

# UM::Autonomy's Pass and No Record COVID

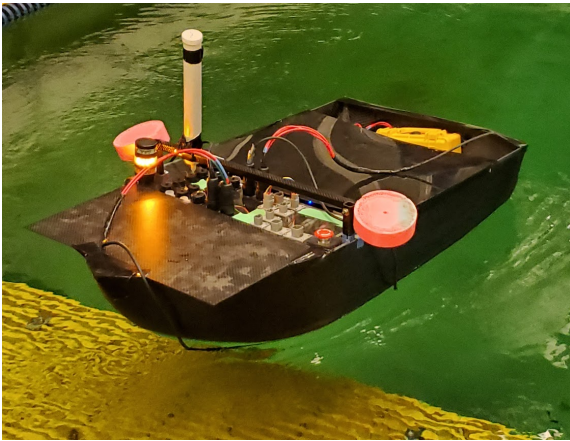
*UM::Autonomy 2020 RoboNation RoboBoat Competition Final Paper*

Rourke Pattullo, Benjamin Johnson, Alex Skillin, Rohan Mallya, Ryan Draves, Gregory Su, Matthew Maurer, Michael Biek, Albert Wei, Ryan Do, Rimaz Khan, Yi Lin Sim, Andrey Ware, Patrick Sherry, Eric Ma, Will Snedegar, Arjun Anand, Allen Ho, Samir Agarwala

*UM::Autonomy, University of Michigan College of Engineering, Ann Arbor, MI, USA*  
Submitted June 21st, 2020

## Abstract

This paper describes the development process and competition strategy of “Pass” and “No Record COVID,”<sup>1</sup> the University of Michigan’s boat and drone submission for the 2020 RoboNation RoboBoat competition. With the events of COVID-19 and the subsequent shutdown, some aspects may fall under the category of “what would’ve been.”



## Introduction

UM::Autonomy is the University of Michigan’s autonomous boat engineering build team. This is the 14th year UM::Autonomy has participated in the RoboNation RoboBoat competition. This year’s development builds off the success of last year’s major overhauls by solidifying our improvements and working towards better designs. With experimental changes to our project management strategy and build cycle, we pushed the bounds of our design process to find what worked and what needed to improve. With new hardware, a reworked sensor suite, and a modified hull form, we also looked to bring our autonomous boat design to the next level of professional

design. We believe these changes have greatly improved our ability to tackle the challenges of the RoboBoat competition and have enabled future improvements and successes.

## Core Subsystems

### Hulls and Systems

The Hulls & Systems team is responsible for the design and construction of the team’s boat each season. The Hulls & Systems team coordinates closely with all the other teams of UM::Autonomy to create a platform suitable to their needs. This includes designing around the boat’s electrical system, accompanying drone, and sensor specifications required by the boat’s autonomous systems. The Hulls & Systems team also helps facilitate physical testing of the boat.

### Electrical

The electrical team is responsible for allocating power to the boat and managing cables and wires from the battery to the computer, sensors, and motors. The electrical team also builds any other electrical equipment that is needed outside of the boat, including a base station for the boat and a server to support the AI team’s boat simulations.

### AI

The AI team is responsible for creating an intelligent system that provides the autonomy of the boat. Building off of our Robot Operating System architecture from last year, AI tasks range from integrating COTS peripherals, to object detection, navigation, top-level task reasoning, and simulating the boat’s performance in a virtual

<sup>1</sup> “Pass” and “No Record COVID” received their names from the W20 semester grading system

competition course. We use a Docker-based environment, Git for version-control, ROS Kinetic for the architecture, and C++/Python for the majority of the source code.

## Drone

The Drone team is the most multidisciplinary subteam in our organization. As the official developer for the Auxiliary Autonomous Vehicles for the RoboBoat competition, it acts as a generalist team instead of specialist, running both hardware and software design for any autonomous system that is not the boat itself. This year, it developed the new UAV “No Record COVID” to replace the former UAV “Icarus.” The codebase is built on top of the ArduPilot framework and uses a Python MAVLink API for controls, improving and expanding the functionality of our drone from the 2019 competition. Other tasks include the CAD for the physical structure for the drone and designing the electrical and sensor system that the drone uses. This team is also responsible for designing the communication interface between the ASV and the UAV.

## Design Creativity

### Team Development

The primary goal of our team is to provide a positive learning environment for our members that teaches skills not available in a classroom setting. With the successes in team development last year, we took several approaches to improve team bonding necessary for a healthy learning environment.

Our first initiative was to offer coffee chats between officers and new members. With the variety of backgrounds students enter the university with, this gives us an opportunity to learn about new members, answer questions about the team, and find a role for them that they are interested in.

Throughout the development cycle, each subteam worked towards team development. For example, the AI team experimented with project teams in the scope of a semester, rather than year-long teams for each discipline, aiming to provide an opportunity for members to engage in

different fields in AI. A more sociable example would be burger nights with the electrical team, where electrical members (and other burger enthusiasts) would gather at a local restaurant.

Overall, we believe these changes had a positive impact on the team. Many new members who engaged in the social aspects of team development have remained involved throughout the year.

### Lateral Thrusters

The 2019-20 season marked the first time UM::Autonomy has attempted to incorporate rotating thrusters into the team’s design. Two of the boat’s four thrusters were constructed with the ability to rotate 90 degrees between a forward-facing position and a lateral-facing position. The goal of this design was to achieve a higher degree of freedom available to the boat’s control system, while at the same time being able to utilize maximum thrust during the bollard pull component of the RoboBoat competition. In the lateral configuration, this setup would give us three degrees of freedom between surge, sway, and yaw, as shown in Figure 1. These motion capabilities will be further expounded upon in the Continuous Pathing section, one of the main goals of the thruster configuration.

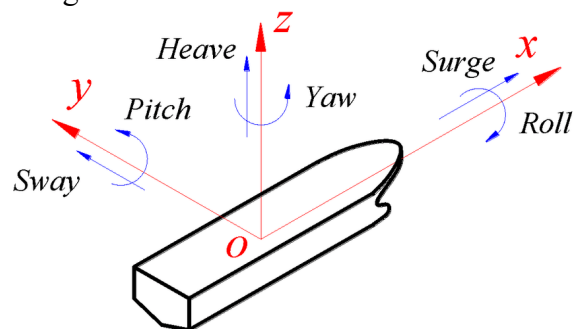


Figure 1: Diagram of rigid body motion

This design change presented a new problem. By taking significantly more space on the underside of the hull for the thrusters, the remaining space was unsuitable for the boat’s hydrophone array. The result was that the hull and hydrophone each worked to accommodate the other between strategic thruster placements and noise filtering, as described in the Acoustic Docking section.

## Vacuum Infusion

In order to expedite the carbon fiber hull fabrication process, we used vacuum infusion to fabricate our first hull, a process that has not previously been used on the team. Later, we had the opportunity to fabricate our second hull using prepreg carbon fiber and an autoclave, allowing us to achieve a lower hull weight. After finishing machining the mold, we could immediately start the vacuum infusion layup process, which took a few hours of manual work and a few days of waiting for the epoxy to reach full strength. By learning vacuum infusion, we have drastically reduced the lead time from finishing the design to having a completed hull, allowing for a shorter design and revision cycle.

Being able to fabricate carbon fiber parts through vacuum infusion also allows us to make more smaller carbon fiber parts over the course of the year with a short notice. We used the CNC router available to us on campus to machine molds from high density foam and then did a vacuum infusion layup within the span of a week.

## Continuous Pathing

To utilize the lateral thrusters, a new control system and path planner were created. Previously, we relied on a PID controller to turn to a waypoint and navigate towards it in a straight line, but this system struggled to follow curved paths. Our new system was built from the ground up to account for curved paths and the boat's physical capabilities.

The navigation stack was updated to use the Theta\* path planning algorithm (Nash, Daniel, Koenig, 2007). While A\* and Theta\* both are both grid-based algorithms, Theta\* paths are not constrained to grid-square edges like A\*. In other words, Theta\* may traverse the grid at any angle. This generates a shorter, smoother, less stair-like path.

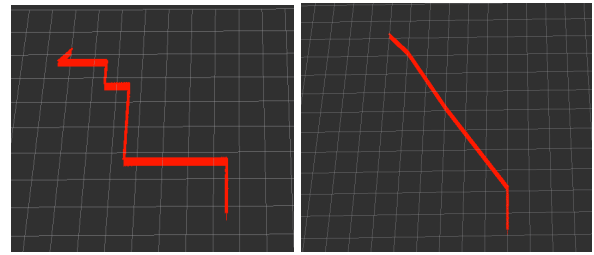


Figure 2: A\* and Theta\* paths to the same point

Once this piecewise path is generated, a splining algorithm will smooth it out into a curved path; we used a cubic spline implementation by Tino Kluge (Kluge, 2016). We found any-angle path planners like Theta\* are better suited to take advantage of spline smoothing than edge-constrained algorithms. Even after smoothing, the frequent right angles of A\* created an unnatural, inefficient route.

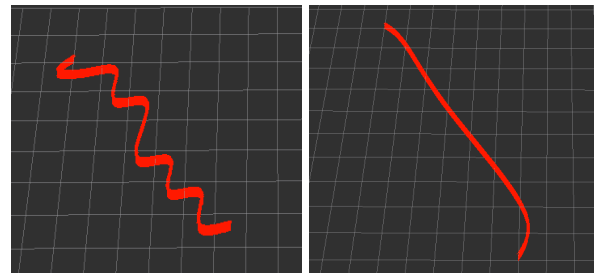


Figure 3: Spline overlaid A\* and Theta\* paths to the same point

Temporal continuity was introduced as well, where the navigation stack will continuously validate, refine, and send new paths as the boat progresses to its target waypoint.

The control stack was updated to use an LQR controller to follow these new paths<sup>2</sup>. The dynamics of the boat's motion are expressed in a set of linear differential equations, allowing us to factor quadratic drag and other naval dynamics into the control of the boat (C.E. Hann, 2010).

The result of the continuous pathing changes is a software stack that fully supports the new control capabilities of our thruster layout. Most importantly, we are now able to generate, model, and follow curved paths. Additional functionalities of the system we can make use of in our task strategies include

<sup>2</sup> Details on the LQR controller can be found at [www.cds.caltech.edu/~murray/courses/cds110/wi06/lqr.pdf](http://www.cds.caltech.edu/~murray/courses/cds110/wi06/lqr.pdf)

strafing, where we move with a fixed angle, or pointed at a fixed target, and “anchoring,” where the controller attempts to actively cancel its momentum and any forces of drift it has in order to remain stationary.

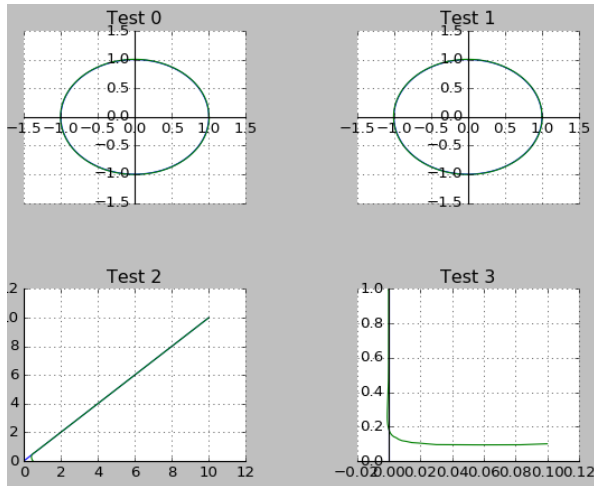


Figure 4: LQR controller testing of various paths

### Continuous Integration

This year we added continuous integration testing to verify the correctness of our code before allowing it into our official code base. One benefit of this testing is that if we make a mistake while writing code during competition, it is much more likely to be caught before it can cause real damage. This will protect us from losing valuable competition runs over a simple error. Moreover we better guarantee a good Git workflow by keeping our main branch in a working state so that developers cannot accidentally introduce errors elsewhere in the codebase.

To implement this, we added a test to our build system which runs a simulation in Gazebo of the boat inside the competition course. The boat needs to pass through the proper zones while avoiding the wrong ones in order for the test to pass. If the test fails, a note is made on our git repository and an email is sent to the person pushing the code.

### Camera Detections

This year we wanted to improve our object recognition system on our boat by supplementing our previous system, which heavily relied on LiDAR detection. Consequently, we decided to add three ZED stereo cameras to the boat and started writing

our own in-house object recognition code which could use data from these cameras to detect buoys and other objects.

The object recognition code relies on color based segmentation of buoys and is written using traditional computer vision operations such as color masking, using HSV color ranges we manually optimized. We also refined initial detections based on contour features such as contour area before performing a shape detection step for recognizing object types and filtering any remaining noise. This approach helps us use image data to extract the shape and the color of objects, the latter of which cannot be extracted from LiDAR data. A preliminary result of this system is shown in Figure 5 below.



Figure 5: Preliminary Buoy Recognition

### LiDAR Detections

In addition to the newly implemented camera-based detection code, the existing LiDAR detection system was also improved. The primary goal of the changes were to increase the overall detection accuracy and the accuracy of the extracted features. To that end, improvements were made in the classification of “circle”-like objects (possibly representing a buoy) and “square”-like objects (possibly representing a dock).

Regarding the algorithm verifying “circle”-like objects, circle fitting is applied to each of the rings of the segment to attempt to extract the center and radius. Previously, a single algebraic fit was performed using the Kasa method. This year, we added a geometric Levenberg-Marquardt fitting algorithm that would be run if the initial Kasa



fit was successful. This additional fit is designed to improve the quality of the first fit, necessary as the radii returned by the Kasa fit is often overestimated. Improving the accuracy of our radius gives us improved capability to classify circle detections as sphere buoys, cylinder buoys, and false positives.

Regarding the algorithm verifying “dock”-like objects, distance and area thresholds involving the forward ( $x$ ) and horizontal ( $y$ ) dimensions with respect to the velodyne frame were implemented. For multi-ring segments, a closest pair algorithm was run to find the pair of bottom and top ring points (with respect to the vertical dimension  $z$ ), then verification was performed using a threshold of the  $xy$  distance between those two points. For single-ring segments, verification was performed using a length threshold of the distance between the ring’s endpoints, as well as an area threshold of the segment’s  $xy$  bounding box, formed by its minimum and maximum  $x/y$  points. Additionally, to account for the (common) case that the boat is approaching the surface of the dock at an acute angle, an  $R^2$ -based optimization algorithm was implemented to locate the point on the ring that most closely represents the dock’s corner. If the segment was determined to be a dock, then the vector representing the front face of the dock, as well as the point in space representing the center of the front face of the dock, were extracted.

### Camera/LiDAR Fusion

The next step of our perception stack is detection fusion between our camera and LiDAR detections. This integration is new to our perception stack, but we found it necessary to accurately complete the obstacle channel challenge, which relies on accurate buoy detections and their associated color. The goal of this component is to follow that “the whole is greater than the sum of its parts” by not only aggregating information about an object, but also improving its overall detection accuracy.

The integration step relies on the Hungarian algorithm<sup>3</sup> in order to minimize the cost of matching a particular camera detection to a particular LiDAR detection. By overlaying the LiDAR detection into the camera’s 2D frame (and preserving distance information), the nearest objects that overlap with a camera detection boundary are associated as matching detections. From there, information is aggregated about the object, fusing positional, color, and object type information.

### Battery Hot-Swapping

Our new battery hot-swap system helps us by allowing continuous power to the boat, making it so that we don’t have to waste time turning off the computer and any power-sensitive sensors before swapping the battery. This battery hot-swap also is safer for the electrical box as any power-sensitive sensor or electrical equipment will not be turned off, and any volatile memory used in the sensors will not be lost in the process. Effectively, the AI team can continue working while the battery is swapped. This change has enabled us to maximize the value of our water testing time.

### Custom Electrical Box

This year we fabricated the electrical box from scratch, compared to the commercially available Pelican cases used in the past few years. The electrical box consists of an aluminum frame and acrylic panels, where the aluminum frame is welded from angle extrusions that make up each edge of the box, giving the box its structural strength. Laser cut panels accommodate our connectors, giving us clean cuts for square connectors. A second revision of the box was planned for the second half of the year, with the aluminum frame being welded from individual panels that are waterjet cut, for additional structural strength and ease of welding. Fabricating the custom box allows us to have an electrical box of commercially unavailable dimensions, as well as making cutting holes for connectors very easy.

<sup>3</sup> [en.wikipedia.org/wiki/Hungarian\\_algorithm](https://en.wikipedia.org/wiki/Hungarian_algorithm)

## Expanded Sensor Suite

This season, the boat's sensor suite more than doubled, replacing previous boats' one webcam with three stereo-cameras and expanding from one 360-degree LiDAR to two. The increase in visual sensors also necessitated improvements to the boat's electrical power and computer system in order to process the higher influx data. This expansion of onboard equipment resulted in an increase in the boat's weight, as the computer system and sensor suite make up the majority of the boat's overall weight.

The primary motivation for this change was to increase the field of view of the autonomous system. Previous year's boats have always featured one front-facing LIDAR and camera. With one LIDAR each on the port and starboard sides of the boat, the field of view has been expanded to cover the front, sides, and rear of the boat. This was accomplished by manipulating a computer model of each LIDAR with its viewfield to determine the mounting angle required to optimize the field of view of the LIDARs. The three cameras are also able to cover the front and sides of the boat.

As the boat's electrical sensor suite increased, we needed some graphics computing power that increased the faster GDDR6 memory capacity. An additional graphics card also allocates more computing power to the AI team, allowing for real time performance with the additional sensors. We eventually bought the Titan RTX to accommodate future sensors that might require more memory.

The computational power increase brought along several electrical limitations. First, we had to increase the overall electrical power of the boat, draining the batteries quicker. To accommodate for this, we decided to integrate the aforementioned battery hot-swap system that allowed for sustained battery power during testing periods. Additionally, we needed to add another power supply to allocate the tremendous power draw of the GPU. The Titan and its supporting power components dramatically increased the overall size of the electrical box, which

required us to stack some of the power control units on top of each other to save some space to accommodate the Titan.

## Custom Hydrophone Array and Processing



**Figure 6:** Early Hydrophone Array testing

Based on our learnings from previous attempts at acoustic docking, we sought to design an FPGA-based embedded system to handle the pinger detection, digital signal processing, and necessary calculations to identify the location of the ultrasonic pinger. To achieve this, a PCB was designed to provide a compact system for amplifying the pinger signal, filtering low-pass noise and converting the analog signal from each of our four hydrophone input channels to digitally sampled pulses. Then, we used Time Direction of Arrival (TDoA) to determine the angle of the incoming ultrasonic pulse to the hydrophone array.

A prototype board was designed using a signal-processing optimized microcontroller instead of an FPGA to reduce complexity of the first iteration. A redesign based on an FPGA could enable faster computation to allow more channels, higher sampling rate, or a higher-order digital filter to achieve greater accuracy. Initial simulations suggested angle estimates would fall within 5 degrees of the true angle, even with a sampling rate limited to 96kHz, which would sufficiently distinguish between the available docking bays. Unfortunately due to COVID-19

closures of the lab, the board has not been assembled and tested for full accuracy, and more work remains to be done to validate the accuracy claims made.

### **Marvelmind Indoor Localization System**

Since much of the academic year in Michigan is spent with frozen lakes, indoor testing is crucial to our development process. Typically however, we're limited in what we're able to test indoors, as we are unable to localize without a GPS signal only open air can provide. Since the density of our point clouds is also much higher indoors, and the lighting conditions are unique to that environment, this leaves us with virtually no software testing that can be accomplished indoors.

In order to expand our indoor testing capabilities, we purchased and set up a system of Marvelmind beacons<sup>4</sup>. By fusing positional data with our IMU, as we normally do for GPS data, we can obtain indoor localization information. The results of this is further described in the Experimental Results section, but the implications are that we would be able to test our control system and navigation system in an indoor environment, giving us a larger testing period into fundamental parts of our software stack.

### **Competition Strategy**

The order of the challenges we would attempt, or if we would skip any, was not developed, as that strategy typically unfolds during the competition itself. However, our boat and drone design were meant to allow us to complete each challenge<sup>5</sup>, so we will discuss our initial strategy for each challenge and how various design aspects would have enabled us to successfully complete the challenge.

### **Mandatory Navigation Channel**

Despite being the "just drive straight" challenge, the navigation challenge can be a rigorous test of nearly every aspect of an

autonomous surface vehicle. Our general approach was to perceive the first buoy gate, localize the boat's position relative to the buoys, chart a straight course through and past the gate, perceive the second buoy gate, and pass through the second gate.

Our redesigned hull form had a primary motivation of improving our ability to drive straight. With a more pointed bow, the hull cuts through water straighter than last year's design with a much more square bow. Before the control system ever begins to make corrections, this gives us reasonable confidence in our ability to complete the challenge in a blindly-drive-straight manner.

Next, our improved sensor suite was designed to improve our object recognition and localization capabilities. As detailed in the Expanded Sensor Suite section, additional and improved sensors and sensor positions should enhance our overall autonomous capabilities for each challenge. In the context of the navigation channel, we would reasonably expect to maintain a constant view of each set of buoy gates as we pass through them, whilst also computing our location with respect to the course obstacles.

Lastly, the introduction of our new thruster configuration and control system to support it was intended to allow for more robust course corrections. Since it can maintain the boat's surge, sway, and yaw, we would be able to seamlessly correct to the original straight line path charted through the first gate.

### **Obstacle Channel**

As a new challenge, the obstacle channel promises to deliver several unique tests to an autonomous system's capabilities. Relying on our enhanced sensor suite and new control capabilities, our approach was to perceive each set of gates and obstacle buoys in order to chart a spline through the channel for the boat to follow.

Since we currently lack the point density to accurately distinguish between A-0 and A-2 from a purely LIDAR based detection, color would need to be the deciding factor between the gates and the obstacles.

<sup>4</sup> [marvelmind.com/product/super-beacon/](http://marvelmind.com/product/super-beacon/)

<sup>5</sup> [robonationforum.vbulletin.net/forum/roboboat/-2020-roboboat/2542-rules-tasks](http://robonationforum.vbulletin.net/forum/roboboat/-2020-roboboat/2542-rules-tasks)

With that in mind, the Camera/LiDAR Fusion project was designed to assist with this task, providing perception of each field element that could accurately classify gates from obstacles.

Constructing a spline through the buoy gates and following the path draws upon our new thruster configuration and control system. Since our thruster configuration provides us with three degrees of freedom in our motion (surge, sway, and yaw), we are able to follow an ideal curved path that passes through the center of each gate while maintaining an angle to keep the next set of gates in the boat's vision. The continuous pathing project provided the generation of curved paths for the boat to follow, and the LQR controller to follow them.

### **Obstacle Field**

Many design aspects already discussed come into play for our completion of the obstacle field challenge. The improvements we made would have given us several strategy options for the challenge.

First, we had planned to test the challenge as we normally attempt it, which is to rely on our object detection and control capabilities in order to navigate through the obstacle buoys, encircle the pill buoy, and attempt to retrace our path through the obstacle buoys. This relies mainly on our object detection capabilities, which have been improved as per the camera/LiDAR detection sections. Our new thruster configuration allows us to achieve strafing motion, which could allow us to have a more nimble navigation through the field, should we get too close to obstacle buoys to continue straight.

Second, a more unique approach that we may have attempted would be to circularly strafe the pill buoy. Entry and exit would have been the same, but instead of encircling it with the obstacle at our side, we would face the buoy as we went around. This would allow us to maintain constant perception of the pill buoy and follow a closer circle away from any straying obstacle buoys. This method far exceeds our prior control

capabilities and would make full use of our continuous pathing and lateral thrusters.

### **Acoustic Docking**

On approach to the dock, we would rely on the dock detection improvements in order to recognize the front side of the dock and the positions of each docking bay. This improved perception of the dock structure should allow the boat to reasonably navigate to each segment.

Once we're in close proximity to the dock, a four-hydrophone array will use time-of-arrival based calculations to estimate the angle of the incoming signal relative to the boat's heading. Overlapping this with the docking bay locations, we would identify and dock in the bay that matches closest to the angle observed.

With the improved control system, we would engage the thrusters in an "anchor mode" to cancel our momentum, wind, and other sources of drift. Ideally, the thrusters will not significantly interfere with the hydrophone array and we could improve accuracy by taking multiple measurements from the same position. If thruster noise caused false positive pulse detections, the measurements could be taken without thrusters instead, but based on previously collected data a 25kHz high-pass filter adequately reduces the effect of thruster noise.

### **Object Delivery**

With last year's development of our drone, we felt reasonably comfortably about redesigning the drone for the object delivery challenge. We committed to a drone-only approach, in which we would try to improve its interoperability with the boat in order to deliver the object payload and attempt a marine landing on the top of the boat.

From the boat, we wanted to improve the information we provide to the drone. The improved dock detection capabilities were also meant to assist in this task by providing the drone an accurate GPS coordinate to initially travel to.



We felt that, because of the weight of our drone and the risk of in-air failure, one of our most important design considerations was the stability of the physical structure during flight. With this concern in mind, we researched various drone design philosophies, and elected to modify the basis of our frame from an X-frame to a custom H-frame. H-frames are generally inferior at high-precision maneuvering in the air, however it happens to be more laterally stable during flight, which makes slight errors in flight control more forgiving to its payload, since it is more likely to re-stabilize itself. Because our application does not require complicated midair maneuvers, we decided that the additional stability afforded by an H-frame would be a superior structure for “No Record COVID.”

In addition, our previous drone, “Icarus,” was generally hampered by a lack of deck space for its electronic payload, resulting in a very vertical, top-heavy configuration. Because an H-frame can use its entire center bar for carrying electronics, instead of a small central square, an H-frame drone also gives us more space, allowing us to design a more stable, safer drone.

Lastly, an important aspect of a UAV-based attempt is ensuring safety and having a fallback plan. At any point after the launch of the drone, we designed its logic to have consistent failure assessments that would abort the challenge and move to a safer ground landing. As a sanity check, we also included feedback information from the drone to boat about its status, allowing the boat to remotely trigger the abort system if it detects a point of failure.

### **Speed Gates**

Our general approach to speed gates is to utilize the design aspects discussed in the prior sections. Since the challenge is meant to be completed at a higher speed than we would for the rest of the challenge, our plan was to confirm the position of the mark buoy before crossing the gates and throttling up. On the return, we would slow down as necessary to

relocalize with the buoy gate and exit the challenge.

### **Return to Dock**

Lastly, for the return to dock, we would one again rely on the systems discussed to return to our starting position. By collecting GPS points throughout the competition as to where each challenge is, we would simplify the process of navigating around them on the return to dock (in addition to active obstacle avoidance).

## **Experimental Results**

### **Simulation Testing**

Simulation testing provides a crucial role in our software development. Aside from a substitute for warm waters in the winter, various forms of testing provide continuous feedback on features in development.

Sensor playback, unit tests, and Gazebo task simulation are our main forms of simulation testing. Rosbags, collected from prior in-water testing sessions, allow us to visualize the performance of individual systems on real-world data, such as camera and LiDAR object detection. Unit tests allow for automated testing of specific routines. Most importantly, our Gazebo task simulations allow for system-level tests of the boat’s functionality, putting the boat through a virtual competition course. Both unit tests and Gazebo task simulations are incorporated into our Continuous Integration system.

As for the results of these tests, we can confidently say that the codebase is healthy and maintained. Each component of our system is thoroughly tested throughout its development and even more so as it’s integrated into our codebase. Our task simulations successfully pass their benchmarks. In the course of a normal year, we would be ready for more thorough in-water testing to reveal problems not shown by the ideal settings of a simulator.

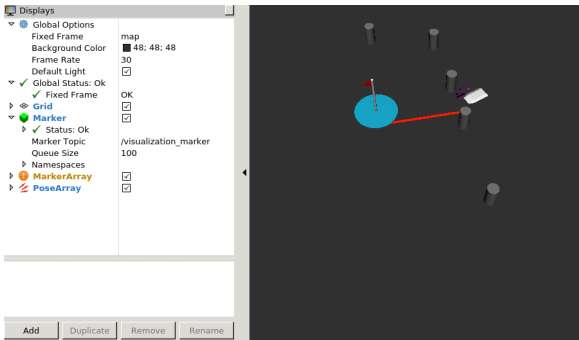


Figure 7: Rviz visualization of a Gazebo simulation

## Indoor Water Testing

Our indoor testing period had been cut short by the shutdown. We had been able to test the physical aspects of the boat, finding much of it to our satisfaction, but were cut short before software testing went underway.

Indoor testing normally provides an early opportunity for crucial feedback on the physical construction of the boat, as well as the testing of basic software systems. With the new Marvelmind indoor position system, we had hoped to expand our indoor testing period to include more control system and navigation tests, as well as full system tests of sphere buoy tasks, such as the mandatory navigation channel, obstacle channel, and speed gate challenges.

Based on the initial successes of our hardware testing in the water, we believe we would have had an extensive period of time to test our software systems before the academic year was finished.

## Outdoor Water Testing

Outdoor water testing is the irreplaceable full-systems check of our autonomous boat. Outside of the competition period, we normally dedicate a week after the academic year towards team bonding and outdoor water testing. Additionally, if time, warm weather, and our development cycle permit it, we will test on the campus pond.

As with indoor water testing, this period was cut short, and we were not able to test this year's boat as we wanted to. However, we had completed an outdoor testing period early in the year, in which we outfitted last year's hull with our new thruster configuration. This allowed us to confirm the motion capabilities we were designing

around, as well as optimize the thruster placement (in accordance to "Daedalus" and "Pass" both being monohulls).

## Conclusion

Although our development cycle and competition period was disappointingly cut short, we believe to have made many important improvements to our design. Building off the successes of last year's major overhauls, we focused on pushing our designs to the next level of professionalism. We are proud of what we accomplished throughout the year, and we eagerly await the next opportunity to compete in the Roboat competition.

## Acknowledgements

The existence and success of our team depends on the incredible support of the University of Michigan, our advisor, our committed alumni, and our industry sponsors. We must highlight the incredible support we receive from the Ford Motor Company and hands-on coaching provided by OffShore Spars in the manufacturing of our carbon fiber hull. A special thanks to Professor Kevin Maki for being the team's advisor and his mentorship in naval architecture, Tony Lockwood at Ford AV LLC for his mentorship in software project management, Assistant Professor David Fouhey for his insight into computer vision, and Dr. Kurt Metzger for advising on the hydrophone processing project.

## References

- C. E. Hann, H. Sirisena and N. Wongvanich, "Simplified modeling approach to system identification of nonlinear boat dynamics," Proceedings of the 2010 American Control Conference, Baltimore, MD, 2010, pp. 5218-5223, doi: 10.1109/ACC.2010.5530459.
- Genstattu. "What is [the] difference between X-Frame and H-Frame?" (2017, December 12). Retrieved from [www.genstattu.com/blog/what-is-difference-between-x-frame-and-h-frame/](http://www.genstattu.com/blog/what-is-difference-between-x-frame-and-h-frame/).
- Kluge, T. Cubic Spline Interpolation in C++. (2016, July 24). Retrieved from [kluge.in-chemnitz.de/opensource/spline/](http://kluge.in-chemnitz.de/opensource/spline/).
- Nash, A. , Daniel, K. , Koenig, S. and Felner, A. ( 2007). Theta\*: any-angle path planning on grids. Menlo Park, CA, AAAI Press , pp. 1177-1183.

## Appendix A

Component	Vendor	Model/Type	Specs	Cost (if new)
ASV Hull form/platform	Internal/Ford/OffShore Spars	Custom Monohull	5 lb, 8 oz Shell	\$500 (Prepreg) \$800 (Vacuum Infusion)
Waterproof Connectors	Souriau/Deutsch	UTS Trim Trio/Deutsch DT	IP68/69K	
Propulsion	Blue Robotics	T-200	Max Forward Thrust: 11.2 lb	
Power System	Custom	M4-ATX		
Motor Controllers	Blue Robotics	BESC30-R3		
CPU	Intel	i7-8700K / i7-8086K	Six-Core, 3.7GHz, 12 MB Cache	
GPU	Nvidia	Titan RTX	<a href="#">Specs</a>	\$2000 (x2)
Teleoperation	Ubiquiti	RocketM5	5Ghz	
Inertial Measurement Unit (IMU)/GPS	VectorNav	VN-300	<a href="#">Specs</a>	\$3000
LiDAR	Velodyne	VLP-16	Accuracy: 3 cm Range: 100 m	Sponsored (x1)
Camera	Stereolabs	ZED camera	<a href="#">Specs</a>	\$449 (x3)
Hydrophones	Aquarian	H2a	10Hz-100Khz	
Aerial Vehicle Platform	Custom	Carbon Fiber	H-Frame	\$150
Motor and Propellers	SunnySky	X2212 980KV Multirotor	930 gf, 12.5 V, 13 A	
Power System	Turnigy	4000mAh 4S		
Motor Controllers	Turnigy	MultiStar 40A		
Companion Computer	Raspberry Pi Foundation	Raspberry Pi 3B	OctaCore, 2 GHz, 5V/4A	\$35
Neural Compute Stick	Intel	Intel Neural Compute Stick 2		\$75
Cameras	e-con Systems	See3CAM_CU135	4K USB	
Autopilot	HolyBro	Pixhawk 4		
Flight Controller	ArduPilot	Copter-3.6		
Inter-vehicle communication	Holybro	Telemetry Radio V3	100 mW Serial Mavlink	
Algorithms	Custom			
Vision	Custom (OpenCV)			
Acoustics	Custom (PCB)			



Localization and Mapping	Vectornav / Marvelmind	VN-300 / Super Beacon Drivers	Untested outdoors / +/- 2 cm indoors	\$0.00
Autonomy	Custom (ROS)			
Team Size	University of Michigan	CS, CE, EE, ME, NAME, BBA	60 Members	
Expertise Ratio (hardware vs software)	Internal		1:2	
Testing time: Simulation (Hours)			200	
Testing time: Indoor Pool testing (Hours)	Marine Hydrodynamics Lab		5	
Testing time: Outdoor Pond testing (Hours)	Ann Arbor, MI		5	