The NUSwan, designed to look like a swan, used only GPS to navigate. Multiple NUSwans could be deployed and communicate with each other via 2-3G networks and were capable of docking at recharge stations. Data for water sampling was collected, however there was no information on control methods [10].

Figure 1.3. NUSwan driving in a local reservoir in Singapore. The prototype utilized only GPS to navigate [10].

A more recent example is the 2019 Roboat, an automated surface vehicle for urban waterways, shown in Figure 1.4. This USV utilized GPS, 3D LIDAR, Camera, and IMU sensors for control. It had 4 thrusters, one on each side to drive the 3Lx1.5Wx0.5H ft chassis. Weighing about 33 lbs., it used a 11.1V LiPo battery, able to operate for about 3 hours on a single charge. The Roboat is stated to drive at its desired speed of about 1 mph in the Charles River in Massachusetts. It is

important to note this USV emphasized obstacle avoidance in heavy traffic, narrow waterways [11]. A research group at MIT is currently taking this design one step further with multiple USVs running in the canal and performing socially compliant navigation, similar to how humans share roadways [12].

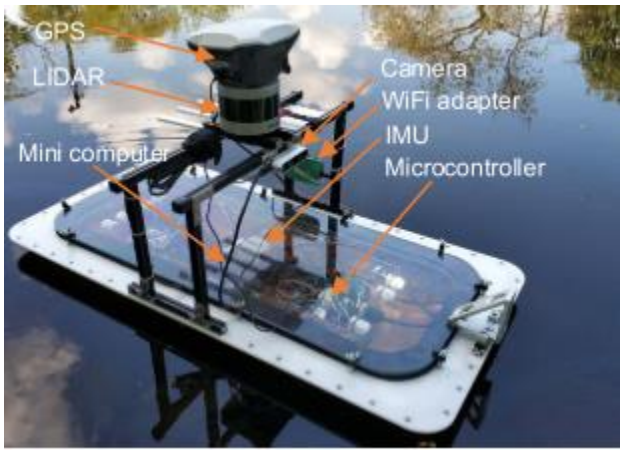

Figure 1.4. Roboat floating in a calm body of water. The prototype included GPS, 3D LIDAR, camera, and IMU sensors [11].

Additionally, there are other studies that focus more on the automation process of collecting and analyzing water samples. A 2017 study utilized a remote-control catamaran with attached sensors to analyze the water remotely, as well as collect samples for further testing. Figure 1.5 shows the onboard sensors that measure pH, potassium, sodium, chlorine, and temperature. The field experiments were performed at Cegielnia pond in Gliwice, Poland [13].

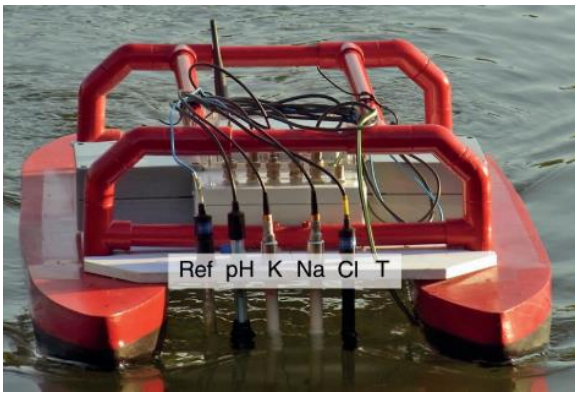

Figure 1.5. Remote controlled catamaran floating in Cegielnia pond in Gliwice, Poland. Select probes attached to the back analyzed water quality [13].

A similar study in 2017 used the previously described automated catamaran, HydroNet ASV. The study focused on the process of collecting water samples using a winch mechanism but did not detail the automated driving process. The field experiments were performed in front of Livorno harbor, Italy during calm water conditions, as seen in Figure 1.6 [14].

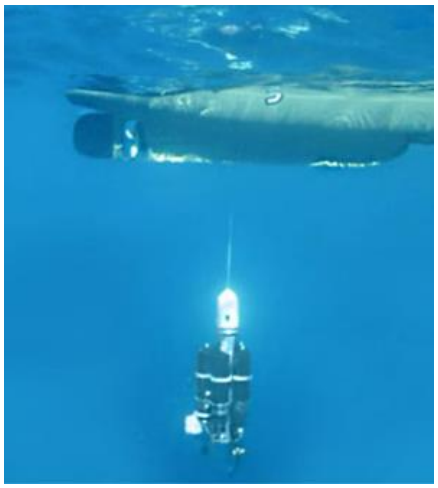

Figure 1.6. HydroNet ASV deploying water sampling collection mechanism. The onboard winch controlled the depth of the deployed water sampler [14].

Figure 1.7 illustrates a more unique study from 2018 that used a full-size boat to collect underwater soil samples. The whole process of collecting the soil sample and driving the boat was automated. It employed a gyro stabilizer to prevent the boat from rolling during the retrieval process. However, the process of automating the USV itself is not explained [15].

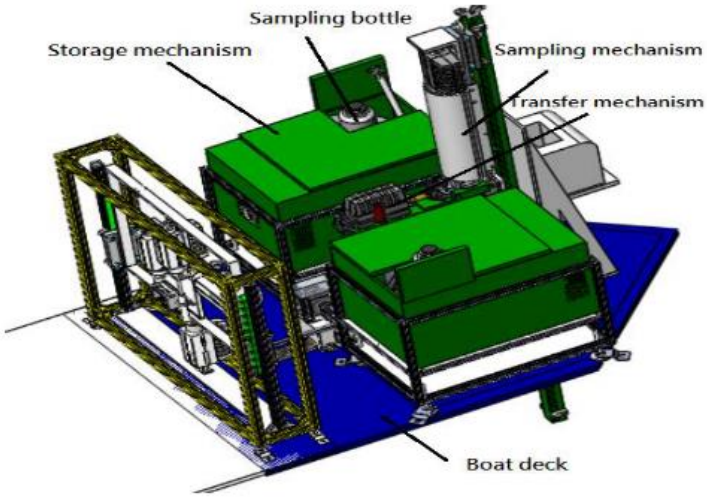

Figure 1.7. Schematic diagram for soil sample collecting USV. The prototype was fit to a full-size boat for testing [15].

Pivoting towards controls and automation logic, most studies emphasize the path planning efficiency algorithms. One such robotic system developed in 2018 involved two autonomous water surface vehicles, an “explorer” and a “sampler”, seen in Figure 1.8. This system was used in areas where there was no prior knowledge about the water. The explorer robot took water quality data, implying that it characterized the physical parameters of its surroundings. The second robot then took the most efficient route to collect data samples in desired regions. The paper did not provide details on how these robots were driven or made, other than it was evaluated in a simulation and in the field [16].

Figure 1.8. Two ASVs deployed in a local reservoir in Durango, Colorado. One is equipped with water quality sensors and the other with a water sampling apparatus [16].

Another type of system developed in 2020 involved using USVs as mobile charging stations for autonomous underwater vehicles, or AUVs, illustrated in Figure 1.9. This study focused on creating a network of robots in water areas, able to perform long-term missions. While numerical studies were done to validate the concept, small scale field experiments were also performed in Lake Superior with an AUV and manned surface vehicle (SV) to test feasibility. The AUV successfully docked onto the SV, but automating the entire system is an ongoing project [17].

Figure 1.9. Sample scenario of three USVs acting as recharge stations for AUVs. Simulations and field experiments are performed to test algorithms [17].

Additional studies have examined high level algorithms and run simulations [18][19][20], however there is less focus on how those methods perform in actual marine environments such as oceans, rivers, and ponds.

With all the different USVs currently available and studies done, there are none that focus on monitoring power consumption, or how to efficiently drive in rivers to conserve power. We are proposing a USV design that can use the moving waters to its advantage. By minimizing power to the USV thrusters and riding the river current, we can ideally drive longer distances and collect more water data. In addition, we are proposing a method of monitoring the power consumed from the battery. By measuring the battery voltage to estimate power draw, we can confirm the USV's logic is driving conservatively downstream; and we can potentially determine the most efficient paths overall to make real time adjustments. This power-conservative driving algorithm, along with the ability to monitor relative power consumption, can be incorporated in other USVs to potentially cover larger water sampling areas in moving waters and overall help better monitor water quality.

This thesis will discuss the in-depth design, fabrication, and automation process for building a USV capable of driving in the Wabash River. The task of this drone is to autonomously drive to designated GPS coordinates and conservatively maneuver across moving waters while measuring battery voltage. The measured voltage will be used to compare the power consumed along each path and confirm power-conservative driving algorithms in moving waters. The depth of the water will also be measured as a proof of concept for collecting other water data, such as water samples.

2. METHODS

2.1 Preliminary Design

To meet thrust requirements, a simple boat structure is modeled to be driving against a 1.7 mph current, which is about the average current of the Amazon River. It is assumed the boat is at steady state, moving at a constant velocity. This is because the force due to drag is much larger than the force required to accelerate a small boat from rest. Since we are focusing on movement in the x direction, only the forces in the x direction are considered.

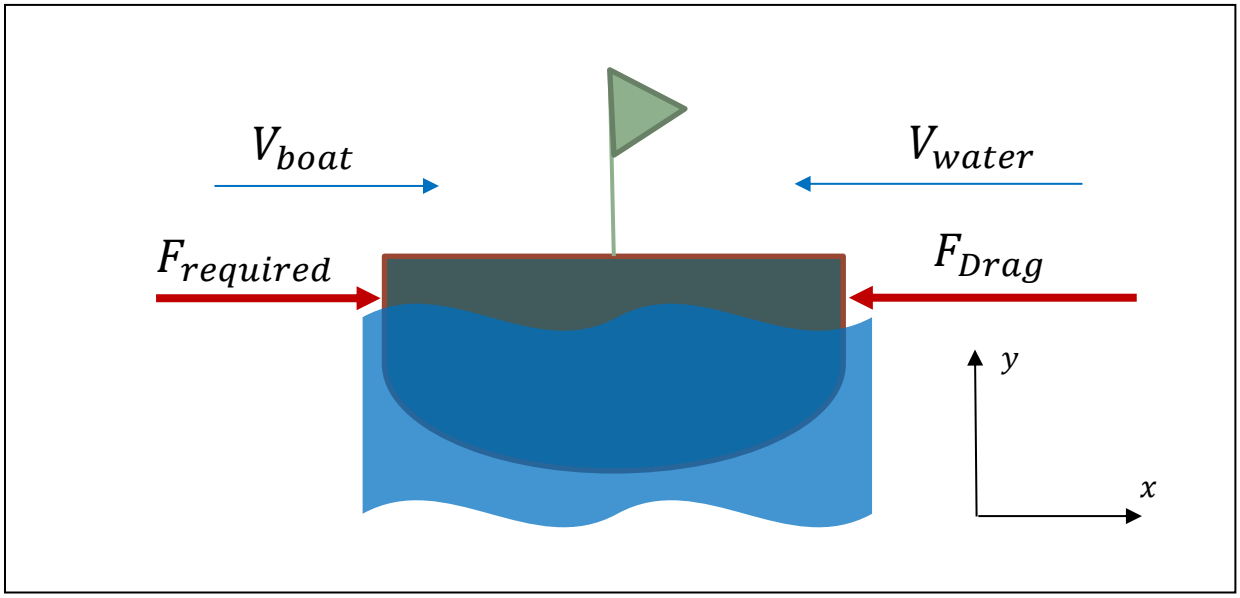

Figure 2.1. A 2D model of boat driving against river current. The forces in the y direction are ignored, since we are focused on moving in the x-direction. Assuming the boat weighs 15 lbs. and accelerates from rest to 5 mph in 5 seconds, the force required to accelerate is significantly less than the force required to overcome drag. Therefore, we can neglect the force required to accelerate. The resulting equation can be seen in Equation 2.2.

2.1.1 Thruster Specifications

Considering this is a dynamic problem, we sum the forces in the x direction.

$$\sum F_x = F_{required} - F_{Drag} = ma_{boat} \qquad \text{Equation 2.1}$$

Substituting in the force due to drag and assuming the boat is moving at a constant velocity, we can rearrange Equation 2.1.

$$F_{required} = F_{Drag} = \frac{1}{2}\rho C_d A(V_{water} + V_{boat})^2 \qquad \text{Equation 2.2}$$

Assuming the shape of the boat is a round capped cylinder to estimate the drag coefficient, the two unknowns are the required force and the steady state velocity of the boat. Choosing the velocity of the boat as the input, we can calculate the required force to move the boat at that desired velocity.

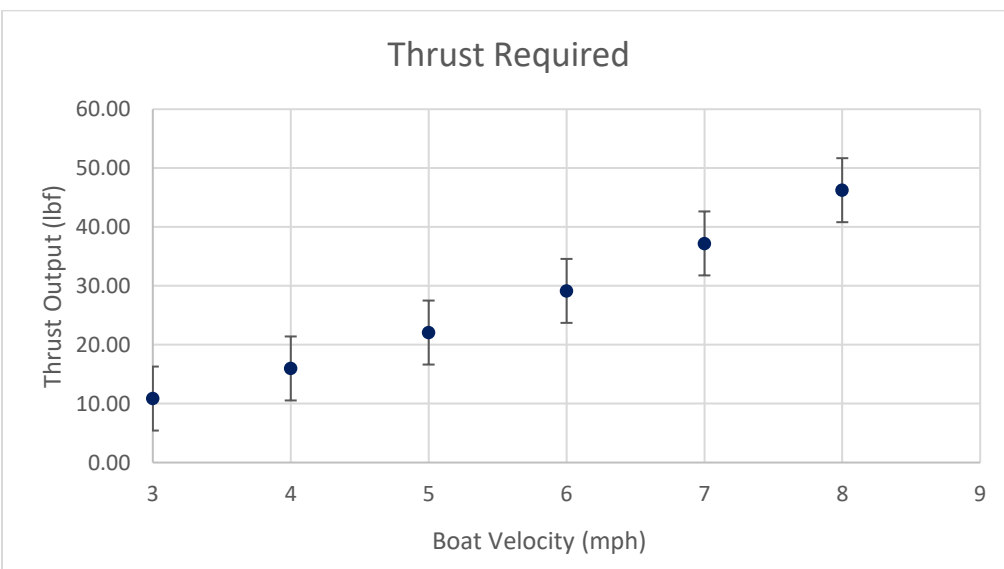

Figure 2.2. Required thrust to drive a boat at varying velocities in a 1.7 mph water current, or about the average speed of the Amazon River. Assume steady state conditions, meaning the boat is driving at constant velocity, only required to overcome drag force. Also assume the boat has the shape of a round capped cylinder. A reasonable output thrust to be achieved will be in the range of 10 – 25 lbf.

Figure 2.2 can be used to choose the appropriate thruster for the boat. The thruster must be water-tight, provide enough thrust to drive in rivers, have reasonably low power draw, and light weight. It was found that the BlueRobotics T200 thruster was the best fit.

Table 2.1. BlueRobotics T200 current draw and thrust output at 16V nominal voltage

Duty Cycle (%)	*Current Draw (Amps)*	*Thrust Output (lbf)*
50	4.6	4.0
75	11.9	7.5
100	23.8	11.5

Fitting two T200 thrusters to the boat provides enough thrust to drive at a stable 3 - 4 mph, with burst speeds of 5 mph. Maintaining a low current draw throughout the driving cycle prolongs battery life, as well as prevents battery and thruster overheating.

2.1.2 Battery Specifications

With the desired thruster, the stable current draw from a 75% duty cycle is approximately 12A per thruster. From Table 2.1 and Figure 2.2, using two thrusters provides a speed of 4 mph. This information can be used to estimate the required battery size to drive against a current for 1 mile.

$$Battery\ Size = \frac{1(mile)}{4(\frac{miles}{hr})} * 12\left(\frac{A}{thruster}\right) * 2(thrusters) = 6000\ mA\ hr \quad \text{Equation 2.3}$$

Thus, a battery size of 6000 mAh and discharge current of 24A is required for the thrusters. A 14.8V 7200mA-hr 4S LiPo battery was chosen, as it provides more than the required capacity and a maximum discharge current of 576A. Lithium Polymer (LiPo) batteries provide compact sizes with an extraordinary amount of discharge currents. They are widely used in RC vehicles due to their power capabilities and low cost – a perfect fit for prototyping. However, LiPo batteries are highly flammable and capable of deforming. Current fuses were placed throughout the system to prevent this from happening. The battery was also isolated to its own water-tight box to contain any damage if it were to occur.

2.2 USV Structure Fabrication

The design of the USV structure was a result of keeping the following parameters in mind. It must be light weight so that it floats, but not too light that it can be easily carried away by water currents. The size must be big enough to hold all electronics and water samples, but small enough to be easily maneuverable. The shape should be fluid dynamic but wide enough such that it remains stable and does not capsize. And lastly, material must be waterproof such that it does not lose structural integrity over time when submerged in water.

It was decided that the USV be built using a 3D printer and PLA (polylactic acid) filament, with a 20% infill. The ease of access to a 3D printer for iterative prototyping posed as a major benefit in the design process. In addition, the porous infill of the 3D printing provides a strong yet buoyant structure. The USV was designed using Autodesk Inventor, and was mainly designed around the placement and safety of the T200 thrusters, as seen in Figure 2.3. Machine screw holes were placed at the two back ends of the USV to attach the thrusters. The side fins act as bump guards for the T200 thrusters, and the empty space is designed in the top center of the USV to reduce weight and hold the battery and other electronics. The proposed design was then evaluated to make sure it would float. Modeling the USV completely submerged in water, we can use Archimedes' principle to calculate buoyancy force, and thus the weight limit.

$$F_{bouyant} = \rho g V_{fluid\ displaced} \qquad \text{Equation 2.4}$$

Based on the proposed design, the resulting buoyancy force was 21.6 lbs. This means that the total weight of the boat could not exceed 21.6 lbs. The estimated weight of the proposed volume for 20% infill PLA plastic is 5.4 lbs. This is well below the weight limit, which is desirable since it does not account for the added weight from the thrusters and other electronics. Several different iterations of the structural design went through the same analysis before arriving at the final design, with dimensions 18 x 17 x 7 inches (LxWxH).

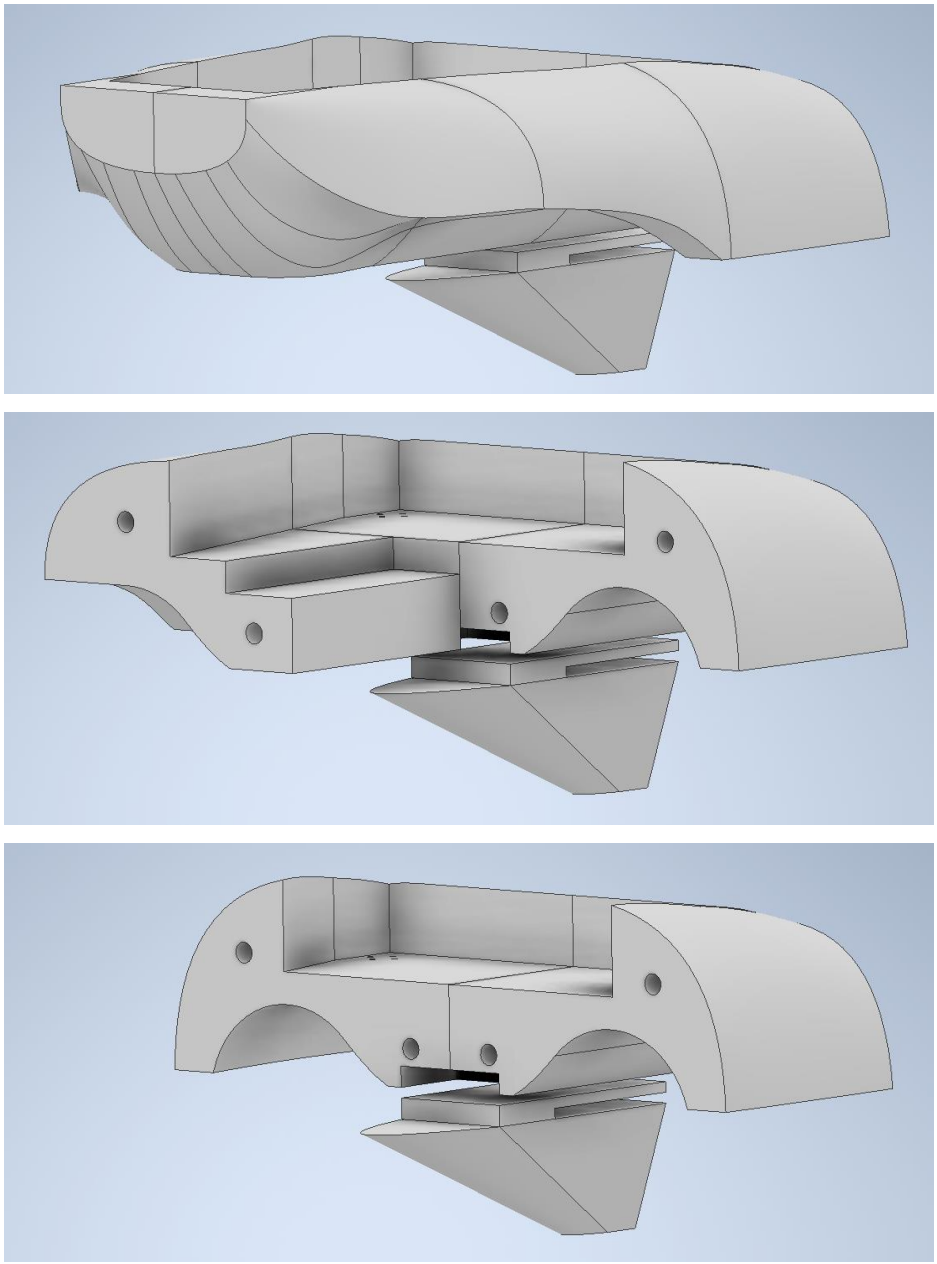

Figure 2.3. Breakdown model of USV structure for 3D printing. The structure is printed in 7 sections due to the print size limitations. The back fin holds the back left and right pieces together. The remaining sections were press fit together using wooden dowels. All connection points were adhered, and water sealed with rubber. The design was modeled around the thrusters while maintaining a fluid dynamic shape.

Figure 2.3 shows how the USV needed to be printed into seven different parts due to the size limitations of the 3D printer. The bottom fin slides into the back two pieces, and the rest of the parts are fit together using wooden dowels. To provide a watertight seal and more solid fit, the

pieces were adhered together using rubber sealant tape. In addition, PLA is a natural polymer based from corn starch, tapioca roots, or sugarcane [21]. This means that the plastic can easily degrade and lose its structural properties over time when exposed to water. Since the USV structure will constantly be exposed to water, an exterior coat of rubber sealant spray paint was applied. The added waterproofing paint and tape result in a final structural weight of 9 lbs. While this is heavier than the estimated model, it is still well below the 21.6 lbs. maximum.

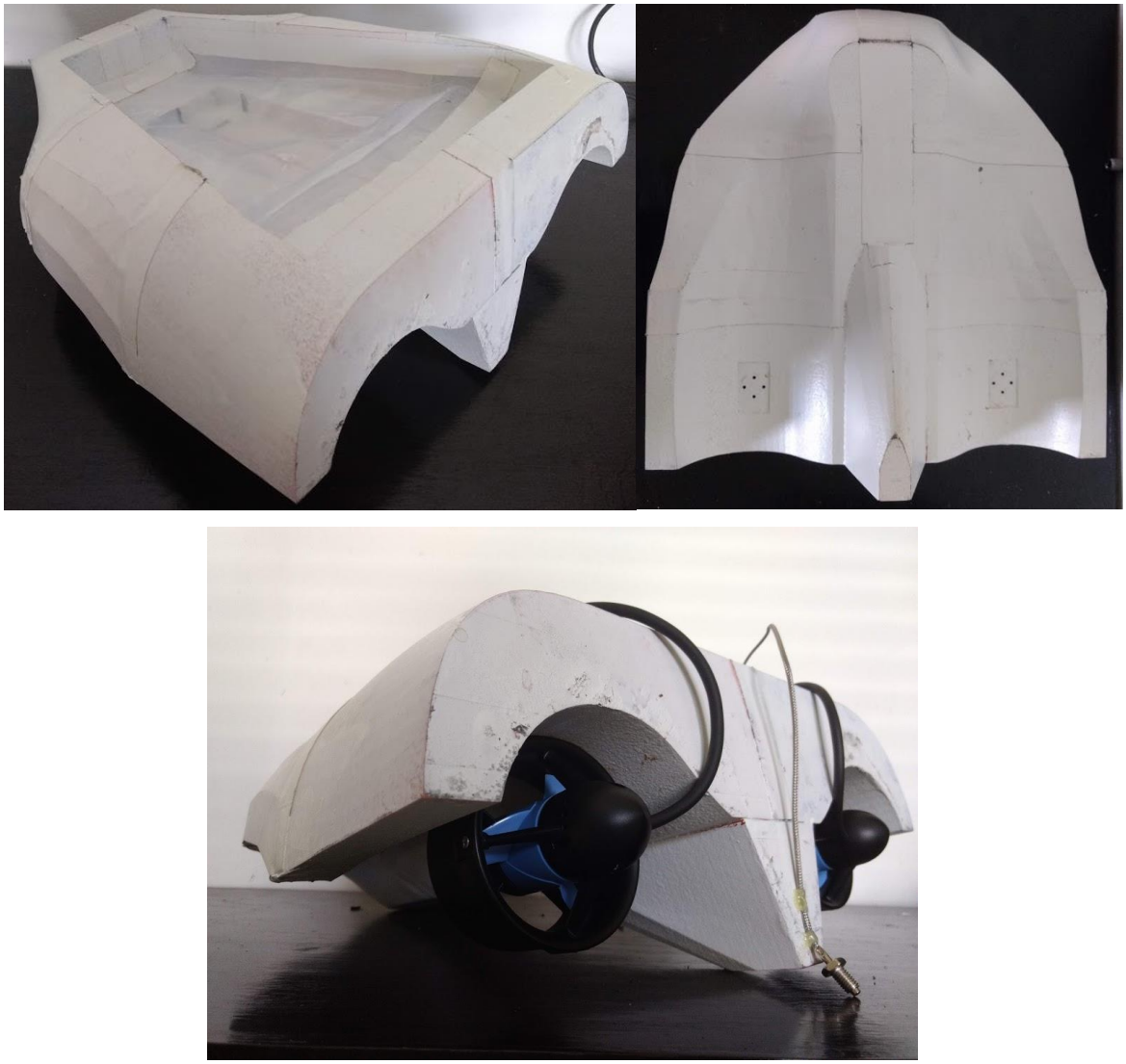

Figure 2.4. Final 3D printed boat. An extra coat of waterproof, rubber sealant paint was used to prevent structural decay of the PLA. The final weight of the structure alone was 9 lbs., likely due to the added waterproofing materials. The thrusters were attached in through holes with machine screws. Note: the thermocouple at the end of the fin was placed temporarily to practice via measuring and storing data. The final dimensions were 18 x 17 x 7 inches (LxWxH).

2.3 Electronics and Circuit Design

The 14.8V LiPo battery supply required to power the T200 thrusters must also be used to power the microcontroller. The Arduino Mega 2560 microcontroller was chosen for its ease of integration with different sensors and its large number of serial ports. The Arduino requires a 9V power supply and the thrusters require the battery's 14.8V supply. The thrusters and microcontroller must be supplied these constant voltages, so they must be placed in parallel. However, the thrusters and microcontroller will only draw as much current as they need, which is constantly changing. The thrusters draw between 1A to 23.8A each, depending on the desired thrust output; and the microcontroller draws between 40mA and 800mA, depending on the number of sensors connected. These consistent changes in current draw created a dynamic circuit such that a simple resistor cannot be used to provide a consistent voltage drop to 9V for the Arduino. Therefore, a voltage regulator was used to provide a constant voltage output of 9V, independent of current draw and input voltage. The thrusters do not need any voltage regulation since they require the battery's full voltage output. Also, while the battery says it provides a 14.8V supply, the real voltage supply at full charge is 16V, and decreases to 14.8V near depletion. This is okay, since it has little effect on the thrusters, and the voltage regulator on the microcontroller maintains the required 9V input.

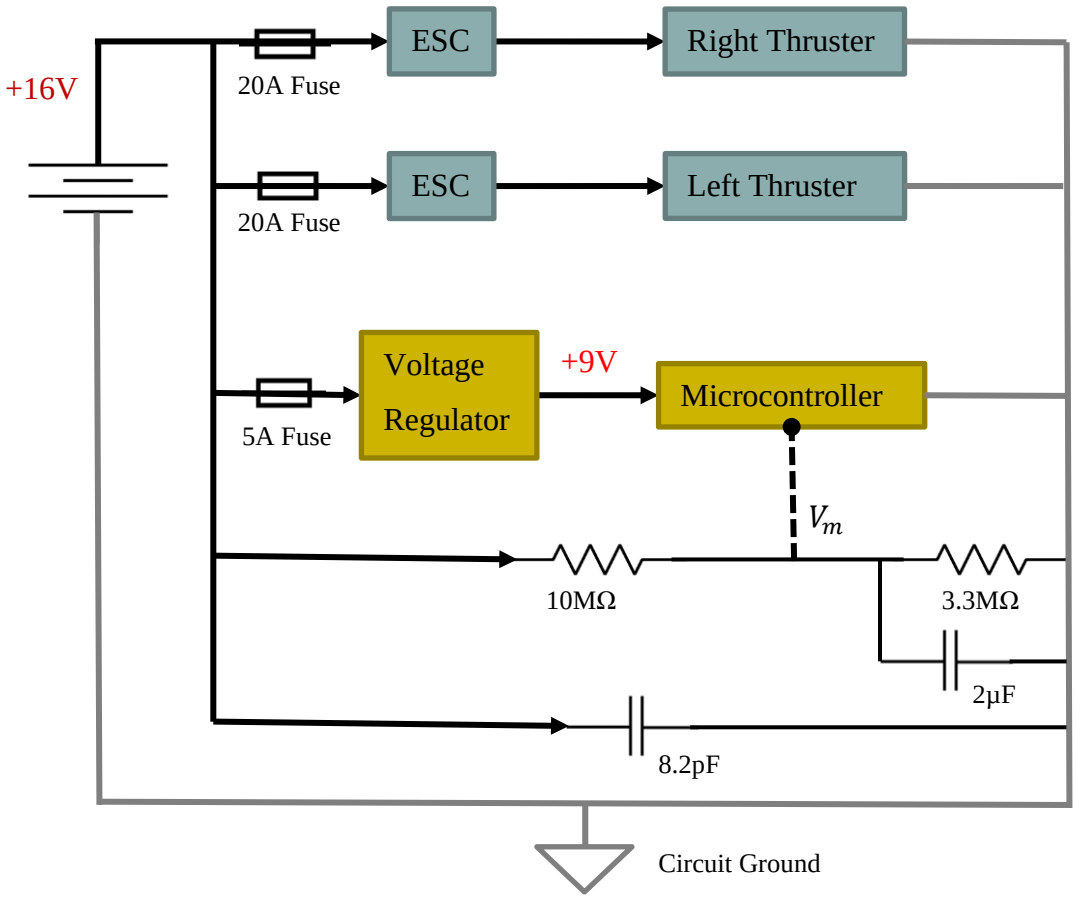

Figure 2.5. Electric circuit in boat. The 16V battery supplies power to the thrusters and microcontroller. The ESCs regulate the duty cycle of the thrusters, which is set by the microcontroller. A voltage regulator is used to maintain an input voltage of 9V to power the microcontroller. The microcontroller powers all other 5V sensors used. Parallel resistors and capacitors are used to measure the DC voltage across the battery. Fuses are placed throughout the circuit to prevent any damage.

Fuses were placed before each branch of the circuit to prevent any current overloads. The thrusters were expected to never go over 90% of their duty cycle, so the max current draw is 19A. The electronic speed controllers (ESC's) were used to read an input pulse width modulation (PWM) from the microcontroller, which corresponds to a desired thrust output. The voltage divider with parallel capacitors at the bottom of the circuit were used to estimate the voltage remaining in the battery. This information was used to quantify power consumption.

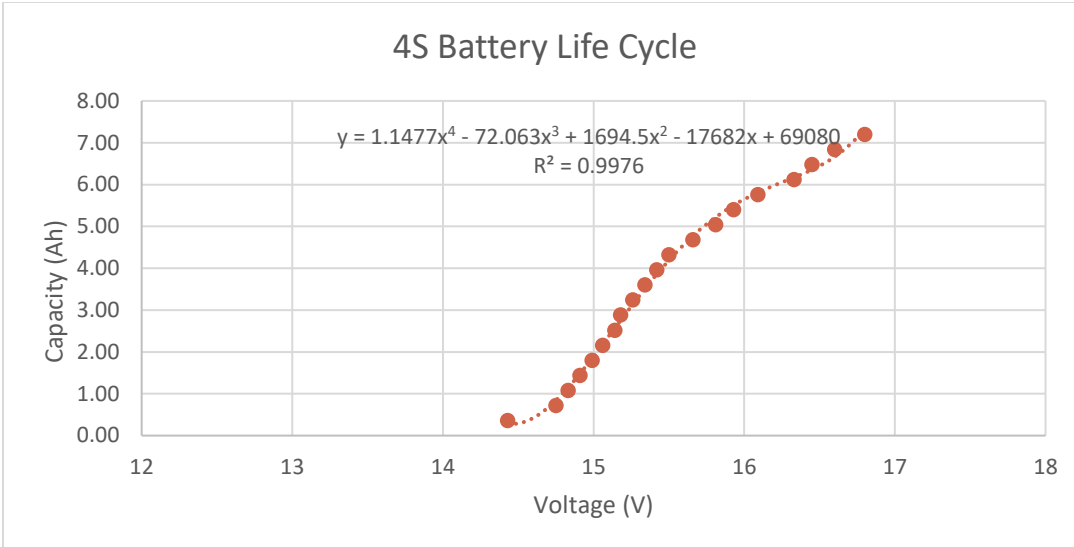

Figure 2.6. 4S Battery capacity vs battery voltage [22]. The voltage of the battery corresponds to the amount of available power left in the battery. These known values are plotted and given a polynomial relationship to estimate power consumption. This specific set of data is only valid for 4S LiPo batteries, which are used in this prototype.

Figure 2.6 relates a 4S, or 4 cell, battery's capacity to voltage. As the battery capacity decreases, the measured voltage also decreases. Using this relationship, the capacity remaining in the battery can be estimated from the measured voltage across the battery. It is important to note Figure 2.6 is valid only for an unloaded battery, meaning there is no current being drawn. As the load across the battery increases, the measured voltage will decrease. For example, if the battery is 100% charged and both thrusters are off, the voltage will read 16.8V; but when the thrusters are turned on to 50% duty cycle, the voltage will read about 16.0V. This phenomenon must be considered when moving forward in measuring power consumption.

A voltage divider must be used since the microcontroller can only measure between 0V – 5V. Very large resistors were chosen to allow very little current to pass through and therefore prevent battery leakage. Capacitors were placed in parallel to act as passive low pass filters, aiming to only pass DC current. Kirchhoff's voltage law can be used to calculate the voltage across the battery from the voltage measured by the microcontroller. This equation assumes the power supply has been on for a long period of time, and the capacitors are fully charged, acting as an open circuit.

$$V_{battery} = \frac{10M\Omega + 3.3M\Omega}{3.3M\Omega} * V_m = 4.03 * V_m \qquad \text{Equation 2.5}$$

2.4 Sensors and Communication Systems

Several sensors were powered and communicated through the microcontroller. To communicate with the USV remotely over long distances, a Digi XBee Zigbee mesh network was used. The chosen XBee S2 modules work up to 300 feet, which is enough for the intended testing. The setup included using two XBee modules, one being the coordinator connected to a laptop, and the other being the receiver connected to the microcontroller on the boat. Using Digi's free XBee configuration software, XCTU, both modules can be set up and then read over a serial monitor. Both XBee's can read and write information to each other in real time, allowing the user to read outputs printed from the boat's microcontroller, as well as input start/stop commands from the laptop.

A BlueRobotics 1D Ping Sonar Altimeter and Echosounder was used to measure the depth of the water; and a micro SD Card Adapter Module was used to store all data measured by the microcontroller.

It was decided to automate the USV using path planning feedback control, seen in Figure 2.7. To achieve this type of feedback, the USV must acquire position and orientation information. A GPS Module and Adafruit BNO055 MEMS accelerometer were used to locate and orient the boat, respectively. The GPS module has a position accuracy of 6 ft and velocity accuracy of 0.33 ft/s. The position accuracy is too low to depend on GPS alone, so a magnetometer was used to orient the USV in the correct direction. The BNO055 MEMS accelerometer has a magnetometer accurate to less than 1-degree angle.

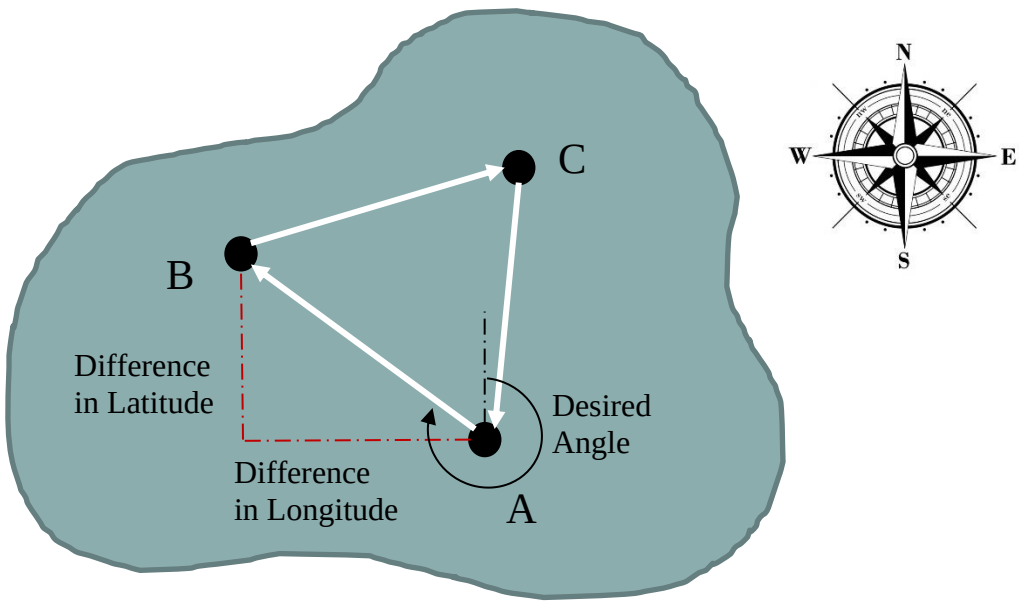

Figure 2.7. Path planning logic using GPS and compass sensors. The desired angle is calculated from 0º North increasing clockwise, same as the onboard compass measurements. The goal of the USV will be to drive from point A to B, B to C, and C back to A. These locations will be predetermined and input as GPS coordinates by the user.

Starting at point A, we can program the USV before launch to reach the GPS locations B, C, and back to A in that order, as seen in Figure 2.7. The USV measures its current GPS location, and calculates the directional degree it should be facing to reach the desired destination. For example, driving from A (current location) to B (desired location):

$$desired\ angle = atan2\left(\frac{Longitude_B - Longitude_A}{Latitude_B - Latitude_A}\right) \qquad \text{Equation 2.6}$$

The function atan2 measures between -180 to +180 degrees, so the program checks if the desired angle is negative and adds 360 degrees. This way the range for the desired angle is 0 to +360 degrees. The desired angle is measured clockwise from 0 degrees North, just as the magnetometer measures the bow (front) of the USV. We can use the difference between the desired angle and the measured angle of the USV's bow to incorporate turning logic.

2.5 Summary of Final Prototype

Figure 2.8. Picture of final working prototype. The final USV weighs 14.2 lbs with all including components. The thrusters and sonar sensor were placed below the USV structure, while the battery and all other electronics were placed in waterproof containers on top. A waterproof antenna for the GPS was placed outside the container for a better signal.

Two BlueRobotics T200 thrusters were fitted to the bottom of the USV using machine screws. The 1D sonar sensor was adhered to the bottom using water-proof epoxy. All other electronics, including the battery, microcontroller, ESC's, and other sensors, were placed in the cavity on top of the USV in waterproof containers, as seen in Figure 2.8. Any wires entering the containers were done so with water-tight cable penetrators and waterproof epoxy.

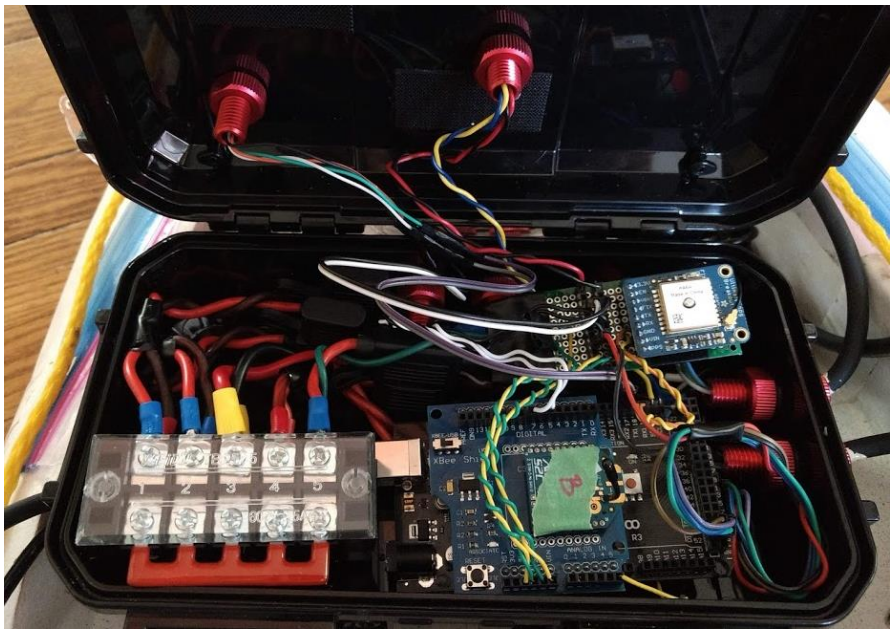

Figure 2.9. Electronics container including GPS, microcontroller, microSD card reader, and voltage divider PCB board. The power distribution (bottom left) is connected to the 14.8V battery, supplying power to the system. The microcontroller with attached Xbee module (bottom right) controls and powers all 5V sensors from the PCB board (top right). Also attached to the PCB board is the GPS module. The ESC and microSD card reader (not seen), are placed under all seen electronics. Any exposed metal/pins are insulated with electric tape.

A 14.8V 7200mAh 4S LiPo battery was utilized to provide the necessary current draw and length of driving time. To prevent any potential damage to other parts, the battery was placed in a separate waterproof container. The battery connects to the electronics container and supplies power to several different parts of the boat, with included fuses to prevent any electrical damage. It powers the electronic speed controllers (ESC's), which power the T200 thrusters. The ESC's establish the duty cycle based on the PWM input from the microcontroller. The battery also powers the Arduino Mega 2560 microcontroller, which in turn powers and reads data from all the 5V sensors used in the system. These include the GPS sensor, compass, 1D sonar sensor, and microSD card module. The Xbee radio module can be seen Figure 2.9, indicated with the green tape as "B".

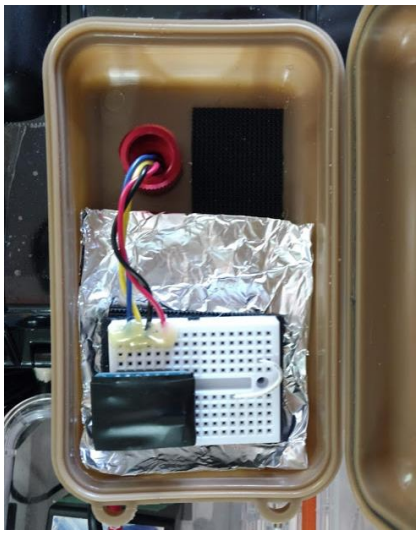

Figure 2.10. Isolated compass placed in separate container on top of electronics container. All connected wires are run to the electronics container below to connect to the microcontroller. A sheet of aluminum foil was placed below for electric shielding. This setup isolates the compass from high current wires inducing electric fields, providing more accurate measurements for the Earth's magnetic field.

The wires in the electronics container and wires coming from the battery container can reach high currents, some potentially up to 19A under extreme conditions. These currents can create a false magnetic field that can alter the readings from the compass. Addressing this issue, the compass was moved away from the wires into its own container and shielded with aluminum foil, seen in Figure 2.10. There was no formal research conducted to see if the aluminum foil helped, but it was an intuitive added measure to provide electrical shielding. These adjustments provided more reliable readings from the compass.

Figure 2.11. Water splash shield and rope frame. The splash shield was made from re-sealable plastic bags, providing a press fit cover and easy access to the electronics container. While the splash shield is not waterproof, it significantly reduced the amount of water entering the upper cavity. The rope frame was tightened around the boat and acted as an attachment point for the retrieval tether.

While the boat still floats when the cavity is filled with water, it dramatically affects the performance and desired outcome of the USV. To prevent this from happening a removable plastic covering was used, as seen in Figure 2.11. Made from re-sealable plastic bags, the attachment points were permanently adhered to the USV, and the cover could be press fit on the top. It is important to note this is not completely water-tight, as there are small openings in the back for the thruster wires. This plastic shield greatly reduced water from splashing in during turbulent waters.

A polypropylene rope frame was attached to the boat to provide a quick connection point for the retrieval rope during testing. This retrieval rope was used for testing purposes only and is not a part of the final design in the USV. It was used for pulling the USV back to shore if any driving failure were to occur. A photo of the final prototype out in field testing can be seen in Figure 2.12.

Figure 2.12. Final prototype floating in Lake Harner. The USV floats at the desired level, such that the thrusters are completely submerged in the water. The retrieval tether seen is a 0.5-inch polypropylene rope, used to pull the USV to shore in case of failure during testing.

2.6 Programming Automation Logic

The USV firmware is programmed in C/C++, using an Arduino Mega 2560. The automation process can be divided into 3 different phases from startup: the calibration phase, the orientation phase, and the auto-drive phase.

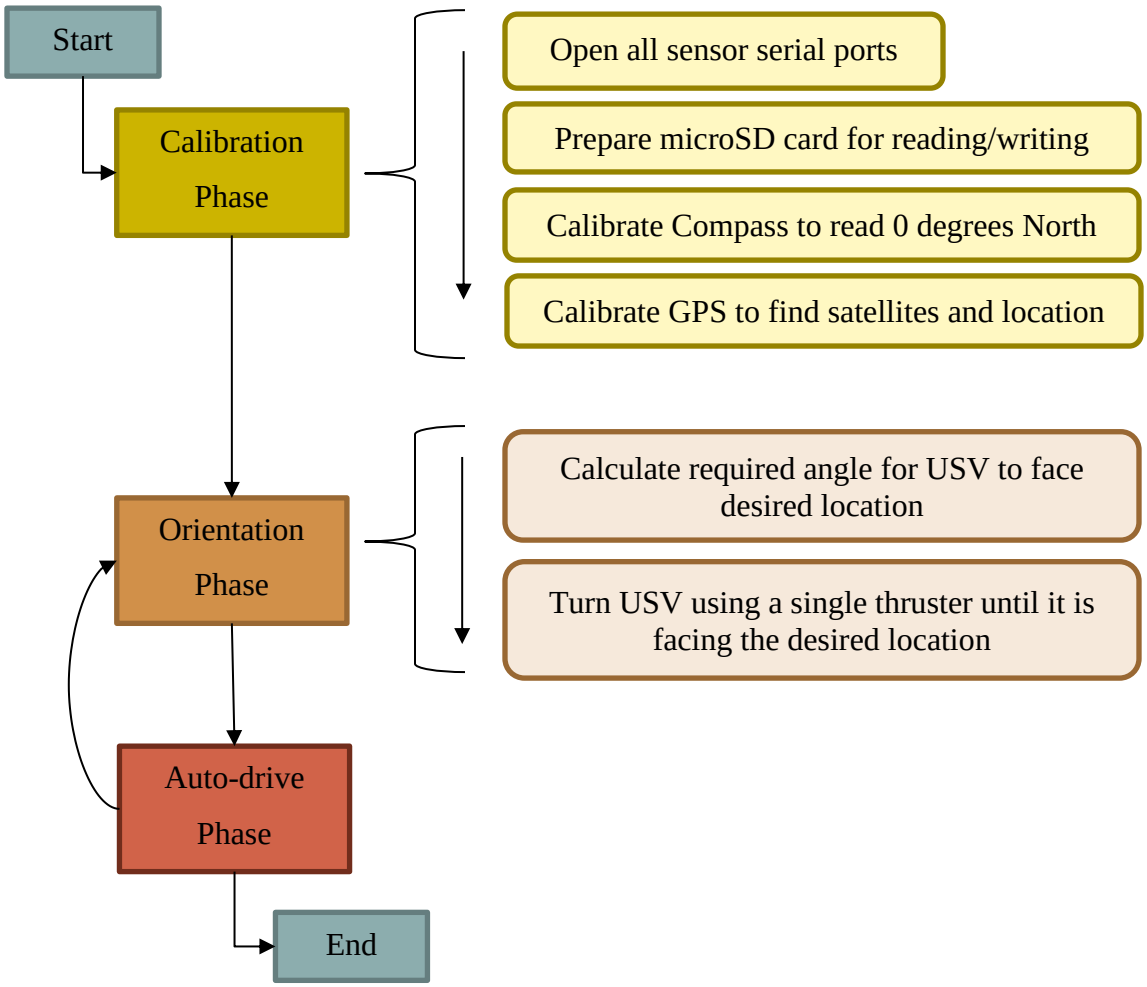

Figure 2.13. Phase breakdown of programming logic. The calibration and orientation phases have simple steps and can be explained in order. The auto-drive phase is more complex and described in Figure 2.14. The orientation phase and auto-drive phase repeat for each driving path described in Figure 2.7. After the last path is complete, the program terminates and waits for the next user-input command.

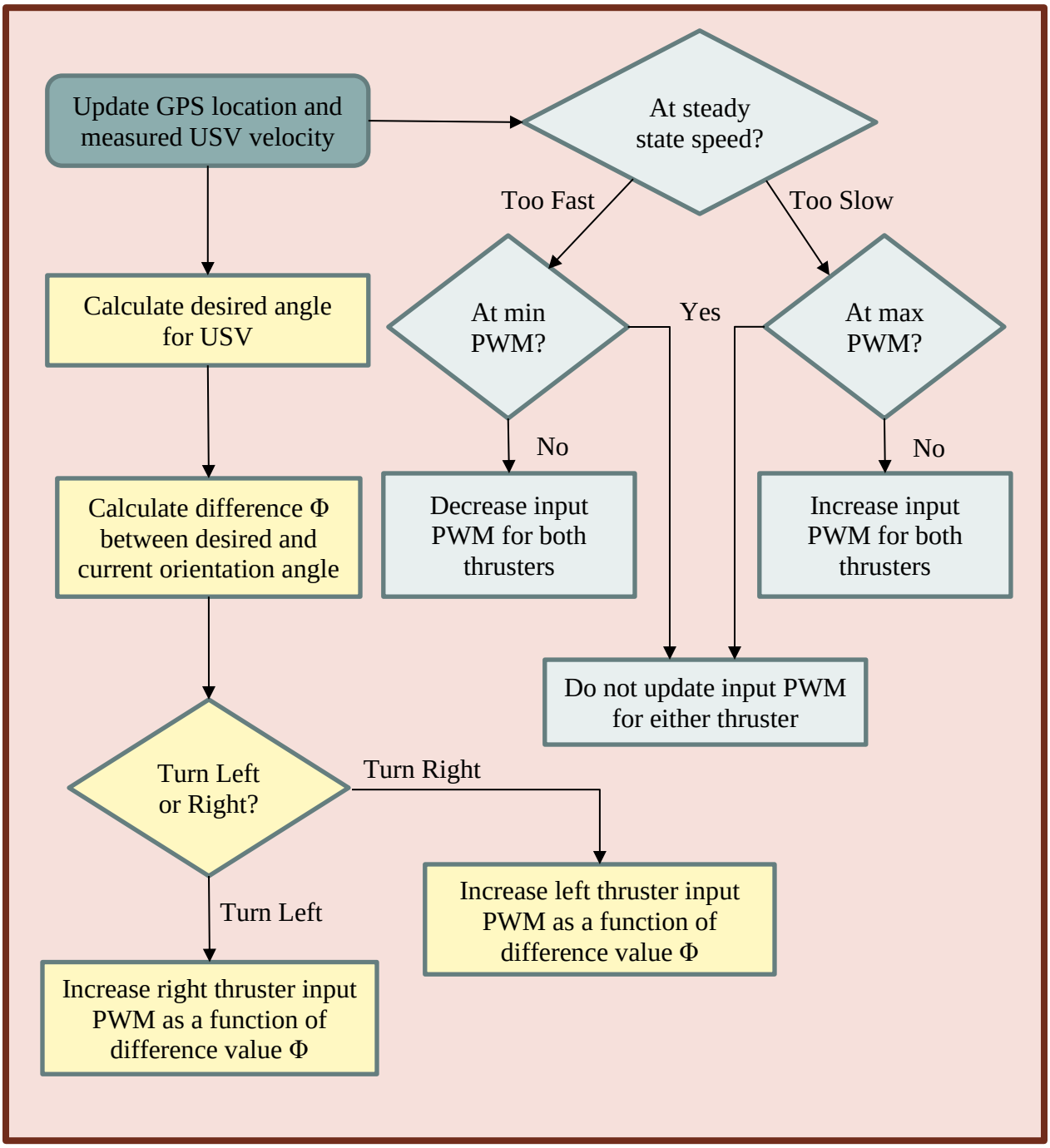

Figure 2.14. Detailed overview of Auto-drive logic. The top left (dark blue) updates measurements every 1 Hz. Each path diverting from the top left executes on a timer. The turning logic (yellow) updates at 3.3Hz, and the speed logic (blue) updates at 1Hz. The angle Φ is the difference between the desired angle and measured compass angle, used for turning logic. The speed logic has limits to input PWM values to prevent thruster damage.

Since the Auto-drive phase is more complex than the other phases in Figure 2.13, the separated Figure 2.14 shows a more detailed flow diagram. Because we are only using one microcontroller, there is only one CPU. This means, theoretically, only one task can be executed at a discrete time. To accomplish this, execution timers were used to separate the frequency the microcontroller updates each task. The data collected and written to the microSD card is on an execution timer to log at 3.3 Hz while the Auto-drive phase is running. The turning logic is set to update at 3.3 Hz, and the GPS and velocity is updated every 1 Hz.

The algorithm used for determining which way the boat should turn is determined by calculating the difference between the desired angle and the measured angle from the compass.

$$\Phi = desired\ \angle - measured\ \angle \qquad \text{Equation 2.7}$$

Using this angular difference, Φ, we can determine which direction to turn the USV based on if the value is positive or negative. Examples of this logic are shown in Figure 2.15.

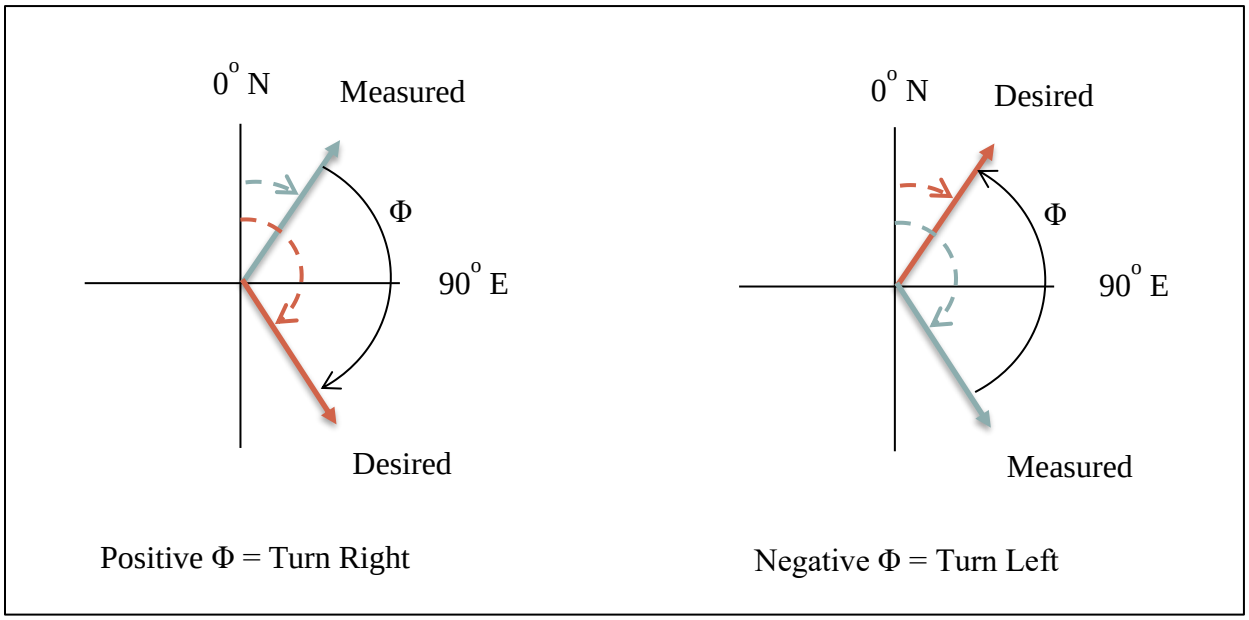

Figure 2.15. Basic turning logic example. The first case (left) shows the resulting Φ is positive, indicating a right turn. The second case (right) shows the resulting Φ is negative, indicating a left turn.

The goal of the USV is to turn in the direction that is closer to the desired angle, resulting in the shorter turning path. However, there are two unique cases when the logic in Figure 2.15 fails. Case

1 occurs when the desired and measured angles are close to, but on opposite sides, of the 0° and 360° limits. Case 2 occurs when the difference between the desired and measured angles, Φ, is over 180°. In both cases, the USV should be turning the opposite direction than the sign of Φ implies. To account for these unique situations, a common condition is found such that |Φ| will be greater than 180°. Therefore, knowing the sign and magnitude of Φ, we can fully define the direction to turn in every situation. These more complex examples can be seen in Figure 2.16.

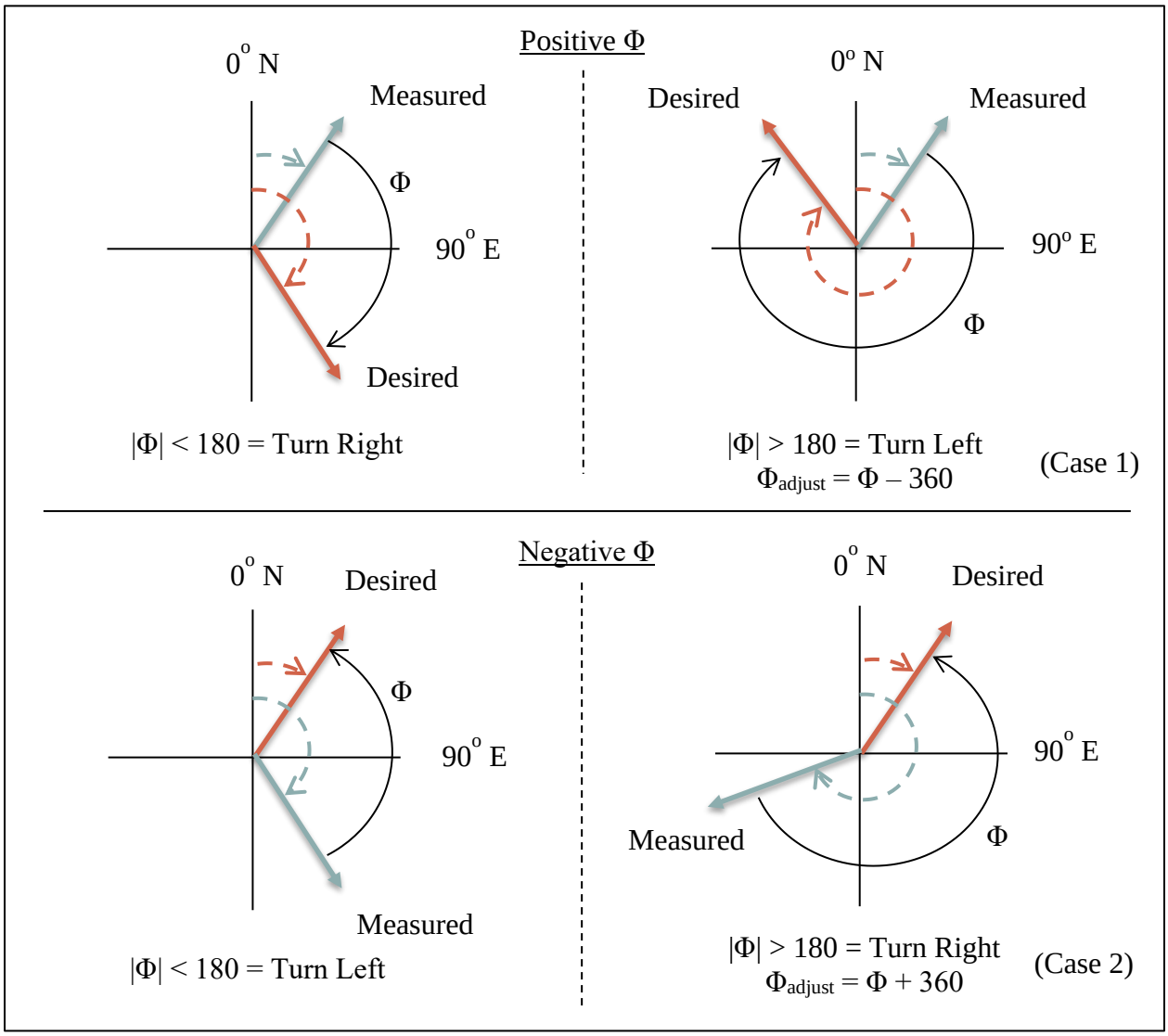

Figure 2.16. Fully defined conditions for turning logic. The first special case (top right) shows the desired and measured angles being on opposite sides of the 0° – 360° limits, resulting in a positive Φ, but with a magnitude greater than 180°. This indicates a left turn, opposite of simple logic in Figure 2.15. The second special case (case 2) creates a similar issue, however both turning directions can be determined by knowing the sign and magnitude of Φ.

It is necessary that Φ be adjusted if the magnitude is calculated to be greater than 180°. This is because the turning logic is a function of how far off the USV's angle is from the desired, meaning the further the USV is facing the wrong direction, the larger corrective input PWM is.

$$PWM_{input} = input_{ss} + floor\left(\left(\frac{2\Phi}{5}\right) * 5\right) \qquad \text{Equation 2.8}$$

Equation 2.8 is executed for only one thruster at a time, depending on which way it should turn. For example, if the USV should turn left, the left thruster will only have the $input_{ss}$ term; but the right thruster will have the $input_{ss}$ PWM plus the additional *floor* term. This larger thrust input to the right thruster will cause the USV to turn left.

In addition, comparing Equation 2.8 to Case 1 in Figure 2.16, it is apparent how important the Φ adjustment is. The equation is executed until the difference is over 45 degrees, where the adjusted PWM is at a fixed offset of 100. It took iterative testing to find that the turning logic works best when multiplying the difference by 2 and adjusting by intervals of 5. This acts as a proportional feedback controller with a gain of 2.

The algorithm for controlling the speed of the USV is determined by the measured speed of the GPS. Both the right and left thrusters have an initial steady state input PWM to drive slow and straight, which we call the $input_{ss}$ PWM as seen in Equation 2.8. The steady state PWM is then increased or decreased to maintain a desired speed of 2 knots (2.3 mph). The chosen range of acceptable steady state input values is 1500 – 1710 µs, based on the specifications of the T200 thrusters. Driving below 1500 µs results in the thrusters driving backwards; and driving above 1710 µs results in a max turning PWM of 1810 (78% Duty Cycle) based on Equation 2.8. A more detailed view of the T200 PWM limits can be seen in Figure 2.17, provided by BlueRobotics.

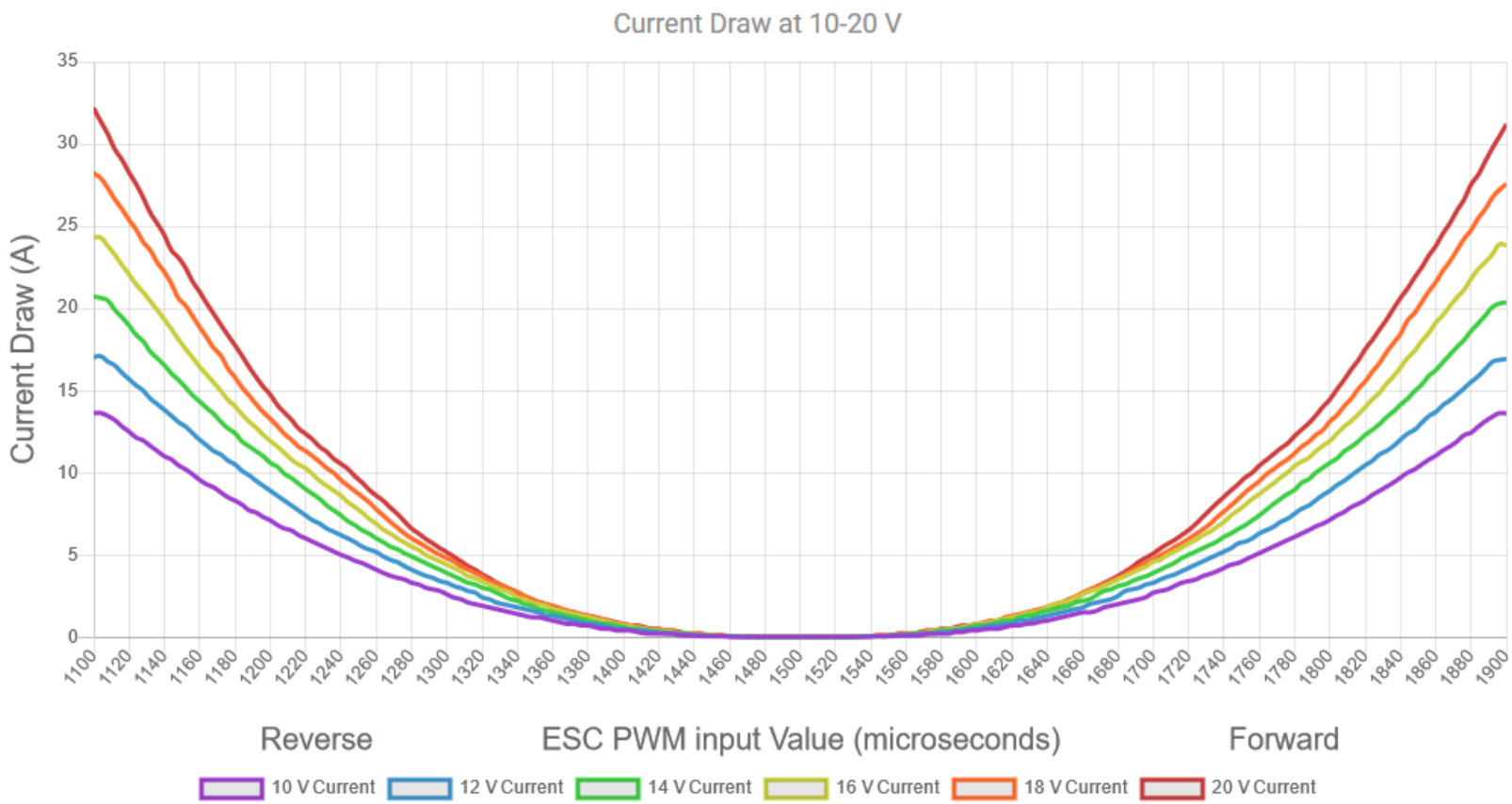

Figure 2.17. BlueRobotics T200 thruster current draw vs input PWM [23]. An input PWM of 1500 µs is off for the thrusters. Below 1500 µs to 1100µs is reverse. Above 1500 µs to 1900 µs is forward. Focusing on the 1500-1900 µs range corresponds to a 0-100% duty cycle.

2.7 Field Experiments

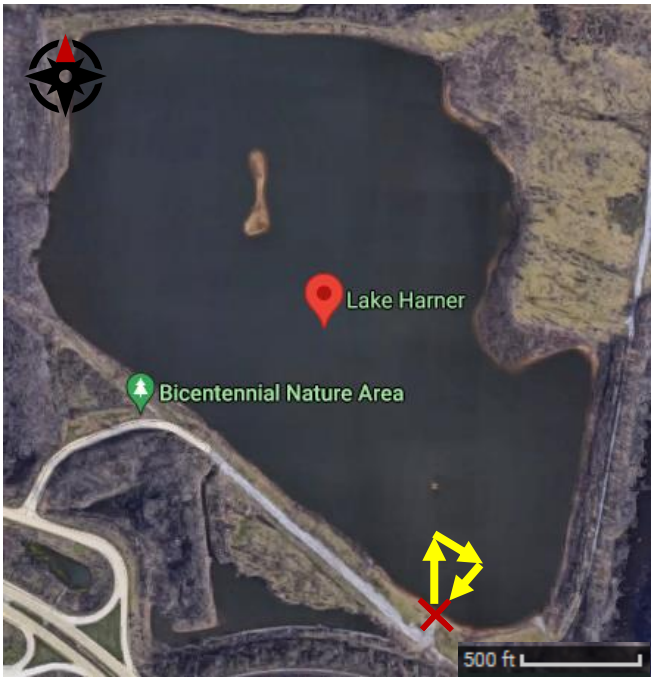

Figure 2.18. Lake Harner, Indiana [24]. The marked point indicates the start location of the tested route, with North pointing up. Only a small portion of the lake was used for testing. The total driving route was about 200 ft extending from the shore, whereas the scale shows 500ft. The full route can be seen in Figure 3.2.

Field tests were performed in two locations. The first was at Lake Harner in Indiana, where experimentation for turning logic, data acquisition, and wireless communication was tested. This included programming the auto-drive phase of the boat, which drives the USV out to the multiple input desired locations in successive, straight paths. For example, the user will input locations in the order A, B, and C for the USV to drive to. The USV will then drive from start to point A, trying to maintain a straight path. Then from point A, it will drive to point B, then to point C, then back to start. All testing was done with a 0.5-inch polypropylene rope attached for retrieving the USV if testing failed.

Figure 2.19. Wabash River at Davis Ferry Park, Indiana [25]. The marked point indicates the start location, with North pointing up. The river current is estimated to be about 1.7 mph flowing West. The full route tested can be seen in Figure 3.4.

After achieving the desired results in both calm and choppy waters, testing was moved to the Wabash River at Davis Ferry Park in Indiana. Here, testing included programming the speed of the boat in moving water. It is desired to keep the USV moving at a similar speed throughout the river. We expect the thrusters to drive harder directly against the river current and almost shut off when riding the river current downstream to conserve battery life. Moving waters led to more difficult controls testing, as the polypropylene rope created a significant amount of drag for the USV due to the river current pulling it in the opposite direction. The rope was replaced with 100 lbf fishing line to reduce effects of drag, however this still introduced external forces that the USV could not fully correct for. True testing for the boat's driving firmware would require no tether to be attached so it can drive without adding any unnecessary external forces, however it is not safe to do so without some future method of retrieval not performed in this study.

Using the onboard GPS, compass, sonar, and microcontroller, the following measurements could be made and saved as a .txt file in the microSD card.

```
Time (ms), Battery Voltage (V), Water Depth (mm), Latitude (N), Longitude (E), Speed (knots), Input ss PWM (us)
298465, 15.6788, 403, 4028.5258, -8652.1220, 0.25, 1585
298766, 15.6788, 406, 4028.5258, -8652.1220, 0.25, 1585
299067, 15.6985, 400, 4028.5258, -8652.1220, 0.25, 1585
299369, 15.6197, 410, 4028.5258, -8652.1220, 0.29, 1595
299670, 15.8955, 415, 4028.5258, -8652.1220, 0.29, 1595
299971, 15.6197, 428, 4028.5258, -8652.1220, 0.29, 1595
300272, 15.5606, 465, 4028.5258, -8652.1220, 0.29, 1595
```

Figure 2.20. Example of data saved on microSD card. All data is manually transferred from the microSD card to a computer, where it is analyzed in MATLAB. Data is sampled at 3.3Hz. This is the maximum frequency allowed for collecting data on the used Arduino Mega 2560. Higher sampling rates result in backlog errors.

This .txt file is then imported to MATLAB where post analysis is performed. Note we are collecting the time, USV voltage measured across the battery, water depth, latitude, longitude, speed, and the steady state input PWM used for controlling the speed. The measured battery voltage is processed by removing any statistically significant outliers (μ +/- 3σ), and a linear polynomial is fit to the data.

3. RESULTS

3.1 Lake Harner

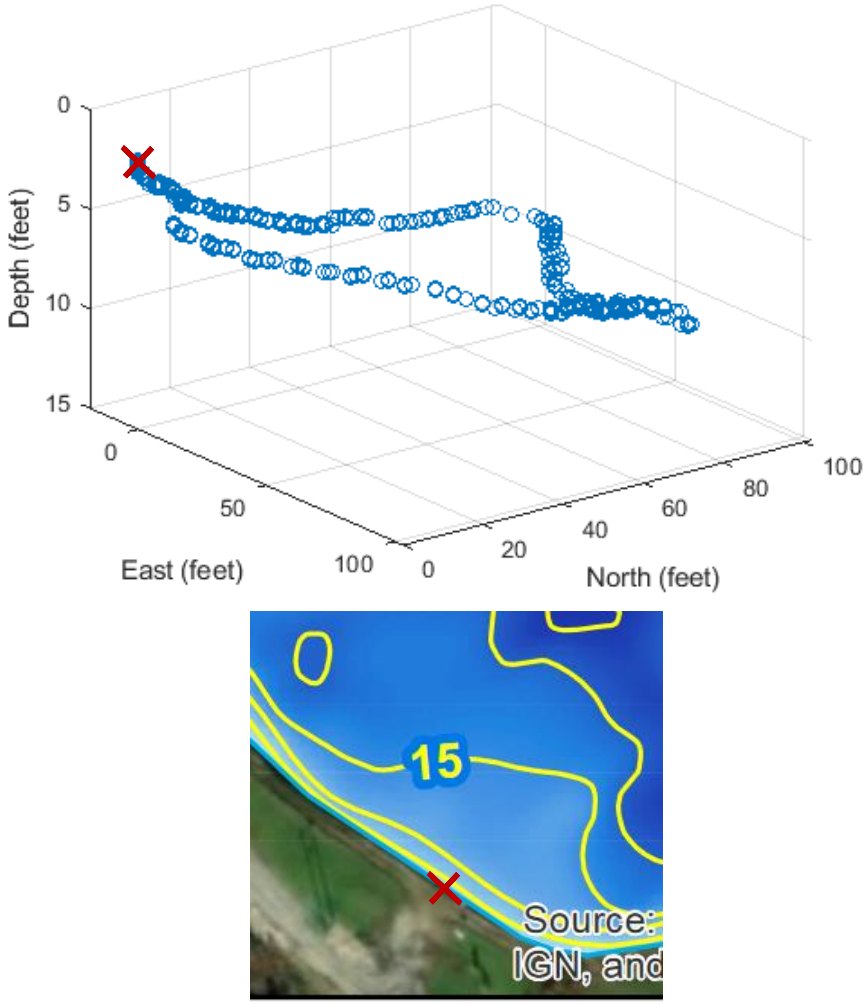

Figure 3.1. Depth of water in Lake Harner measured by sonar sensor (Top). Depth of Lake Harner with 5 ft contours reported in 2017 by Indiana Department of Natural Resources (Bottom) [26]. The depth measured by the sonar sensor closely matches the contour map, indicating the measurements are accurate. A 2D map of the USV driving route can be seen in Figure 3.2.

Initial field experiments were performed in Lake Harner to test sensors, communication systems, and turning logic. Results show all systems work as expected. Sensor data was successfully collected and stored to the microSD card at 3.3 Hz, and later analyzed in MATLAB.

Figure 3.1 shows the measured depth of the water from the USV compared to Indiana Department of Natural Resources' depth recordings. The deepest measured depth from the sonar sensor is 10 ft, and the DNR contour plot indicates a range of 10 – 15 ft in the tested area. The DNR results are from 2017, so they are not expected to match perfectly. However, similar trends and close readings indicate accurate results from the sonar sensor.

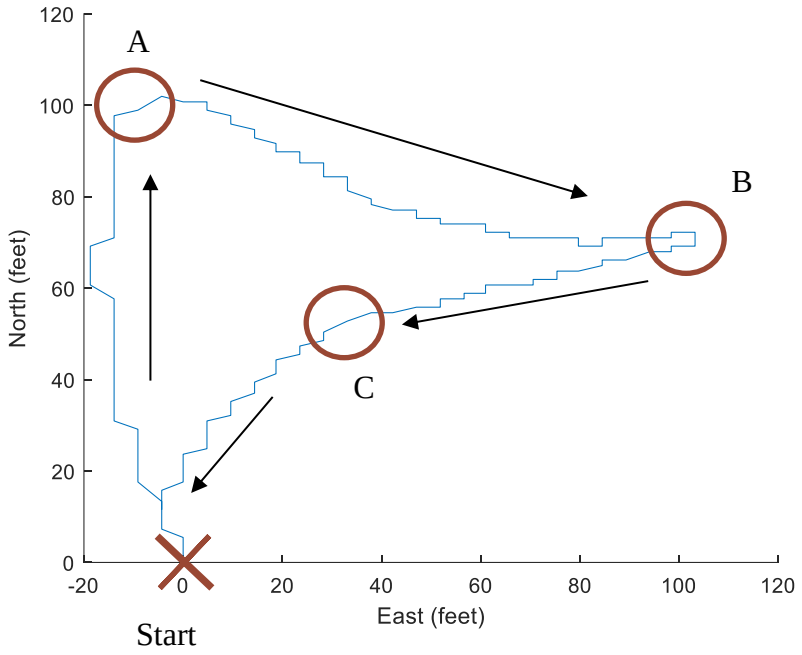

Figure 3.2. Path driven by SVU on Lake Harner. The USV reaches within 12ft of the desired locations. As the USV deviates from the desired straight path, it corrects itself using the turning logic described in Figure 2.14. The black arrows indicate the direction traveled by the USV, and the letters indicate the desired location. The scale is modified to 0 ft as the starting location and increasing North and East.

Figure 3.2 shows the USV successfully drives to each location in near straight paths with little deviation. The graph is normalized to show a starting location at 0 ft North and 0 ft East. The USV drives from the starting location to points A, B, C, and then back to start. The USV is programmed to move on to the next path after reaching within 12 ft of the desired location. This is the best accuracy we can consistently achieve due to the 6 ft resolution of the GPS. The GPS's resolution is indicated by the step like jumps in Figure 3.2.

The turning feedback logic is most prevalent in the path from points A to B in Figure 3.2. The USV starts to deviate from the path, but as it gets closer to point B, the difference angle Φ becomes larger. This results in a larger correcting input PWM to the right thruster, as expected from Equation 2.8, and the USV turns left. The deviation from the straight path is expected to be a result of wind, waves, or drag from the attached retrieval rope. It is also possible that the USV is in fact facing the desired direction but drifting in the water due to reasons previously stated. In future studies, it is desired to record the USV's heading angle to detect this behavior. The speed feedback logic is not used in these tests, as each thruster has a constant $input_{ss}$ PWM. The purpose of this experiment was strictly to troubleshoot turning logic and sensor programming.

3.2 Wabash River

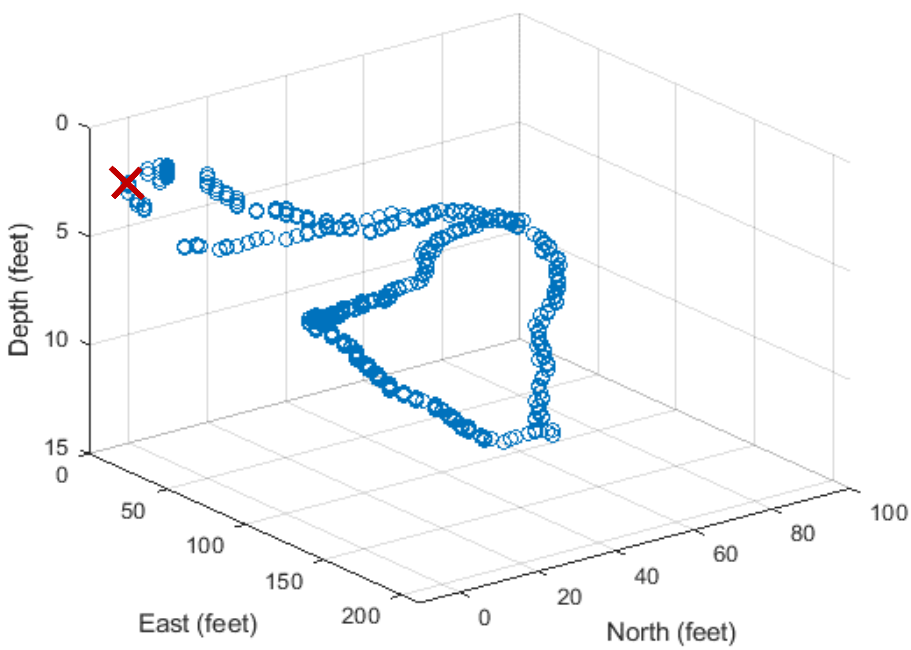

Figure 3.3. Depth of water in Wabash River. Although there is no official recorded depth of this location, the accuracy tested in Lake Harner indicates these results are also accurate. A 2D map of the USV driving route can be seen in Figure 3.4.

Field experiments in the Wabash River are performed to test the speed feedback logic, as well as the turning feedback logic in moving water. Again, sensor data was successfully collected and stored to the microSD card at 3.3 Hz and later analyzed in MATLAB. Figure 3.3 shows the

depth of the water measured from the sonar sensor. The deepest measured depth is about 10 ft, which is expected for the area. There is some difference in the measured depth between the starting and final path, likely due to a larger chance of error in shallow waters or passing solid objects in the river.

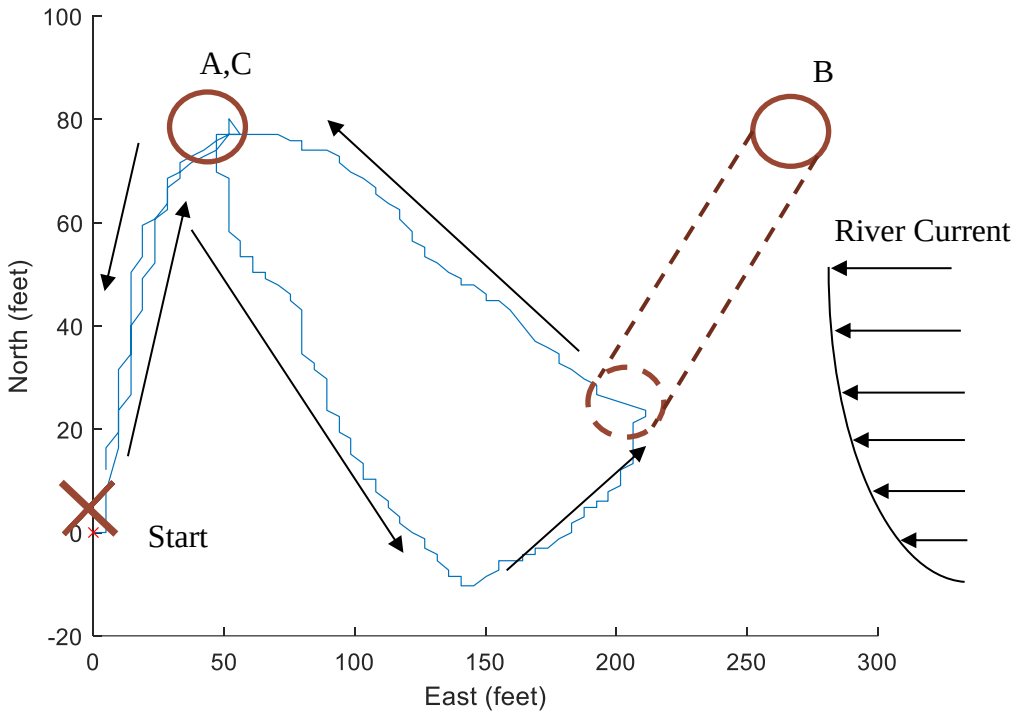

Figure 3.4. Path Driven by USV on Wabash River. The USV is manually skipped to C after struggling to make it to point B. The dotted circle attached to B indicates where the path was manually terminated. The river current arrows indicate a stronger current at the center of the river. The desired path structure was an "L" shape, however, drag from the retrieval tether made it too difficult to test the auto-drive phase up-stream.

Confirming observations during the experiment, the resulting path of the USV in the Wabash River did not yield a straight path when moving against river current. The USV was programmed to drive in straight paths from the start to point A, B, C, and back to start. Figure 3.4 shows the path from point A to B was not successful. This is predicted to be due to the retrieval tether attached for testing. Since the river current drags the retrieval tether away, there is an unaccounted-for force that is pulling the USV downstream. As the USV approaches the center of

the river, the current increases significantly, at a predicted speed of about 2-3 mph. The USV drives at full throttle but remains in the same spot, not able to drive upstream. To fix this, the user testing from the shoreline must pull in the extra slack to reduce any excess drag. This causes the boat to deviate from its original path and closer to the shoreline, as seen in Figure 3.4. Since the river current is not as strong along the shoreline and most of the slack is pulled in, the USV could then autocorrect itself on the desired path. After driving toward point B again for a short time, the increasing river current and re-added slack continued to stall the USV. It was at this point the path to point B was manually terminated by the user on the shore, due to the tether running out of slack and creating too much drag for the USV.

While there were experimental issues driving against river current, the USV does successfully drive in a straight path to all other points. It is reassuring to see not only can the USV drive in a straight path downstream, but it can also drive perpendicular to the river current. Figure 3.4 shows the USV follows almost the exact same path moving to and from the starting location.

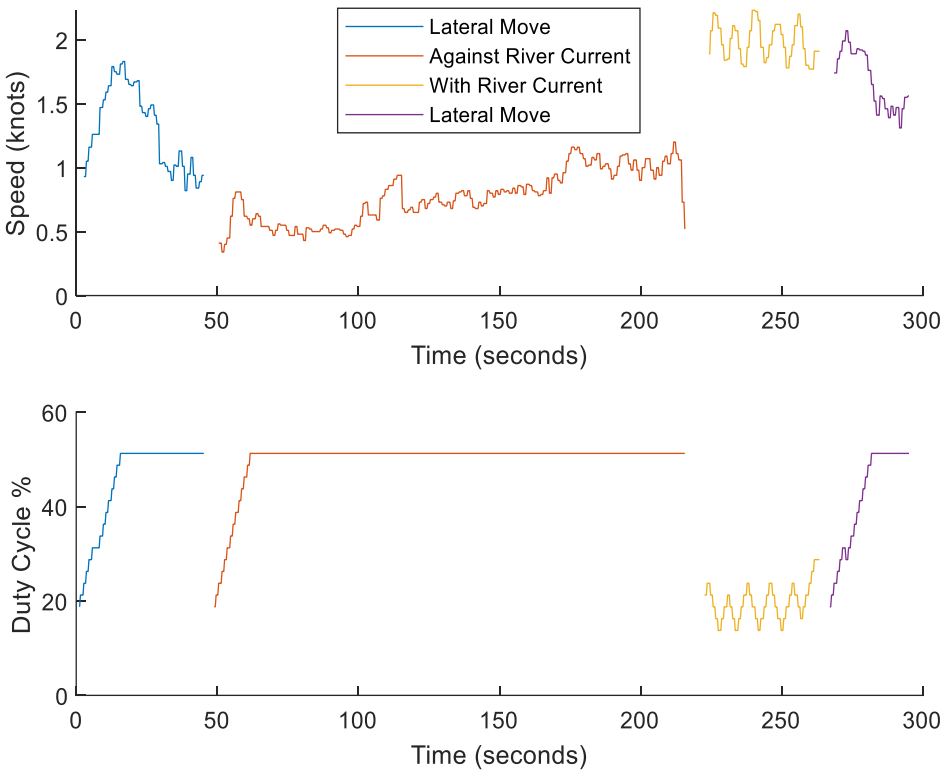

Figure 3.5. Measured USV speed and input steady state PWM. Each break/color indicates a new driving phase, after reaching each desired location. The path moving against the river current (red) was the path manually terminated. The speed and thruster duty cycle are compared to show the successful speed logic.

Although the USV did not complete the path upstream, the speed logic and power consumption can still be analyzed. Figure 3.5 shows the measured speed of the USV related to the $input_{ss}$ PWM (recorded as duty cycle %) for each path, analyzing successful speed control. The goal of 2 knots, or 2.3 mph, is achieved when possible with duty cycle limits set to 50% as full throttle, demonstrating successful speed feedback logic. It can be seen the measured speed from the GPS along the peak duty cycle is reached in the lateral moves as well as driving against river current. In these paths, the USV is permitted to drive slower, so long as the desired full throttle is maintained.

The lateral path moving to the middle of the river shows the speed ramps up to about 1.8 knots, where the USV is at full throttle with max duty cycle of 50% being reached. As it drives farther out, the river current becomes stronger and the USV slows down, as seen at the 20 second

mark in Figure 3.5. The path driving against the river current shows the USV is at full throttle yet is struggling to reach its maximum achieved speed of 1 knot. This indicates the upper limit of the speed logic is working correctly, since the USV is driving at the maximum PWM limit in an attempt to achieve the desired 2 knots. It is predicted the speed is slightly increasing over time due to the USV being pulled closer to shore, where the river current is slower. When the USV is moving with river current, the thrusters nearly shut off to conserve battery. This can be seen by the speed fluctuating around 2 knots and the duty cycle around 20%. This indicates the feedback speed logic is also working correctly, since the USV is fluctuating its $input_{ss}$ PWM to maintain the desired 2 knots. The lateral move driving back to shore is similar to that of the first lateral move. However, it is recorded to have started with an initial speed of 2 knots, likely because it is slightly driving with the current. Overall, results show the speed feedback logic works correctly.

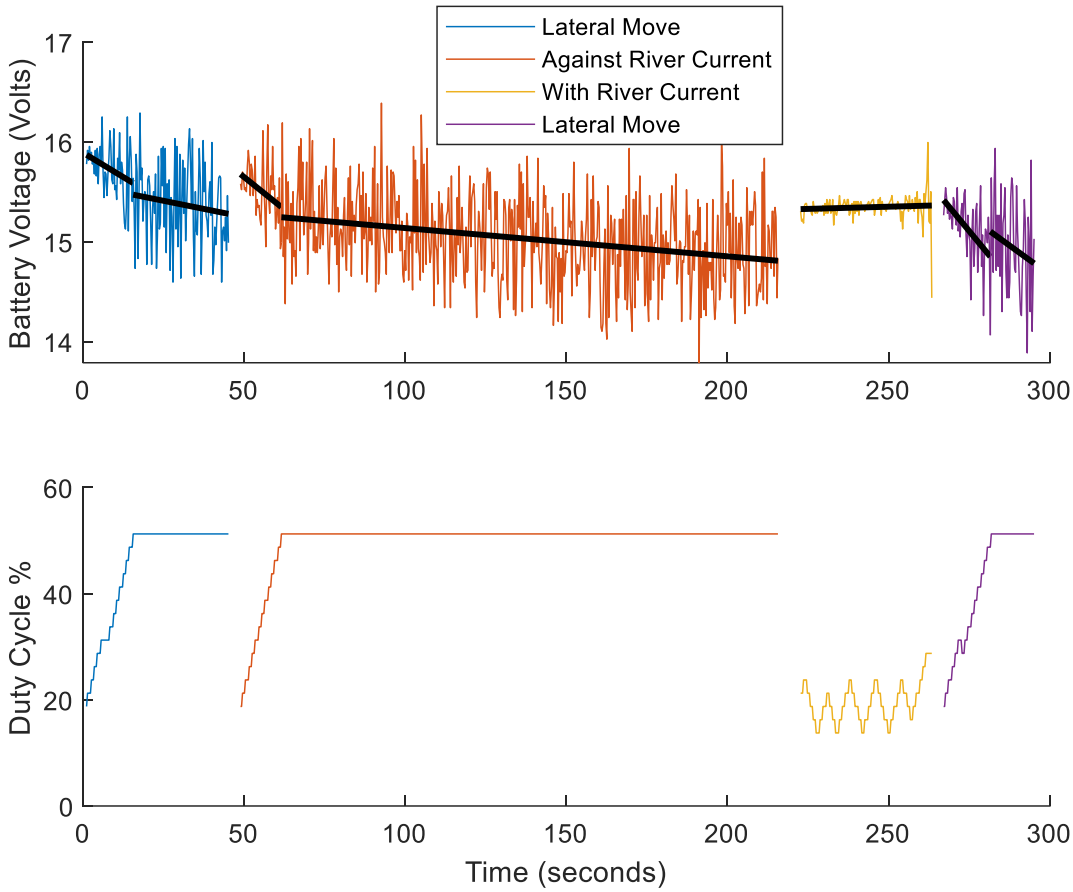

Figure 3.6. Measured Voltage across battery and input steady state PWM. Each break/color indicates a new driving phase, after reaching each desired location. The path moving against the river current (red) was the path manually terminated. The battery voltage and thruster duty cycle are compared to show the relationship between power consumption and required thrust.

Figure 3.6 shows the measured battery voltage of the USV related to the $input_{ss}$ PWM (recorded as duty cycle %) for each path, providing insight on power consumption. Looking at the lateral and against river current paths, the measured battery voltage significantly decreases over time. This can be seen by the negative slopes of the linear fit polynomial. The polynomial is separated into two sections per path, where the input duty cycle is in a transient state and steady state. The voltage drop is much larger during the transient state, confirming that increasing the load on the battery causes a decrease in output voltage. The decrease in voltage during steady state conditions is a result of decreasing battery capacity. Based on Figure 2.6, it is known that the decrease in battery voltage is related to a decrease in battery capacity. This measured decrease in voltage indicates the battery has lost some stored energy. The opposite can be seen when the USV

moves with the current, where there is almost no voltage change indicated by near flat slope of the linear fit polynomial. This indicates there is little to no change in battery capacity. This trend indicates there is less power needed and consumed when driving with the river current.

While the results confirm the USV conserves power when driving downstream, the exact quantitative amount of battery power consumed could not be accurately determined. This is due to the fluctuating battery load and not being able to map the measured voltage to an accurate battery capacity. Therefore, the results in Figure 3.6 are more used as a comparative technique in measuring relative changes in battery power consumption. Mapping battery voltage to capacity at different loads may be something to be studied in the future to obtain real-time battery life.

4. CONCLUSION / FUTURE WORK

The designed USV successfully drives in still waters using point path logic. Although it was not able to fully drive against the current in the Wabash, it is expected to work after removing the attached retrieval tether, with minimal firmware adjustments. Using a GPS and compass, we can provide enough information to control the USV to drive in a pre-determined route, with a 12 ft accuracy. Results show the turning and speed feedback control works as intended, allowing the USV to conserve power while riding the river current downstream. This logic can be generally interpreted as proportional feedback control. Also, comparing changes in battery voltage between each route confirms there is less power being consumed when the USV drives downstream. This driving logic and power-measuring technique may be incorporated in other USVs to monitor efficiency and cover larger sampling areas.

With more time, there are a few improvements that can be done with this specific USV experiment. The first would be to obtain a kayak or some other boat to be able to retrieve the USV without having to attach a tether. This would eliminate any unnecessary external forces applied to the system. In addition to this, water would still frequently splash into the top of the boat, adding too much weight to the front end and sometimes causing the USV to nosedive. While the boat still floats when completely submerged, it does not drive as intended. Creating a more watertight splash shield while still allowing access to the electronics would likely fix this issue. Lastly, improved battery voltage measurements/analysis could yield more specific, accurate results for power consumption. Mapping the measured battery voltage to predicted load could be a worthwhile investigation to allow real-time quantification in battery consumption. Looking at ways to further reduce noise from voltage measurements could also improve results.

Many future studies can also be done with this USV prototype. These include collecting water samples, automated docking, obstacle avoidance, recharging battery techniques in water, or even communicating with other robots in the water or moving on shore. Focusing on conserving power, it would be interesting to explore passive methods for collecting water samples. One idea would be to have a hitch that can wind and unwind using the rivers current to deploy sampling tubes. Automated docking could also be studied using the current USV. Since the GPS is only accurate to 6 ft, pinpointing an exact location to dock would require some additional sensor(s) and control. Having the dock flash a red and green light and equipping a photosensitive sensor to the

USV could potentially recognize and drive precisely to the dock. Placing multiple communication towers along the shore or including position measurements from the MEMS accelerometer could also improve USV position accuracy. USVs are still relatively new, and there is still much to be discovered. Combining past and present research to continuously improve USVs is leading to an impressive field of water robots, ultimately improving the lives of our communities.

REFERENCES

[1] NOAA, "30 Years of Restoring Waterways After Pollution | Damage Assessment, Remediation, and Restoration Program," *DARRP*, 2021. https://darrp.noaa.gov/history-program/30-years-restoring-waterways-after-pollution (accessed Mar. 16, 2021).

[2] F. R. Jennifer and J. A. Clark, "The Water We Drink," *PennState Extension*, 2013. https://extension.psu.edu/the-water-we-drink (accessed Mar. 16, 2021).

[3] K. Bianco, R. M. Albano, S. S. A. de Oliveira, A. P. A. Nascimento, T. dos Santos, and M. M. Clementino, "Possible health impacts due to animal and human fecal pollution in water intended for drinking water supply of Rio de Janeiro, Brazil," *J. Water Supply Res. Technol. - AQUA*, vol. 69, no. 1, pp. 70–84, 2020, doi: 10.2166/aqua.2019.061.

[4] A. G. Affonso, C. Barbosa, and E. M. L. M. Novo, "Water quality changes in floodplain lakes due to the amazon river flood pulse: Lago grande de curuaí (Pará)," *Brazilian J. Biol.*, vol. 71, no. 3, pp. 601–610, 2011, doi: 10.1590/s1519-69842011000400004.

[5] J. Paul, "Storm water project aims to stop sewage from entering Wabash River," *WLFI*, 2021. https://www.wlfi.com/content/news/Storm-water-project-aims-to-stop-sewage-from-entering-Wabash-River-574145641.html (accessed Apr. 27, 2021).

[6] C. Badue *et al.*, "Self-driving cars: A survey," *Expert Syst. Appl.*, vol. 165, no. July 2020, p. 113816, 2021, doi: 10.1016/j.eswa.2020.113816.

[7] D. Phan *et al.*, "Intelligent energy management system for conventional autonomous vehicles," *Energy*, vol. 191, no. 2020, p. 116476, 2020, doi: 10.1016/j.energy.2019.116476.

[8] F. Zhang, O. En-Nasr, E. Litchman, and X. Tan, "Autonomous sampling of water columns using gliding robotic fish: Control algorithms and field experiments," *Proc. - IEEE Int. Conf. Robot. Autom.*, vol. 2015-June, no. June, pp. 517–522, 2015, doi: 10.1109/ICRA.2015.7139228.

[9] G. Ferri, A. Manzi, F. Fornai, F. Ciuchi, and C. Laschi, "The HydroNet ASV, a Small-Sized Autonomous Catamaran for Real-Time Monitoring of Water Quality: From Design to Missions at Sea," *Ieee J. Ocean. Eng.*, vol. 40, no. 3, pp. 710–726, 2015.

[10] T. B. Koay *et al.*, "Interactive monitoring in reservoirs using NUSwan - Preliminary field results," *Water Pract. Technol.*, vol. 12, no. 4, pp. 806–817, 2017, doi: 10.2166/wpt.2017.089.

[11] W. Wang, B. Gheneti, L. A. Mateos, F. Duarte, C. Ratti, and D. Rus, "Roboat: An Autonomous Surface Vehicle for Urban Waterways," *IEEE Int. Conf. Intell. Robot. Syst.*, pp. 6340–6347, 2019, doi: 10.1109/IROS40897.2019.8968131.

[12] "Roboat." https://roboat.org/ (accessed Apr. 28, 2021).

[13] J. Wiora, A. Kozyra, and A. Wiora, "Towards automation of measurement processes of surface water parameters by a remote-controlled catamaran," *Bull. Polish Acad. Sci. Tech. Sci.*, vol. 65, no. 3, pp. 351–359, 2017, doi: 10.1515/bpasts-2017-0039.

[14] F. Fornai, G. Ferri, A. Manzi, F. Ciuchi, F. Bartaloni, and C. Laschi, "An Autonomous Water Monitoring and Sampling System for Small-Sized ASVs," *Ieee J. Ocean. Eng.*, vol. 42, no. 1, 2017.

[15] G. Yang, X. Wu, J. Chen, B. Lu, H. Chen, and C. Zhu, "Design and Analysis of an Oil-contained Water Sampling Device for Unmanned Surface Vehicle," *Proc. 2018 IEEE 3rd Adv. Inf. Technol. Electron. Autom. Control Conf. IAEAC 2018*, vol. 2, no. Iaeac, pp. 1989–1993, 2018, doi: 10.1109/IAEAC.2018.8577578.

[16] S. Manjanna, A. Q. Li, R. N. Smith, I. Rekleitis, and G. Dudek, "Heterogeneous Multi-Robot System for Exploration and Strategic Water Sampling," *Proc. - IEEE Int. Conf. Robot. Autom.*, pp. 4873–4880, 2018, doi: 10.1109/ICRA.2018.8460759.

[17] B. Li, B. R. Page, B. Moridian, and N. Mahmoudian, "Collaborative Mission Planning for Long-Term Operation Considering Energy Limitations," *IEEE Robot. Autom. Lett.*, vol. 5, no. 3, pp. 4751–4758, 2020, doi: 10.1109/LRA.2020.3003881.

[18] G. Yi, Z. Liu, J. Q. Zhang, and J. Dong, "Research on Underactuated USV Path Following Algorithm," *Proc. 2020 IEEE 4th Inf. Technol. Networking, Electron. Autom. Control Conf. ITNEC 2020*, no. Itnec, pp. 2141–2145, 2020, doi: 10.1109/ITNEC48623.2020.9085222.

[19] B. Qu and J. J. Liu, "Design of control system of PMSM based on STM32," *Appl. Mech. Mater.*, vol. 513–517, pp. 863–866, 2014, doi: 10.4028/www.scientific.net/AMM.513-517.863.

[20] F. M. Raimondi, N. Scibilia, V. Franzitta, and A. Viola, "A innovative laterals hydro floating tilting systems for USV or surface nautical vehicles," *MTS/IEEE Ocean. 2015 - Genova Discov. Sustain. Ocean Energy a New World*, pp. 1–9, 2015, doi: 10.1109/OCEANS-Genova.2015.7271598.

[21] R. J. Gutierrez, “PLA Plastic/Material: All You Need to Know | All3DP,” *ALL3DP*, 2020. https://all3dp.com/1/pla-plastic-material-polylactic-acid/ (accessed Apr. 26, 2021).

[22] “Lipo Voltage Chart: Show the Relationship of Voltage and Capacity,” *AMPOW*, 2018. https://blog.ampow.com/lipo-voltage-chart/ (accessed May 04, 2021).

[23] “T200 Thruster for ROVs, AUVs, and marine robotics,” *BlueRobotics*. https://bluerobotics.com/store/thrusters/t100-t200-thrusters/t200-thruster-r2-rp/ (accessed May 04, 2021).

[24] “Lake Harner - Google Maps.” https://www.google.com/maps/place/Lake+Harner/@40.4508451,-86.872496,687m/data=!3m1!1e3!4m5!3m4!1s0x881302e79aa2684f:0xc27416fb921d0eb8!8m2!3d40.4508451!4d-86.8703073 (accessed May 04, 2021).

[25] “Davis Ferry Park - Google Maps.” https://www.google.com/maps/place/Davis+Ferry+Park,+N+9th+St,+Lafayette,+IN+47904/@40.4731122,-86.8719279,688m/data=!3m2!1e3!4b1!4m5!3m4!1s0x881302573dce8c09:0x62cd3d1665b05e41!8m2!3d40.4733713!4d-86.870914 (accessed May 04, 2021).

[26] “Bicentenial Nature Area Ponds Bathymetry Map,” *IDNR*. 2018.