UCR RoboSub: Design and Implementation of Leviathan

University of California at Riverside

Abstract -- Leviathan is an autonomous underwater vehicle (AUV) developed by a team of undergraduate students at the University of California at Riverside (UCR). Leviathan is the third vehicle created by the team, and was completely designed and constructed in this season. The vessel is designed to be "future-proof" and extended upon by the team for the next five years. This new vehicle contains foundational competitive capabilities including ease of motion, passive stability, bi-drectional vision, and designated locations for future components. This paper details the design and implementation of Leviathan.

I. INTRODUCTION

The University of California, Riverside's RoboSub Project gives students a unique extracurricular engineering experience building an Autonomous Underwater Vehicle (AUV) to learn and apply multidisciplinary skills, the engineering design process, project management, and practical aspects of engineering not covered in course material. The team is academically diverse, representing students across nearly all engineering disciplines and several physical sciences. This year the UCR RoboSub team designed and constructed a new AUV, Leviathan, and intended to compete in the 2020 RoboSub Competition.

II. COMPETITION STRATEGY

Leviathan is designed for two core features: hydrostability and planar navigation. These features were sought to allow for more complex maneuvers and alignments than our previous AUV, Seadragon. This increased mobility will allow the team to attempt the more complex tasks of the competition, such as the torpedo task and the dropweight task. Leviathan currently is capable of completing the gate task and the buoy task, and is designed to accommodate future subsystems for the torpedo task, dropweight task, and octagon task. The team's focus for this competition was building a strong foundation for Leviathan to complete the basic mission objectives, and for the team to build upon that platform for the 2021 RoboSub competition.

III. DESIGN STRATEGY

This year the team retired Seadragon, and followed a thorough engineering process for the complete design, construction, and verification of a new AUV, Leviathan. Seadragon had been in iterative use since 2016, and its aging structure and design limitations inhibited its ability to succeed in the tasks that required careful alignment of the AUV, such as the torpedo task and the dropweight task. Leviathan's design draws upon the strengths of Seadragon, addresses Seadragon's weaknesses, and features forward-looking design elements.

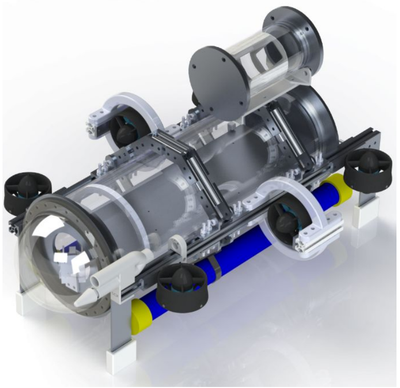

Fig 1. 2020 Design Render of Seadragon

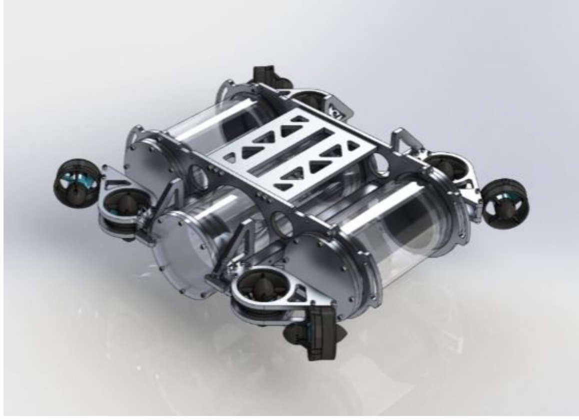

Fig 2. 2020 Design Render of Leviathan

The design changes can be broken into Mechanical, Electrical and Software subsystems.

A. Mechanical Design

Leviathan was designed around a three cylinder and eight thruster design. The geometry of the AUV is square-like for easier alignment of the center of gravity and center of buoyancy, and the team sought to optimize Leviathan being both compact and maintainable.

1) Machining

The AUV was designed for almost every metal part to be laser cut from a 1cm-thick aluminum plate. The parts were designed to require as little post-machining as possible and share uniform fasteners. Post-machining included threading holes and drilling through on some parts. 3D printed guides were made to help drill the holes evenly and correctly.

2) Electronics Chassis

The three cylinder design was intended for housing electronics, batteries, and an undesignated third subsystem. The center cylinder holds most of the electronics, including the printed circuit boards (PCBs), cameras, and computers. One of the side cylinders holds the batteries. The opposite cylinder presently serves as a counterweight for the battery, but will house future sub-systems for the torpedoes, dropweights, or pinger locator. The center cylinder is positioned as forward as possible to maximize a front camera's field of view, and to leave room for a bottom camera to see without obstruction.

1060-T Aluminum was chosen for the frame because of its ease of manufacturing, corrosion resistance, and effectiveness as a heat sink. The endplates feature a double o-ring flange with o-rings made from stock rubber cording. The grooves were dimensioned according to published SAE o-ring standards [1]. The flanges were intended to be manufactured by the team, however were instead purchased to ensure consistent machining accuracy.

3) Motors and Configuration

The motors and their configuration allow for five degrees of translational and rotational freedom. The presence of four depth thrusters allows for steady vertical motion and the four diagonal thrusters along the center of mass allow for planar motion. The AUV is designed for the center of mass to also be at the geometric center of the square profile, with the thrusters positioned accordingly to minimize undesired moments of inertia.

4) Electronics Rack

The electronics rack is a removable structure inside the center cylinder with 3D-printed mounts for PCBs, indicators, and other components. The weight of this rack helps lower the buoyancy force of the cylinder, offsetting our external ballast requirements. Aluminum bolts with central through-holes were used to pass cables through the endplates to the exterior of the cylinder.

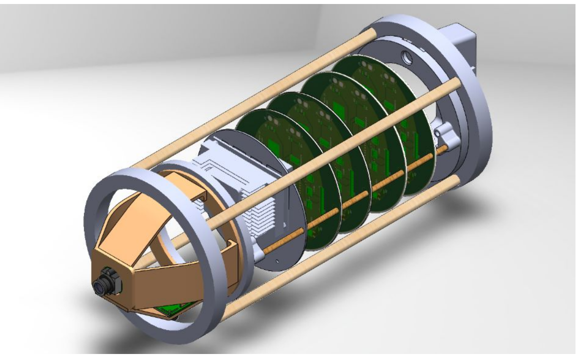

Fig 3. Electrical Rack

B. *Electrical Design*

For the 2020 competition we prototyped, designed and tested printed circuit boards for the electronics. We unit tested our hardware with our power based components, microcontroller (MCU) boards, and a Nvidia Jetson TX2 module stationed on the Orbitty carrier board.

1) *Hardware components*

Our electrical design consists of four circular PCBs, which includes a peripherals power distribution board (PDB), peripherals board, and a thruster power distribution board. They integrate with:

- (2) Teensy 3.2 MCUs
- Jetson TX2 computer module board
- Bar30 Pressure sensor
- 12V solenoid valve
- LED strip
- Altitude and Heading Reference System (AHRS)
- (2) Cameras
- (8) electronic speed controllers (ESCs) connected to (8) thrusters.

The design is powered by two 14V LiPo Batteries, of which one's voltage is stepped down by the peripherals PDB to power the MCUs and their integrated hardware. The other battery connects to the ESCs to power the thrusters.

2) *Printed Circuit Board design*

a) *MCU/Peripherals PDB*

This PCB has an array of buck converters that receives a 14V input and steps the voltage down to 5V and 12V to provide peripheral power for the MCUs and other hardware.

b) *Task Board*

This PCB holds one teensy 3.2 MCU. It controls all auxiliaries. The led strip and LCD screen are used to relay information from the MCUs and Jetson to a screen. The solenoids are used to launch 2 torpedoes at a target. This MCU also communicates to the thruster board via USART and to the Jetson through ROS.

c) *Thruster PDB*

This PCB houses our other teensy 3.2 MCU. It distributes power to the thrusters, sends PWM signals to the ESCs that control the thrusters and also contains automotive grade MOSFETs that are used for a kill switch in conjunction with a magnetic reed switch. The teensy communicates with the Task board teensy through USART.

d) *Jetson PCB*

This PCB is used as a safety net for the Jetson TX2 module, it has a short-circuit protection circuit and a polarity protection circuit to prevent the Jetson from being powered incorrectly. Circuits were also designed allowing the AUV operator to power on and restart the Jetson CPU from outside the AUV.

C. *Software Design*

For the 2020 competition, the team integrated computer vision into the software system. Robot Operating System (ROS) was utilized for message-passing between systems. The software was written in C/C++ and Python.

1) Behavior

The mission planner was implemented as a complex state machine. A master state machine controls which competition task to perform next. This decision is based on information from the computer vision system and previously completed tasks. Once a competition task is enabled it will move through each of its states, performing a simple action in each (move forward, rotate, change depth, or shoot torpedo). Once a task is completed, it sends a message to the master state machine to move on to the next task. Messages between state machines, computer vision, and the motor control system are passed via the ROS publisher/subscriber system.

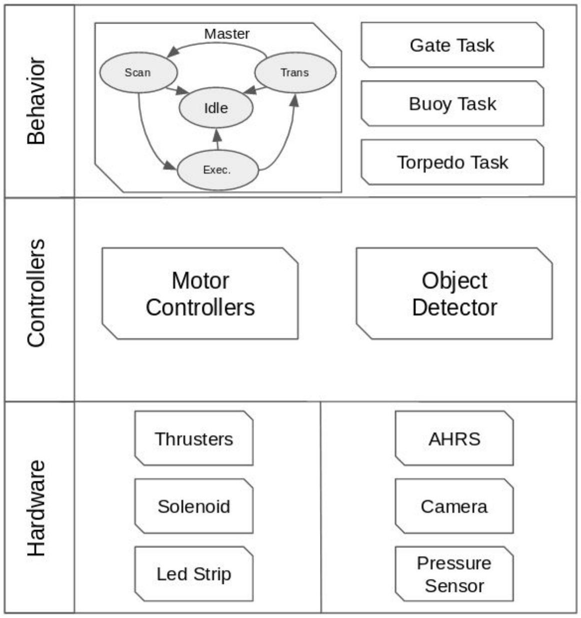

Fig 5. 2020 Software High Level Diagram

2) Computer Vision

This year the software team focused on implementing computer vision as it is critical in completing most of the competition's tasks. To achieve this, we utilized an existing deep learning framework called Darknet, in combination with the OpenCV library for image processing. We collected hundreds of images, differing in rotational and translational information, for each object that we wanted to detect. For this year, we trained a deep neural network to recognize the gate and the various images shown on the competition's obstacles. To do this, we trained our own model based off the "You Only Look Once" yoloV3-tiny model**.** This allowed us to get a sufficient frame rate and high enough accuracy for real time object detection. Of course, training this network involved the use of a video card for GPU acceleration, for which we used the CUDA toolkit.

3) Control System

Leviathan utilizes an AHRS (attitude and heading reference system) and depth sensor to provide feedback to several PID controllers. These elementary controllers allow the submarine to maintain its heading and depth. The PID controllers integrate with ROS for value storage and communication with other processes.

3) GUI

We create a Graphical User Interface (GUI) to receive telemetry from the submarine to interpret and display the data in a meaningful manner. In addition, the GUI will allow us to send precise commands to the submarine for execution.

IV. EXPERIMENTAL METHODS

1) Waterproofing: The waterproofing method is broken into several phases:

I) After the core frame and chassis improvements were done, we test for leaks by submerging the submarine at 12ft depth for thirty minutes.

II) After integrating the electrical systems, we repeat the "sink test" to ensure that the penetrators passing the electronics cables through the endcaps are watertight. To determine any leak locations of the chassis, paper towels are placed along the inside of the acrylic cylinders. When wet, the towels became noticeably darker, indicating a leak.

2) *Water Testing:* In-water software testing includes the following:
I) Connecting an onshore laptop to the electronics in the chassis, and testing the electronics with manual thruster control to ensure the power distribution boards are working properly.
II) PID tuning to ensure the motors and on-board computer have correct feedback control parameters.
III) Using constructed mock-competition obstacles to test the autonomous mission planning and feedback control systems.

3) PCB Testing: PCB testing consists of the following:
I) Unit stress testing each PCB's functionality with border cases to ensure no abnormal behavior and to ensure expected results.
II) Unit testing each PCB with long duration periods to examine the electronics' durability and heat generation due to longer-than-anticipated operation.
III) Multiple voltage inputs are tested to ensure the power distribution boards worked as intended in failure modes.

4) Software Testing
Before water-testing, we use unit tests and the Gazebo sandbox simulator to examine the reliability of the software separate from the mechanical and electrical parts. Once all unit tests for the state machines pass successfully, they are tested on the actual submarine. When the mechanical and electrical systems are tested, functional, and integrated, the software system is tested with the actual submarine in local swimming pools.

V. ACKNOWLEDGEMENTS

UCR Robosub could not be possible without the support of UCR's faculty, friends, students, and sponsors:

The team would like to thank:
- Dr. Hossny El-Sherief
- Altium
- BK Customs
- BlueRobotics
- ConnectTech
- Electronics Warehouse
- Industrial Metal Supply
- UCR's Department of Electrical and Computer Engineering
- UCR's MESA Schools Program
- UCR's IEEE Student Chapter
- UCR's ASME Student Chapter

VI. REFERENCES

[1] "Cross Section & Groove Design Data." O-Ring Cross Section & O-Ring Groove Design Data. N.p., n.d. Web. 21 June 2017.

Appendix A: Expectations

Below is the scoring table showing the points associated with each task. Enter the points you expect to score with the vehicle(s) that you have designed and engineered. At the end of the competition, enter the points you actually scored in the last column.

Subjective Measures

	Maximum Points	Expected Points	Points Scored
Utility of team website	50	50	
Technical Merit	150	100	
Written Style	50	45	
Capability for Autonomous Behavior	100	80	
Creativity in System Design	100	80	
Team Uniform	10 (N/A)	N/A	
Team Video	50	0	
Pre-Qualifying Video	100 (N/A)	N/A	
Discretionary points	40	10	
Total	650 (540)	365	

Performance Measures

	Maximum Points	Expected Points	Points Scored
Weight		-0	
Marker/Torpedo overweight		-0	
Gate: Pass through	50	50	
Gate: Maintain fixed heading	150	150	
Gate: Coin flip	300	300	
Gate: pass through 60% section	200	200	
Gate: pass through 40% section	400	0	

Gate: style	+100 (8x max)	800	
Collect Pickup: Crucifix, Garlic	400 / object	0	
Follow the “Path” (2 total)	100 / segment	0	
Slay Vampires: Any, Called	300, 600	600	
Drop Garlic: Open, Closed	700, 1000 / marker (2 + pickup)	0	
Drop Garlic: Move Arm	500	0	
Stake through Heart: Open Oval, Cover Oval, Sm Heart	800, 1000, 1200 / torpedo (max 2)	0	
Stake through Heart: Move level	400	0	
Stake through Heart: Bonus - Cover Oval, Sm Heart	500	0	
Expose to Sunlight: Surface in Area	1000	0	
Expose to Sunlight: Surface with object	400 / object	0	
Expose to Sunlight: Open coffin	400	0	
Expose to Sunlight: Drop pickup	200 / object (Crucifix only)	0	
Random Pinger first task	500	0	
Random Pinger second task	1500	0	
Inter-vehicle communication	1000	0	
Finish the mission with T minutes	Tx100	0	

Appendix B: Component Specifications

In the past, a detailed list of components constituted the bulk of many paper submissions. This practice is discouraged as it distracts from the underlying strategic thinking, system engineering decisions, or novel contributions. For the record, teams should list the components actually used in the vehicle in the table below.

Component	Vendor	Model/Type	Specs	Cost (if new)
Buoyancy Control		N/A		
Frame	Industrial Metal Supply	Aluminum T1060		$62
Waterproof Housing	Tap Plastics	Acrylic Cylinder		$80
Waterproof Connectors	Fisher Connectors			Reused
Thrusters	Blue Robotics	T100 / T200	Purchased 2 T200	$362
Motor Control	Afro ESC			Reused
High Level Control		N/A		
Actuators		N/A		
Propellers		N/A		
Battery	Pulse Battery		14.8V, 35C, 6600mAh	$94
Converter	(homemade)			~$35
Regulator	(homemade)			~$20
CPU	Nvidia Jetson TX2			$0 (borrowed from RoboNation)
Internal Comm Network	Robot Operating System			
External Comm Network		N/A		

Programming Language 1	Python		For state machine and general com	
Programming Language 2	C		For microcontrollers	
Compass	MyAHRS+			Reused
IMU	MyAHRS+			
DVL		N/A		
Cameras				Reused
Hydrophones		N/A		
Manipulator		N/A		
Algorithms: vision			Neural network based on DarkNet	
Algorithms: acoustics		N/A		
Algorithms: localization/mapping		N/A		
Algorithms: Autonomy			State machines	
Open Source Software			OpenCV, ROS, ImgLabel	
Team Size			20	
HW/SW expertise ratio			HW: 70% SW: 30%	
Testing Time: Simulation			120 hours	
Testing Time: in-water			0 hours	