Crystal Clear: RoboBoat 2023: Technical Design Report

Team Leads:
Shawn Coutinho (Captain), Manuel Roglan (Software), Aaron Wu (Electrical)
Team Members:
Jorge Ortiz Solano, Sean Fish, Chatpol Lerdthuwanon, Douglas Chin, Brisa Maneechotesuwan, Claire Harding

I. Abstract

GT MRG is a returning RoboBoat team based out of the Georgia Institute of Technology with a whole new cast of team members. This Technical Design Report aims to explain the competition strategy, design of our new Autonomous Surface Vehicle (ASV), and results from our testing. Given the lack of returning members on our team, we initially decided to attempt every task available while simultaneously building the pontoons designed by the old team three years ago. However, as the season progressed, we have reevaluated our priorities and have focused on navigation-related tasks and a racquetball launcher.

II. Competition Strategy

A. Overall Strategy and Task Selection

GTMRG's ASV, Crystal Clear, is a vehicle focused on maneuverability, thrust, and accurate navigation with a racquetball launching system attached. It is equipped with a variety of sensors, namely GPS and IMU for localization, Stereo Vision for obstacle and color detection, and 2D LIDAR to aid with perception. As such, the tasks the team focused on are those that can be accomplished solely with a drivetrain or racquetball launcher. While inroads were made initially into other tasks such as Tasks 5 and 7, work on these was paused after either rule clarifications or once the complexity of the task was better understood. In preparation for future RoboBoat competitions, the team will resume development on the requisite subsystems to accomplish these tasks reliably and efficiently during the summer of 2023.

B. Perception and Navigational Algorithms

Crystal Clear utilizes multiple forward-facing cameras in combination with a 2D LIDAR to form the core of its perception capabilities. The LIDAR effectively identifies objects in a 30° field of view in front of the ASV, centered on the center line of Crystal Clear, creating a point cloud based on surface contacts. This point cloud is then correlated with data from our stereo vision algorithm to classify contacts as buoys as well as assign colors to said contacts. This effectively allows us to create waypoints for our navigation routines to utilize such that our ASV can maneuver autonomously throughout different tasks. These algorithms are described in more detail in the Design Creativity section of this report.

C. *Task 1 & 2: Channel Navigation*

Crystal Clear's approach to both tasks 1 and 2 relies heavily on its perception algorithms and thereafter navigational ones. The robot, when turned to autonomous mode in front of Task 1, searches for a set of buoys through LiDAR contacts and, once found, verifies their identities with its stereo vision. With the LiDAR contacts now identified as the two starting buoys, Crystal Clear autonomously creates GPS waypoints for the two gates and sets a GPS waypoint in the middle. As a result of testing, it was determined that a preemptive measure was necessary at this stage given the approximate size of the field and the limited range (while maintaining a semblance of accuracy) for the second set of buoys in task 1. This resulted in Crystal Clear additionally setting a waypoint 25 feet forwards relative to its first waypoint from the direction of its initial position. This means that default movement of Crystal Clear will maintain motion forwards while searching for the second set of buoys. Once detected, the ASV verifies the identities of the new set and resets its guessed waypoint to one in between the second set of buoys, thus completing the first task.

Crystal Clear's approach to Task 2 is quite similar, aiming to identify each set of gates sequentially, however, the default movement is quite different. To avoid a situation where the ASV does not find its target and continuously moves forward, thus leaving the course, Crystal Clear's default movement once reaching each waypoint in between sets of buoys is to rotate slowly until the next set of buoys is identified. Along the way, Crystal Clear also plots the location of the black and yellow obstacles it detects to create routes avoiding as many obstacles as possible. This approach, while not as time-efficient, ensures that the ASV can achieve the maximum points possible while avoiding incurring penalties from contacting obstacles.

D. *Task 3: Color Docking*

Crystal Clear takes a rather simple approach to task 3, color docking. Once the dock has been identified through the perception system, the ASV plots a course that circles the dock, utilizing its strafing capability to continuously aim the stereo vision cameras towards the dock while progressing in describing a circle around the dock. Once the stereo vision and LiDAR verify the direction the openings to dock in are facing, the ASV will set a waypoint to one side of the dock, facing it straight on. From here, Crystal Clear will generate three waypoints that line up with each bay, assess and record the color of each, then maneuver to the one assigned to it prior to the start of the run and dock there. When at each of the three bays, Crystal Clear will employ a station-keeping algorithm such that the Stereo Vision cameras can get clear images of each banner. At this point, Crystal Clear will run an image processing algorithm on the banner to count the symbols present by isolating the color and creating bounding boxes on the distinct colored areas.

E. *Task 4: Buoy Loop*

The ASV's approach to Buoy Loop is much the same as its approach to Task 1 in that once the ASV has identified the starting buoys, it maneuvers to the center point of the two and plots a waypoint 40ft directly in front of it until the mark buoy is identified. If not identified, Crystal Clear sets another waypoint 60ft in front of this and returns to the start of the task if the mark buoy isn't identified in this range. If the mark is identified, Crystal Clear circles it using its strafing alongside forward thrust to describe a semicircle and efficiently change the direction of the ASV. It then returns to the waypoint it plotted at the start of the task.

F. *Task 5: Racquetball Collection*

This task is one of the scrapped tasks as the mechanical and software complexity of creating a grabber that could reach into a closed-off area and pull racquetballs back into the ASV in addition to a hydrophone was assessed to be far too great.

G. Task 6: Racquetball Shooting

Crystal Clear is armed with a racquetball shooter utilizing a single flywheel design with a series of feeder wheels acting in synchronization with the Stereo Vision system. Once Crystal Clear identifies the platform with the target, it circles the target with GPS waypoints until the purple frame is detected. Once here, Crystal Clear lines its racquetball shooter up with the frame and sets the RPM of the flywheel according to a velocity table for distance to target versus RPM as determined during dry testing. It then autonomously shoots the racquetballs at an interval, allowing each time for the flywheel to restore its RPM to a suitable number.

H. Task 7: Water Shooting

This task is the other scrapped one as the team's inexperience with water pumps and the physics behind them discouraged the team from proceeding with development.

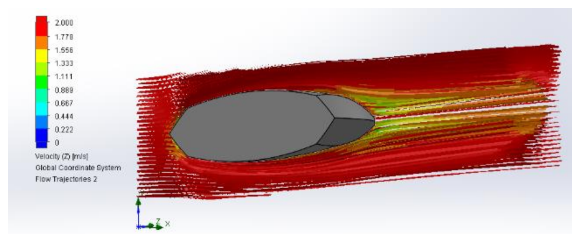
I. Task 8: Return

Crystal Clear's approach to Task 8 is quite simple due to the fact that the ASV stores all waypoints it creates while going through the course. As such, Crystal Clear navigates to its starting waypoint and from there, slowly rotates until both black buoys are identified, at which point Crystal Clear sets a waypoint in between them and finishes the course.

III. Design Creativity

A. Hull Design

The hull design is a dual pontoon configuration originally designed for RoboBoat 2020 [1]. Like previous hulls, the pontoons are constructed out of milled closed-cell foam laminated with epoxy fiberglass with aluminum extrusion rails on the top and bottom to provide attachment points. This design enables easy reconfiguration and eliminates the possibility of sinking due to a water leak. The dual pontoon configuration also allows for stability about the roll and pitch axes. The hull was in a state of partial completion, with the interior foam milled and partially assembled. The hull was completed with the addition of aluminum extrusion and two layers of hand laid fiberglass. The inclusion of a second layer of fiberglass addresses structural weaknesses seen in the previous hull's outer layer. Improvements over the team's previous entries include reduced weight and drag. The hull is also shorter than in past years to facilitate ease of transportation to testing sites.



Pontoon Hydrodynamic Testing

B. Powertrain Design

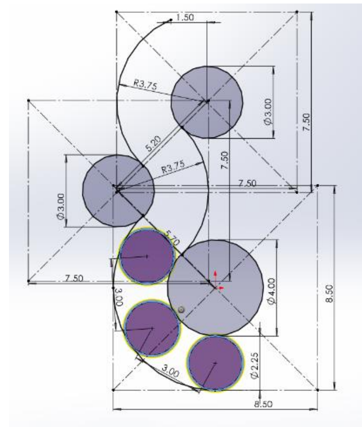
The ASV's powertrain is powered by a 14.8V 4s battery. The battery's power is split between 3 relays controlling power to the motors and a buck converter supplying 5V to the Arduino Due microcontroller. The motors are configured in an H-drive setup, with 4 corner motors facing forward, like in tank drive, and the 2 middle motors facing sideways. Each pair of motors (front, middle, back) are turned on and off by a single 60A relay. The three motor relays are controlled by a battleswitch (radio-controlled relay) to allow us to kill the motors remotely. The battleswitch is powered by the microcontroller. The remote-kill signal is received by an orangeRx receiver, which is powered through the microcontroller. There is a flyback diode placed in parallel with the battleswitch to prevent backwards current flow into the receiver. Each motor's speed is controlled by a 30A electronic speed controller. The speed controllers are configured by the Arduino microcontroller. When in autonomous mode, the microcontroller communicates with the main computer over micro-ROS to control the speed of the motors. When in manual mode, the microcontroller communicates with the remote-control transmitter to control the speed of the motors.

C. Bellypan Design

The bellypan design utilizes a lexan sheet as its backbone, focusing on minimizing weight, increasing flexibility of placement of electrical components, and general longevity. Due to the lack of additional mechanical subsystems necessary to complete all other tasks, Crystal Clear is able to utilize a plexiglass sheet as its bellypan and mounting surface. This has allowed the ASV to easily route wires to all relevant motors, batteries, and emergency stops while also increasing troubleshooting efficiency due to the visibility of all components.

D. Racquetball Shooter Design

The racquetball shooter design is inspired by traditional single flywheels with high unloaded RPM transferred to the ball at the point of contact and followed through the course of the hood. The hood for this shooter is fixed at 45° (subject to change) past horizontal, as seen below, for optimal distance and height gained when firing. The subsystem relies on distance from target data fed to it by the perception system such that the optimal flywheel RPM can be applied to control the strength of each shot. Each of the racquetballs are preloaded to a dual wheel, single motor loading system that feeds each when suitable RPM is detected on the flywheel, waiting for the flywheel to recover before feeding the next.



Racquetball Shooter Skeleton Design

E. Software

The software for the ASV is built with ROS2, an open-source environment containing various libraries and tools which greatly aid the development of robotic applications. The software stack is divided into five major packages:

1. Localization: responsible for fusing GPS and IMU information to provide us with odometry.
2. Perception: responsible for identifying and locating various objects, including buoys.
3. Navigation: responsible for path planning to a waypoint or set of waypoints.
4. Controller: responsible for translating paths into motor commands.
5. Autonomy: responsible for determining what actions the ASV should take given its current state and objective.

a. Localization

Localization is accomplished through the Robot Localization package, which allows for fusion of multiple sensors through an extended Kalman filter. The sensor inputs are GPS position, GPS velocity, IMU rate gyro angular velocity, and IMU magnetometer heading adjusted for magnetic declination.

b. Perception

For our perception, we are using multiple PlayStation Eye cameras and a 2D Hokuyo LIDAR. Using the LIDAR, we are able identify the location of objects ahead of the ASV. With the cameras on the front of the ASV, we can classify the objects. For example, a red buoy vs a black buoy. Together, we can use the sensors to give our ASV a better understanding of its environment than each sensor could provide independently.

Over the past few months we have also been working on adding stereo vision to the software stack. Stereo vision allows us the cameras to identify the location of buoys that may be below the height of the 2D LIDAR, which is the case for the smaller buoys used in competition. We are currently implementing stereo vision with two PlayStation Eye cameras spaced a known distance apart on the bow of the ASV.

c. Navigation

The core of our navigation stack is Nav2, an open-source project which allows us to change and tune our path planning algorithm. Nav2 generates a global and local costmap of the environment surrounding the ASV using the filtered LIDAR data provided by our perception package. When the ASV needs to navigate to a specific waypoint, we can send a request to Nav2 for a path. Using the information from the global costmap, Nav2 runs a Hybrid-A* Dubins path planning algorithm which generates a path for the ASV to follow. We have the ability to tune the path by adjusting parameters such as the inflation layer around obstacles detected.

d. Controller

The controller's objective is to follow the path planned by the navigation server. The lowest layer is the motor command fuser which takes in target force goals in each axis and a target torque goal and translates these to direct motor commands using differential thrust. In the event of a change of motor configuration, this is the only layer that requires changing. A PD (proportional differential) controller is used to generate target torques to achieve the commanded heading. For distances under 2 m from the goal position, a simple PID (proportional integral differential) controller based on position and velocity is used to generate target force commands. For these distances, the vehicle will maintain its goal heading and utilize holonomic translation to maintain its position. For larger distances, a PD controller is used to target a certain velocity and the vehicle's target heading is set so that it is facing in the direction of the target velocity. This velocity is generated as a sum of a vector in the direction parallel to the path and a vector perpendicular to and in the direction of the path. The further away from the path, the larger the vector perpendicular to the path is weighted. The greater the error between the vehicle's heading and target heading, the smaller the overall velocity to allow the vehicle control authority to reorient itself.

e. Autonomy

The autonomy server contains the logic for the vehicle to carry out the tasks. Each task has its own script which executes the steps necessary for the task – for example, setting a goal between two buoys identified by the perception server.

f. Firmware

The firmware for the electronics is developed from the motor control firmware developed by the team for RobotX 2022 [2]. The firmware is Arduino C++ and utilizes the micro-ROS library to communicate with the ROS2 software stack.

IV. Experimental Results

A. Simulation

The software was tested in Gazebo using the Virtual RobotX framework. The controller was tested using the built in wave and wind plugins to ensure that it could station keep in severe conditions, though the thrust and mass characteristics of the vehicle were different. Navigation and perception software were tested for the channel navigation and docking tasks. The vehicle successfully navigated a series of buoys using LIDAR and docked with LIDAR and camera.

B. Dry Testing

Dry testing was conducted to ensure that the sensors, software, and motor controller functioned individually and when integrated. This proved invaluable for troubleshooting integration issues prior to testing the vehicle in the water.

C. Pool Testing

The team tested an older vehicle with a variety of motor configurations under remote control in a pool to identify the best one. The configurations tested were two versions of 4 motor holonomic X-Drive, 4 motor tank drive, and 6 motor holonomic H-Drive. Given the lack of continuous use of all motors, the team finally chose the 6 motor H-Drive for its superior thrust and maneuverability at the cost of current draw.

D. Lake Tests

The software, propulsion, and electrical systems were tested at a lake under moderate current conditions using an old hull. The vehicle was able to successfully navigate to GPS waypoints and station keep. It was also able to identify the locations of the buoys relative to the vehicle. However, error in the heading estimation led to the absolute position being incorrect. This error will be rectified for the next lake test. In future lake tests, the vehicle's ability to complete the navigation channel test will be tested by ensuring that it can correctly detect the absolute position of the two buoys and then navigate between them. In addition, the vehicle's ability to detect the colors of the buoys will be tested. Once navigation has been proven using LIDAR as the primary sensor, binocular vision will be tested by using these algorithms to detect the positions of the buoys instead of LIDAR. This will ensure that the vehicle is able to perform the channel navigation and buoy loop tasks. In addition, a GPS waypoint saver will be tested. This waypoint saver will be used to return the vehicle to the starting location at the end of a competition run. Throughout these tests, continually tuning will be done with the controller system to adjust the PID gains to minimize overshoot while keeping movement responsive.

Acknowledgements

A. Sponsors

The Marine Robotics Group thanks all of our sponsoring organizations that have contributed to the team, whether financially, through hardware, or any other form of support demonstrated. Those sponsors and supporters include Greenzie, Fischer Connectors, Altium, Connect Tech Inc., and Pattent, Georgia Tech Student Government Association, Georgia Tech Student Organization Finance Office, and Georgia Tech Student Foundation.

B. Mentors

The team thanks Dr. Michael Steffens for being our chief adviser on the project, Tanya Ard-Smith for her help with competition logistics, Professor Dmitri Mavris and the Aerospace Systems Design Laboratory for providing for the team at large.

References

[1] A. Kazmi, A. Sinha, A. Velte, C. Ramee, D. Foreman, D. DiCarlo, E. Jahoda, E. Fu, E. Phan, J. Pattison, J. Adrande, J. Chau, L. Dorrian, M. Schlaff, M. C. Martin, M. Gilmartin, P. Meyer, P. Eddu, R. Rameshbabu, R. Kuramshin, R. Vepa, S. Fish, T. Daino, T. Campbell, and V. Chow, "Roboboat 2020: Technical design review," *RoboBoat 2020*.

[2] S. Fish, M. Roglan, D. Chin, J. Pattison, J. Ortiz Solano, "Virtuoso – A New Architecture for RobotX 2022," *RobotX 2022*.

Appendix A: Component Specifications

Component	Vendor	Model/Type	Specs	Custom/Purchased	Cost	Year of Purchase
ASV Hull Form/Platform	In-House	Dual Pontoon	4'x3' footprint	Custom	$300	2020-2023
Waterproof Connectors	Huayi-Fada Technologies LTD.	Various	IP68	Purchased	Unknown	2015
Propulsion	Blue Robotics	T200 Thruster	T200 Thruster: ROV thruster for marine robotics propulsion (bluerobotics.com)	Purchased	$200	2016
Power System	Multistar	4S 12C LiPo	4S1P, 14.8V, 10Ah	Purchased	$60	2019
Motor Controls	Arduino	Due	32 Bit; 512KB memory	Purchased	$51	2016
CPU	Intel	NUC5i7	Intel NUC Kit NUC5i7RYH Product Specifications	Purchased	$540	2016
Teleoperation	OrangeRx Spektrum	R615X DX6e transmitter	3.7-9.6 V; 2.4 GHz	Purchased	$17	2016
Compass	ArduSimple	ArduSimple RTK	simpleRTK2B receivers & affordable multiband RTK antennas - ArduSimple	Donated	$619	2020
Inertial Measurement Unit (IMU)	LORD MicroStrain	3DM-GX3-25	3DM-GX3® -25 - , (microstrain.com)	Purchased	$2640	2017
Doppler Velocity Logger (DVL)	-	-	-	-	-	-
Camera(s)	Playstation	Eye	640 x 480 60 fps	Purchased	$43.99	2023
Hydrophones	-	-	-	-	-	-
Algorithms	Nav2 Package	-	-	-	-	-
Vision	OpenCV Library	-	-	-	-	-

Localization and Mapping	Robot Localization	-	-	-	-	-
Autonomy	Internally Developed	-	-	-	-	-
Open-Source Software	See Virtuoso Repository	-	Virtuoso GitHub	-	-	-