Embry-Riddle Aeronautical University

Robot for Advanced Intelligent Navigation

15 May 2021

Team Captain

Andrew Strazds *strazdsa@my.erau.edu*

Team Members

Claude Watson III	*watsoc20@my.erau.edu*	Ana Alvarez	*alvara34@my.erau.edu*
Joseph Corry	*corryj2@my.erau.edu*	Zachary Moser	*moserz@my.erau.edu*
Katie Lane	*lanek9@my.erau.edu*	Gabriel Alkire	*alkireg@my.erau.edu*

Faculty Advisors

Charles Reinholtz	*reinholc@erau.edu*
Patrick Currier	*currierp@erau.edu*
Eric Coyle	*coylee1@erau.edu*
Christine Walck	*daileyc1@erau.edu*

Required Faculty Advisor Statement

I certify that the engineering design of the vehicle described in this report has been significant, and that two of the team members have earned four semester hours of senior design credit for their work on this project.

Charles F. Reinholtz, Department of Mechanical Engineering

CONDUCT OF THE DESIGN PROCESS, TEAM IDENTIFICATION, & TEAM ORGANIZATION

Introduction

For the 28th Annual Intelligent Ground Vehicle Competition, the Auto-Nav team from Embry-Riddle Aeronautical University – Daytona Beach has significantly improved upon their entry from the previous competition. Robot for Advanced Intelligent Navigation, or RAIN, is an autonomous vehicle designed to self-navigate unknown environments of varying terrain with minimal setup and no user input during operation. Upgrades include significant improvements to the electrical and mechanical systems, as well as a complete redesign of the software system to simplify operation and fit more closely with current military and industry practice. The conversion from LabVIEW-based code on all previous ERAU-DB entries to ROS-based operations on the current platform has been the most notable innovation of the 2020 entry. Modularity was a point of focus to facilitate tuning during competition, which for the first time will be conducted on a paved course, as well as enhance future expansions. Safety has been the team's highest priority, both in strictly observing COVID-19 protocols during development and to ensure the safety of personnel during operation.

Organization

The ERAU Auto-Nav team is comprised entirely of Mechanical Engineering undergraduate students. The chart below provides the area of concentration for each member as it pertains to RAIN. To better streamline task completion, three work divisions were created: Mechanical, Hardware, and Software. The Mechanical Division was charged with ensuring optimal component layout as well as creating CAD designs for the system and for various components. The Hardware Division was responsible for selecting sensors, processors, electronics, and drive system components. The Software Division explored the uncharted territory of creating a new ROS-based software suite in which the robot will operate.

Table 1: Team Member Areas of Concentration

Name	Major / Year	Mechanical	Hardware	Software	Hours of Contribution
Andrew Strazds	ME / Sr.	X	X	X	360+
Claude Watson III	ME / Sr.	X	X	X	250+
Ana Alvarez	ME / Soph.		X	X	70
Joseph Corry	ME / Sr.	X	X		50
Katie Lane	ME / Fr.	X			30
Zachary Moser	ME / Jr.	X	X		30
Gabriel Alkire	ME / Soph.	X			25

Design Process & Assumptions

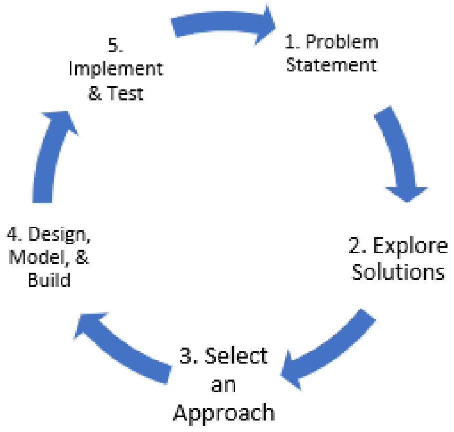


Figure 1: Design Process Approach

The team began the design process by using the five-step design cycle shown in Figure 1. At the start of the design cycle the team took note of the condition of RAIN's sensors, hardware, and software. Likewise, its performance in previous competitions and during testing was also closely reviewed. Throughout each step in the design cycle, consideration was given to each division mentioned above (Mechanical, Hardware, and Software) to ensure the best product would be presented.

To begin, a problem statement was introduced to the team and from that the customer needs were deduced. In this case, the IGVC rules formed the basis for determining the customer needs. Next, the team explored solutions by brainstorming amongst each other and reached out to experts as needed. This led to determining which components from the previous model would remain, and which would be upgraded to fulfill the customer's needs. To single out the most ideal solution, a document of pros and cons was made for each possible approach. From that, a final decision was made while considering timeliness, cost, reliability, and ease of implementation (i.e., if it was outside of our skillset). Next, designs were drafted, and models were created to help visualize the approaches to be taken. The components were then built from the final versions of the models. During the implementation and test phase, each component was closely monitored during use while checking for faults in the design for unforeseen errors. If further adaptions were required, the cycle was repeated until all the customer needs were satisfied.

EFFECTIVE INNOVATIONS

Robot Operating System

The programming interface has seen a major overhaul with the implementation of ROS. This decision was made to more closely align RAIN with industry standards while ensuring proper prioritization of the decision-making matrix. The system utilizes a "publisher/subscriber" system, meaning a node publishes data for any subscriber to receive and use. In our case, all sensors (LiDAR, Vision, IMU, etc.) are considered a publisher, while the motor controlling node will subscribe to each. Furthermore, ROS is a powerful tool which enables users to build each sensor's "decision making" within a designated node, making the software more modular and improving the dissemination of written programs. ROS's ability to work with C++ and Python ensures our ability to work with a wide array of users and sensors.

Updated LiDAR

The Velodyne VLP16 Puck Hi-Res uses 16 channels (laser beams) for a reliable mapping of the surrounding environment up to 100 meters. The real-time LiDAR sensor provides 3D point cloud data to increase the accuracy of object detection. This was changed from the Hokuyo UTM-30LX-EW, which was only capable of 2D distance sensing at a range of 30 meters, so the VLP16 allows for more accurate data collection at longer ranges.

Modular Frame

The previous sub-frame had worked loose over time and caused instability in the operation of the robot. Instead of simply tightening the sub-frame on which the motors were mounted, it was decided that the best approach was to give more modularity to the robot while simultaneously remedying the issue of instability. This change added two pounds to the overall weight but was ultimately determined to be in the best interest of the project. With an eye for future expansions, a 2-foot section of aluminum extrusion, 8020 3030, was used to replace the previous frame. 8020 is a widely utilized material with universal mounting properties and is structurally sound. Currently, 3D-printed GPS antenna mounts are installed on the frame. Previously, the GPS antenna mounts were attached using two bolts that were common with the sub-frame. Meaning, to remove the GPS antennas, the sub-frame had to be removed as well. The addition of the 8020 also enabled a quick turn-around on adapter plates on which motors were to be mounted. This minimized the cost and time spent waiting on new motor mounts to be machined.

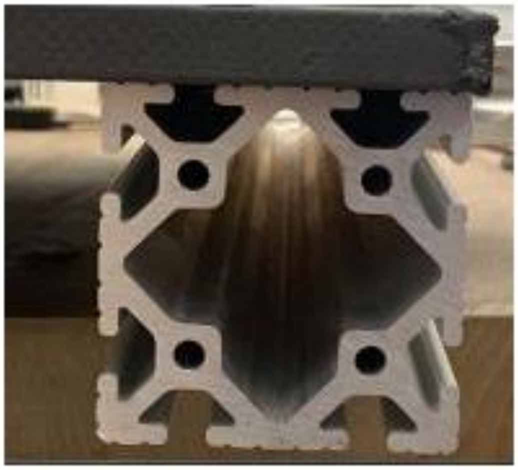


Figure 4: Extruded Aluminum Sub-Frame

Main Circuit Breaker

To enhance safety, a circuit breaker was added to the main power line from the battery. This was a major improvement compared to the previous vehicle since several electrical problems had occurred and were now resolved. The power supply used by RAIN is a 6-cell LiPo with a capacity of 6.6 Ah. Connecting and disconnecting this power source was an area of concern as it could cause harm to the system and/or connector contacts by connecting or disconnecting it to power up or power down the robot. Likewise, in case of an emergency, such as an electrical fire, where removing power quickly is paramount, it can be done more promptly and safely by hitting a button on the circuit breaker rather than pulling two tight fitting connectors apart.

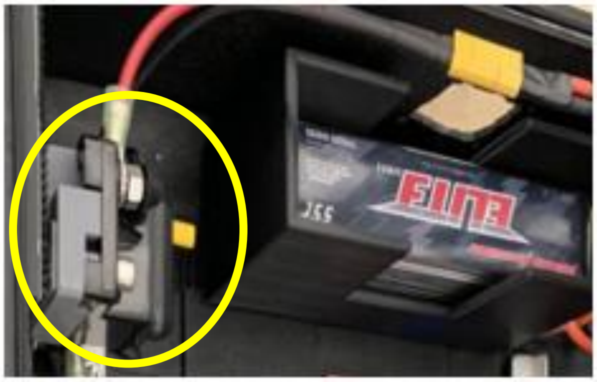


Figure 2: New Circuit Breaker

Fuse Panel

To protect the sensors and computing hardware, a fuse panel was installed to ensure each component whose power supply passes through it is protected. Should they need to be replaced, the cost of these components combined amounts to more than what a single semester's budget would allow for them to be replaced. Therefore, it was decided that protecting these vital instruments was crucial. The Blue Ocean fuse panel has six fused passthroughs as well as a grounding block for those same six components. Since the majority of the components operate on 12 volts, it was an ideal choice for use in RAIN's system. Currently, the LiDAR, Jetson TX2, 12-to-24 volt converter for the POE injector, and the camera are all fused by this panel. Using this fuse panel also makes troubleshooting and system testing easier by enabling the user to remove power to one or more components at a time.

Figure 3:Fuse Panel

DESCRIPTION OF MECHANICAL DESIGN

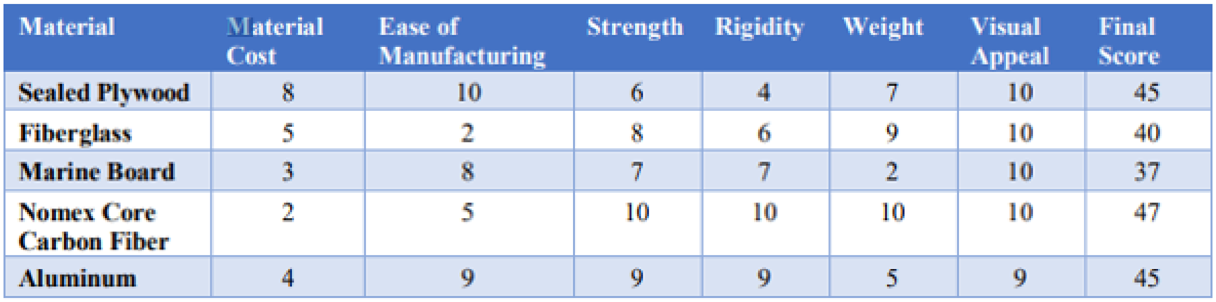
Overview

The previous version of RAIN was already a solid platform which sized well under the limits of the competition. The dimensions come in at 2'5" wide, 3'7" front to back, and 5'2" tall. It is a differentially driven robot with a caster wheel in the front which uses an eight-inch pneumatic tire. To protect the sensitive electronics, a waterproof IP65 rated Pelican Air Case was utilized. Additionally, the case makes the electronics more modular and easier to transport. The electronics' connections pass through a panel which utilizes IP65 rated connectors to allow sensor communication with the internal computing components while allowing the system to remain water resistant. The overall system is designed to be modular, allowing for easy assembly, disassembly, and transport.

Chassis

RAIN's chassis is made of a carbon fiber platform with Nomex sandwiched between the outer layers. The combination of these materials gives the robot an incredibly rigid platform while remaining lightweight. The electronics case is mounted to the chassis using two threaded studs, making assembly quick and easy. As this frame was made for the 2019 entry, it was decided that the strength-to-weight ratio could not be improved. The decision-making process for the materials of the frame is illustrated in Table 2.

Table 2:Decision Matrix for Frame Structure



Material	Material Cost	Ease of Manufacturing	Strength	Rigidity	Weight	Visual Appeal	Final Score
Sealed Plywood	8	10	6	4	7	10	45
Fiberglass	5	2	8	6	9	10	40
Marine Board	3	8	7	7	2	10	37
Nomex Core Carbon Fiber	2	5	10	10	10	10	47
Aluminum	4	9	9	9	5	9	45

Sensor Pole

The sensor pole is made of 1” square carbon fiber tubing mounted upright from the base of the frame. Carbon fiber was chosen for its rigidity, strength, and lightweight properties. The sensor pole holds the camera, LiDAR, safety lights, emergency stop button, and Ubiquiti Omni-Directional Antenna. The wires for the components are fastened to the sensor pole in a manner that allows for rapid swapping of sensors or lights. The components are fitted using 3D printed friction mounts designed to allow for secure attachment to the sensor pole. These mounts were also designed to prevent undue stress which could compromise the structural integrity of the carbon fiber, while also providing the highest amount of stability for the sensors.

Drivetrain

For the purposes of this section, the drivetrain will include the motors, their mounts and adapter plates, and the sub-frame. The sub-frame, as mentioned earlier, is a 2-foot section of double 15 series T-slot 8020 that provides a strong, rigid mounting structure. An adapter plate was machined to match the size of the inherited motor mounts and 8020 rails. The motor mounts are designed to hold the motors using friction by clamping onto the cylindrical gearbox. The motors generate the 20 in-lbs. of torque needed to reach and maintain the maximum speed limit of 5MPH while going over the ramps described in the IGVC rule document while also carrying the required 20-pound payload.

Vehicle Cost

The cost of the vehicle can be split up into two different cost models, as seen in Table 3. The first model represents the raw cost of the platform if constructed brand new. The second model represents the cost to the team for the year. Due to sponsorships and the reusability of several parts on the platform, the vehicle was of minimal cost to the team. The final cost of this system was designed to be considerably lower than other alternatives. This allowed for a versatile system on which sensors and mechanical parts can be swapped at a low cost. Overall, this allowed for a very low-cost system that is both versatile and efficient. Many of the new sensors were either donated or purchased during or prior to the canceled 2020 competition.

Table 3: Cost Analysis

Item	Unit Cost	Qty	Raw Cost	Team Cost
Jetson Tx2	$400.00	1	$400.00	$0.00
Wiring and Misc.	$200.00	1	$200.00	$200.00
Power board	$200.00	1	$200.00	$0.00
Camera	$849.00	1	$849.00	$0.00
IMU	$5,000.00	1	$5,000.00	$0.00
Lidar	$4,000.00	1	$4,000.00	$0.00
Carbon Fiber Frame	$550.00	1	$550.00	$0.00
Pelican Case	$200.00	1	$200.00	$0.00
RC transmitter	$200.00	1	$200.00	$0.00
Ubiquity Network system	$380.00	1	$380.00	$0.00
Lipo Batteries (6s)	$216.00	3	$648.00	$648.00
8020 3030 (24')	$48.00	1	$48.00	$48.00
Wheels	$30.00	2	$30.00	$0.00
Quicksilver Motors	$2,200.00	2	$4,400.00	$0.00
Total			**$17,135.00**	**$896.00**

DESCRIPTION OF ELECTRONIC AND POWER DESIGN

Overview

This year's entry has a mostly revamped power system. The skeleton structure has remained, but almost every component has been either modified or upgraded. The focus was to rely less on the custom power board and make the robot more COTS component friendly, which also reduced potential points-of-failure.

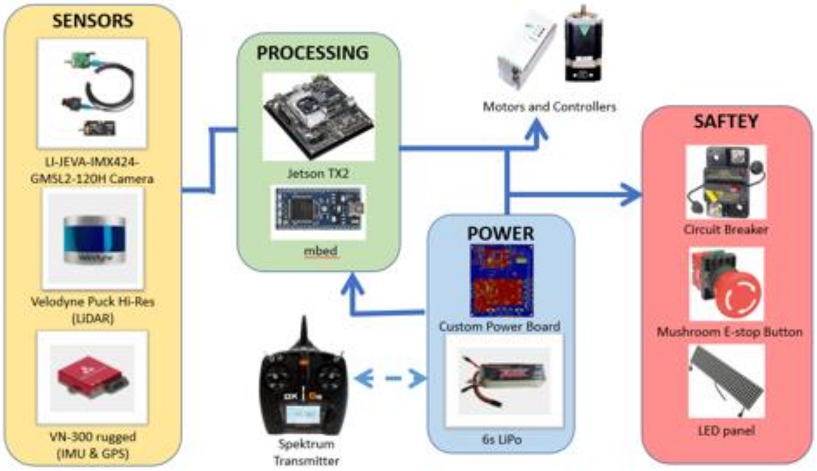


Figure 5: System Integration Diagram

Power Distribution

With the change from the laptop used in previous iterations to the Jetson TX2, the system draws more power from the power supply than the previous version, DAISI-C. Therefore, larger capacity batteries were purchased to compensate for the increased power demand. The new batteries are TP6600-6SE55 and allow a runtime of 1-1.5 hours, which is similar to previous iterations of the robot.

Computer

The central processor of the system uses the Jetson TX2 with the complete development board to process and communicate all data. It is a 2.0 GHz ARM processor, 8GB RAM, with a 512 GB fast-writing SD card, and 2 TB optional internal SSD hard drive for data collection. The TX2 runs Linux Ubuntu 20.04, and ROS to run the software. The majority of RAIN’s programming language is written in Python using PyCharm Community Edition IDE.

LiDAR

The Velodyne Puck Hi-Res is a 16-beam LiDAR capable of full 360 horizontal FOV with a resolution of 0.1-0.4 degrees, and a 20-degree Vertical FOV which gives 1.33 degrees between channels. The LiDAR generates 300,000 points per second, and can measure a range up to 100 m with an accuracy of ±3 cm. It is rated for IP67 environmental protection. An aviation passthrough connector was used to feed the LiDAR through the pelican case and into the Velodyne interface box.

IMU/GPS

The Vectornav VN-300 Rugged is a high-performance Dual Antenna GNSS-aided INS. The system uses two Tallysman GPS Antennas to give accurate heading measurements, along with the IMU to ensure accuracy and precision with and without the vehicle moving. The IMU collects data at 800 Hz while the GNSS has an update rate of 5hz. Sensor fusion occurs within the VN-300 itself to provide the most accurate data to the system.

Camera

RAIN’s camera is the LI-JEVA-IMX424-GMSL2-1202H. It is an industrial grade Automotive ADAS and Machine Vision camera and is meant specifically for the Jetson TX2 module. It features up to a 3849x1929 pixel resolution with a 1/1.7 format. It also features a 120-degree horizontal FOV, making it ideal for lane detection, which is its primary use.

LED Panel

The LED panel is an 8x32 LED pixel panel that displays basic operating information at a glance. This system utilizes an mbed microcontroller which will interface with the Jetson TX2 to send serial display commands using ROS. It can display basic information such as “Stop,” “Auto,” and “RC” to inform nearby pedestrians of what state the robot is currently in.

Safety Devices

RAIN incorporates a direct voltage cutoff system built into the power board as part of the safety system requirements. This system cuts off power to the motors but keeps the sensors running to avoid an extended restart time. E-stop buttons are located on the sensor pole and the RC controller. In addition to the mechanical emergency stop system, an LED panel indicates to bystanders when the system is under autonomous or manual control. The addition of the circuit breaker provides a way to shut off all power from the source in the event of an absolute emergency, such as an electrical fire.

In addition to the hardware E-stop, the power board has a software E-stop for the motors as a secondary safety option. Whereas the hardware E-stop kills the power to the motors, the software E-stop sends a zero-speed command to the motors, which allows for a shortened restart time after removal of the stopped state. The RC controller emergency stop has a range of 0.25 miles. Should RAIN go beyond that range and lose connection with the controller, it is automatically stopped.

SOFTWARE STRATEGY AND MAPPING TECHNIQUES

Overview

The intelligent navigation software that operates RAIN will be loaded to the onboard NVIDIA Jetson TX2 prior to deployment. The software provides feedback to verify that all systems are operational and optimized for navigation. Once all systems are online, the user will simply press “go”, and the vehicle will begin autonomous operation.

Obstacle Detection & Avoidance

Different detection and avoidance algorithms have been in development for RAIN. Since the field of the competition has moved from a grassy terrain to a parking lot, much of the software had to be created late in the timeline to adjust for the drastic change. Historically, the LiDAR has been responsible for obstacle detection and avoidance, while the camera has generally been used for lane detection. The change to pavement opened the possibility of using the LiDAR to detect the lanes from the pavement by using the change in intensity value. Using the LiDAR for lane detection instead of the camera would ease the processing required of the Jetson TX2. It is believed that the program would also be more stable, since specific thresholding of the vision will no longer be required. With LiDAR, lane detection is still in the early stages of development, so the camera still utilizes the following algorithm. The algorithm combines sensor inputs to generate the best path forward. The general approach for the autonomous challenge is detailed below in Figure 6. The process cycles continuously during lane following, adjusting the vehicle heading and speed to effectively navigate the course until it is commanded to stop.

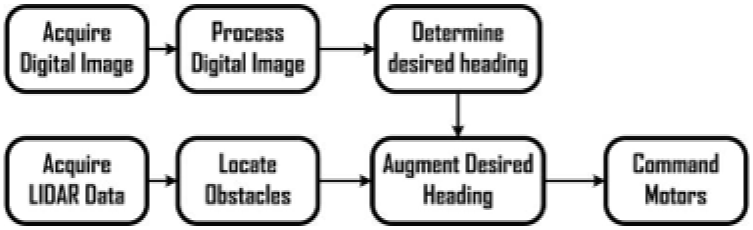


Figure 6:AutoNav Lane Following Challenge Algorithm

The image processing methodology is detailed in Figure 7. The acquired image is pre-processed by extracting specific image values from the composite RGB color image. To reduce processing time, the image is down sampled from the 3849x1929 native camera resolution to 160x120 pixels and converted to grayscale. The area of the image including the vehicle is then removed and the image is then split in two, representing the view to the left and right of the vehicle. To determine the strongest linear relationship in the images, a Hough transform is used and the dominant line occurring in each half of the image is identified. This method works equally well for solid or dashed lines. A decision tree is implemented to determine a vehicle heading based upon situational line detection cases. The obstacle avoidance capability subsumes the vision derived heading. The final vehicle heading, being the composite of the vision and obstacle avoidance data, is then used to command the motors.

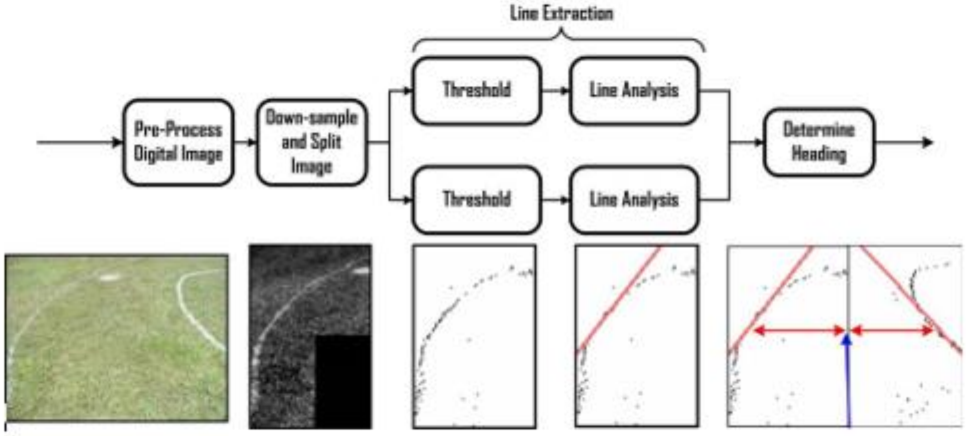


Figure 7:Image Processing Algorithm for Boundary Line Extraction

The final commanded path is represented as a curve yielding a fluid motion as the vehicle navigates the course. The use of curved path motor commands accounts for vehicle dynamics. While either a curved or straight path command assumes that the vehicle can instantaneously change velocity, the curved path accounts for the dynamic response of the commanded wheel speeds and more accurately renders the actual motion of the differentially steered vehicle.

Software Strategy & Path Planning

The path planning algorithm will be based on a priority system. The waypoint navigation is the lowest priority, #3, but will act as the initial and fallback to inform the system of the relocation goal that RAIN is trying to navigate to. The #2 priority is vision, which will be responsible for lane detection if the LiDAR is unable to detect lanes. It will also detect the ramp and potholes in the path. The highest priority, #1, will be the LiDAR since it will be responsible for both lane and obstacle detection. Once the vehicle detects that it has entered the GPS portion of the course, it switches to a GPS-based navigation algorithm using sensory input from the differential GPS, digital compass, and LiDAR and is described in Figure 8.

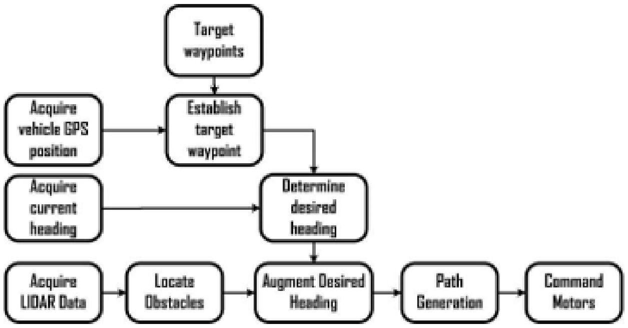


Figure 8:Overview of the GPS-based Navigation Algorithm

Goal Selection & Path Generation

The trajectory planner is responsible for generating an appropriate heading for RAIN to follow based on the information fed to it from the parallel running nodes. It is made up of two parts: the desired heading and the desired goal. The heading is selected based on available data from the software subsystem hierarchy. The desired path uses a reverse priority order: waypoint navigation, lane heading, and then obstacle avoidance. The goal heading is the next desired waypoint. The desired heading is the desired outcome from the algorithm.

Path generation is derived from the known position and orientation of the vehicle and the next target position which the vehicle is to achieve. A polynomial curve fit is used to plan a suitable path between waypoints. The LiDAR subsumes the GPS based heading and behaviorally conditions the vehicle’s desired heading until the obstacles are cleared and the vehicle can approach the target waypoint unhindered. When RAIN reaches the last target waypoint, it returns to the initial starting position and ceases motion, indicating that the mission has been accomplished.

Map Generation

RAIN does not rely on previously generated maps, but instead works on a reactive system in the same manner as an autonomous car. It has a local state of information which is constantly updated, plus the goal heading which it ultimately aims for. This was chosen since the real world does not stay constant. Examples of inconsistencies would be road construction, pedestrians, and other vehicles. RAIN counts on this being reflected in the IGVC competition since the judges are likely to change the course between runs. Still, this data does not go to waste as it is logged and can be played back to understand how RAIN reacted to the environment in a MATLAB simulator.

FAILURES & LEARNING POINTS

Failure Modes and Resolutions

Although RAIN was designed for maximum rigidity and modularity, failures are always possible. To combat this possibility, several different failure modes have been identified, and plans have been put in place in order to minimize the impacts of failure. A major struggle the team faced was the learning curve behind ROS. The team had no recurring members from previous years and lacked proficient programmers for either LabVIEW and/or ROS. Therefore, software development was started from scratch. With that, the team needed to learn the basics of the programming languages used. Since ROS is an open source and well documented program, there are pre-built sensor nodes and options available for the team to use and reference.

Vehicle Failure points

Mechanically, the vehicle continues to use a few hardware components and sensors from previous years. These components were thoroughly tested to ensure they keep their integrity. The vehicle's design and construction were focused on the concept of modularity. Meaning, in the event of a failure, high risk components are easily removable and replaceable. Attached to the vehicle is an onboard toolkit which contains the tools necessary for replacing any physical component at any time. Because of the changes in components, some connectors have been changed and others removed, leaving small openings in the pelican case, potentially losing the IP65 rating the previous entries have had. Using knowledge gained and from these failures, there are plans to further waterproof the vehicle.

SIMULATIONS

The data logging system helps immensely with the testing and refining process by identifying problems that cannot be immediately noticed by vehicle performance inspection during a test. The output of the program is shown in Figure 9. The black rectangle represents the robot, while the blue represents obstacles, and the green circle is the target waypoint. The green semicircle extending from the vehicle is the obstacle avoidance range, which will cause a reaction from the robot. The red dots show the vehicle's GPS trail. The left-hand side shows numerical values that can be customized to meet the user's desired values, including elapsed time, wheel speeds, and latency. Data can be fed into the system simulator from the data simulator to observe the output of the system, which is useful in determining software issues without going outside for a test run after each change.

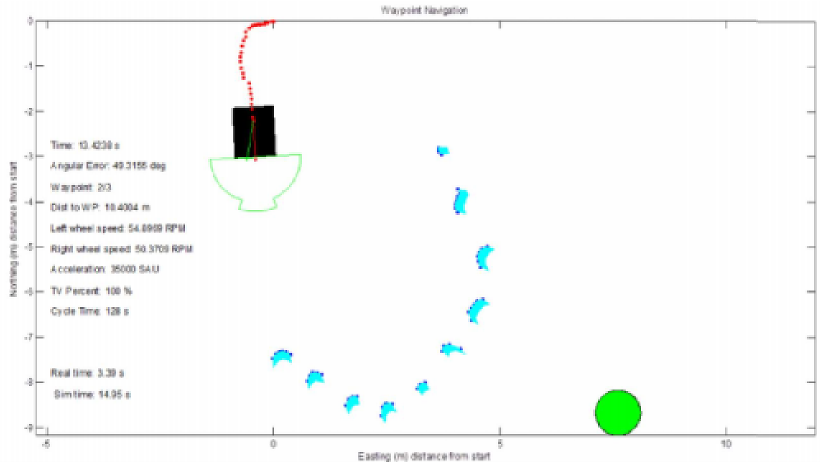


Figure 9: MATLAB System Simulator

Progress for simulations using Gazebo in ROS is in development for the 2021 entry. The LiDAR and camera information can be collected with the SD card or internal hard drive for troubleshooting post-run data.

PERFORMANCE TESTING

The system has been tested and proven for reliable operation of hardware and electrical subsystems in teleoperated control. RAIN can maintain its speed between 1-5 MPH and will easily carry the required payload up a ramp of over 10 degrees without difficulty. The autonomous path planning and navigation is still being developed for functionality but is planned to qualify for a run before competition. The subsystems have all been individually tested in a lab environment and are successfully able to gather their respective data.

INITIAL PERFORMANCE ASSESSMENT

At the time of this writing, RAIN is in good working order in terms of its power supply and hardware and is mechanically sound. Manual (remote) operation is reliable and shows great potential for a high performing robot. The functionality in terms of software is still being developed and has yet to complete even the most basic tasks of autonomously moving to a GPS waypoint, following a line, or avoiding any obstacles. Moving forward, attention will be steered solely toward building a functioning program for RAIN to accomplish the tasks outlined in the IGVC rules and meet the customer needs determined through the Design Process of this project.

CONCLUSION

RAIN is designed to be a fully autonomous robotic vehicle developed, manufactured, and tested by mechanical engineering students at ERAU-DB. The team gave special attention to the requirements outlined by both the IGVC competition and those set forth by their professors and peers to accomplish their goal of creating a lightweight, safe, maintainable, and accessible robot with a modular design. While RAIN requires some fine tuning, it is planned to be ready for the 28th IGVC competition this summer. The team feels confident that RAIN will not only be capable of completing the basic and advanced practical courses, but that it will also impress the judges with the innovative changes made throughout the year.

References

[1] IGVC Rules committee, “IGVC Rules 2020,” http://www.igvc.org/2020IGVCRules.pdf

[2] Middlebrooks, N., and others, “Embry-Riddle Aeronautical University – DAISI,” http://www.igvc.org/design/2018/4.pdf. Annual Report. Embry-Riddle Aeronautical University, 2018. Web.

[3] Middlebrooks, N., and others, “Embry-Riddle Aeronautical University – DAISI-C,” http://www.igvc.org/design/2019/4.pdf. Annual Report. Embry-Riddle Aeronautical University, 2019. Web.

[4] Schoener, M., Middlebrooks, N., and others, “Embry-Riddle Aeronautical University – DOLLE,” http://www.igvc.org/design/2017/3.pdf. Annual Report. Embry-Riddle Aeronautical University, 2017. Web.