Bob Jones University Robotics Team
Bruin 3

Date Submitted: May 14, 2019

Team Captain: RJ Ring | rring214@students.bju.edu

Fall 2018 Team Members:

Daniel Clauser | dclau415@students.bju.edu
Nathan Collins | ncoll943@students.bju.edu
Elizabeth Franklin | lfran167@students.bju.edu
Joshua Grimm | jgrim876@students.bju.edu
Lemuel Jacobson | ljaco134@students.bju.edu
Jacob Koechig | jkoec341@students.bju.edu
Ruth May | rmay657@students.bju.edu
Jared Mundy | jmund674@students.bju.edu
Bradley Pauley | bpaul675@students.bju.edu
Lydia Petersen | lpete480@students.bju.edu
Ezra Pio | epio175@students.bju.edu
Natalie Reed | nreed914@students.bju.edu
Sinjin Seiber | sseib267@students.bju.edu
Carter Shean | cshean@bju.edu
Jeremy Tan | jtan716@students.bju.edu
Steven Vanphavong | svanp649@students.bju.edu
Kyle Weberg | kwebe254@students.bju.edu
Nathanael Winslow | nwins689@students.bju.edu

Spring & Summer 2019 Team Members:

Lane Camfield | ccamf948@students.bju.edu
Alex Raddatz | aradd575@students.bju.edu
Marcela Martinez | mmart372@students.bju.edu
RJ Ring | rring214@students.bju.edu

Faculty Advisors:

Bill Lovegrove | blovegro@bju.edu
Will Woodham | wwoodham@bju.edu

Statement of Integrity:

I certify that the design and engineering of the vehicle by the current student team has been significant and equivalent to what might be awarded credit in a senior design course.

Faculty Advisor: William K. Woodham, M.S.P.D.
Assistant Professor
Department of Engineering
Bob Jones University

Faculty Advisor Signature: _______________ Date: 5-14-2019

INTRODUCTION

As part of the engineering Mechatronics course, eighteen students from Bob Jones University implemented and integrated several subsystems to transform a Polaris GEM e2 vehicle into an autonomous vehicle named Bruin 3. The vehicle was designed with an E-stop system and the following capabilities: lane following, navigation, computer controlled steering, acceleration, and braking. This report details Bruin 3's design process.

ORGANIZATION

In the Fall 2018 semester, six student teams organized by subsystems contributed to the overall vehicle. Table 1 lists members of each team and the number of hours contributed to the project for a total of 1,219 hours for the Fall semester. This does not include subsequent student hours contributed in the Spring and Summer 2019 terms.

Table 1. Team Members and hours worked.

Team	Name	Hours
Mechanical Integration Team	Jacob Koechig	55
	Nathanael Winslow	70
	Steven Vanphavong	79
Electrical Integration Team	Lydia Petersen	45
	Ezra Pio	70
	Ruth May	52
Steering Integration Team	Daniel Clauser	73
	Bradley Pauley	74
Braking and E-Stop Integration Team	Sinjin Seiber	63
	Jeremy Tan	52
Sensor Integration Team	Josh Grimm	63
	Lemuel Jacobson	70
	Jared Mundy	56
	Natalie Reed	72
	Lauren Elizabeth Franklin	38
	Kyle Weberg	70
Software Integration Team	Nathan Collins	117
	Carter Shean	100

DESIGN ASSUMPTIONS AND DESIGN PROCESSES

Bruin 3's design was derived from the design of the former autonomous vehicle, Bruin 2. Although design elements were copied, Bruin-3 is a completely new vehicle. Updates and modifications to the design included implementing a new method of acceleration, updated and new operating program, a new vehicle, and integration of several new sensors.

The team followed a seven-step design process shown in Figure 1.

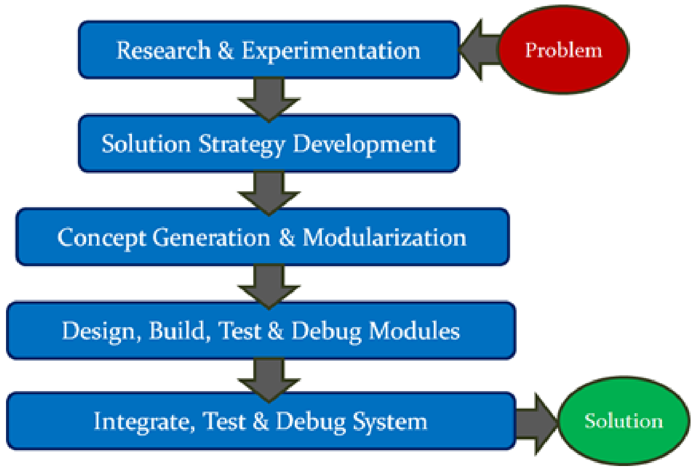

Figure 1. Design Process.

The first step taken in the design process was researching and experimenting with the various sensors that were to be integrated on the vehicle. Researching previous work on similar projects was also conducted during this initial stage.

The second step was developing a solution strategy. Each team brainstormed possible solutions and methods for accomplishing each goal.

The concept was generated and modularized by creating a system block diagram and electrical schematic outlining the hardware and wiring. Additionally, the vehicle with all systems was assembled in SolidWorks.

Upon modeling each system, sensors and actuators were built, tested, and debugged to ensure compatibility with ROS before installation on the vehicle. This seven-step process ensured that each subsystem performed at optimal efficiency and without error.

INNOVATIONS

RTK

The RTK (Robot Technology Kernel) software was provided by United States Army CCDC Ground Vehicle Systems Center (formerly TARDEC) and integrated into our vehicle, as part of a grant to help develop RTK. RTK works alongside the software designed by the team members, but the difference between them is that RTK is the brain that tells the vehicle what to do. According to the information or input that comes in, RTK decides what the next action should be and then the system designed by the team sends signals to the various parts that control the vehicle.

Independent hydraulic braking system

A Hydrastar hydraulic pump activates the rear hydraulic brakes, both for normal autonomous braking and for estop. An independent hydraulic system operates the front brakes via the manual brake pedal. A backup battery provides power to the Hydrastar to ensure estop in case of a system power loss.

Triple Cameras

There are three different cameras that will be used in the Bruin 3 project.

1. Stereo Camera: will be used for object detection in front of the vehicle
2. Lane Detection Camera: detecting lines on the road
3. Road Sign Detection Camera: detecting stop signs

Optical Flow Odometer

A PX4 Flow Optical Odometer was included to provide feedback on the speed of the vehicle. Mounted between the two rear wheels, the odometer determines the speed of the vehicle by capturing images of the ground, rather than with a rotary encoder on the wheels as is traditional.

MECHANICAL DESIGN

Overview

The rear of the vehicle is divided into two sections, the bottom rear and the top rear. The bottom rear of the vehicle contains the Hydrastar, the PX4-Flow optical odometer, and the back-up battery. A metal plate sits on top of the back-chassis frame and supports the three pieces of equipment. The top rear section contains the two CPUs (Fanny and Freddy), a 16-port ethernet box, two power converters, the IMU, and the Lidar ethernet box.

The following sensors are located on the top of the vehicle: LIDAR, Mako camera, GPS unit, and a light beacon. These are attached on top of a metal plate that is secured with brackets via the T-slot feature on the sides of the car.

The front of the vehicle contains the RADAR and the stereo camera. The RADAR is mounted to the front, diagonal beams of the frame. The stereo camera is attached to the vehicle right below the windshield in the charging port area.

Drive-by-wire kit

The team designed and installed their own drive-by wire-kit allowing the computer to control the vehicle's steering, brakes, and acceleration pedal.

The system consists of three actuators. The brake actuator is a pump called the Hydrastar. This pump is in charge of pushing hydraulic fluid into the brakes. Next there is an electric motor designed to turn the steering wheel. Finally, an electronic accelerator pedal interface was created by the team that produces the same signals as the accelerator pedal from the factory car.

Suspension

The 2018 GEM e2's front suspension is a MacPherson strut and the rear suspension is an independent trailing arm. No changes were made to the vehicle's stock suspension.

Weather proofing

Bruin 3 must be protected from the weather. Concerning the sensors, the three cameras, LIDAR, and Radar, GPS unit, and IMU are unprotected. The enclosed sensors are the DAC, PX4Flow odometer, and the steering equipment. In the back, a water proof box protects the vehicle's batteries and various electronics. Soft doors and with a back window were purchased to protect the rest of the vehicle.

DESCRIPTION OF ELECTRONIC AND POWER DESIGN

Overview

The vehicle uses a 48V Battery Pack as power source and various converters that power the computers and sensors. Additionally, a 12V battery pack serves as a backup battery for the HydraStar braking system.

The vehicle consists of three main computers and sensors. The sensors include cameras for obstacle detection and localization sensors.

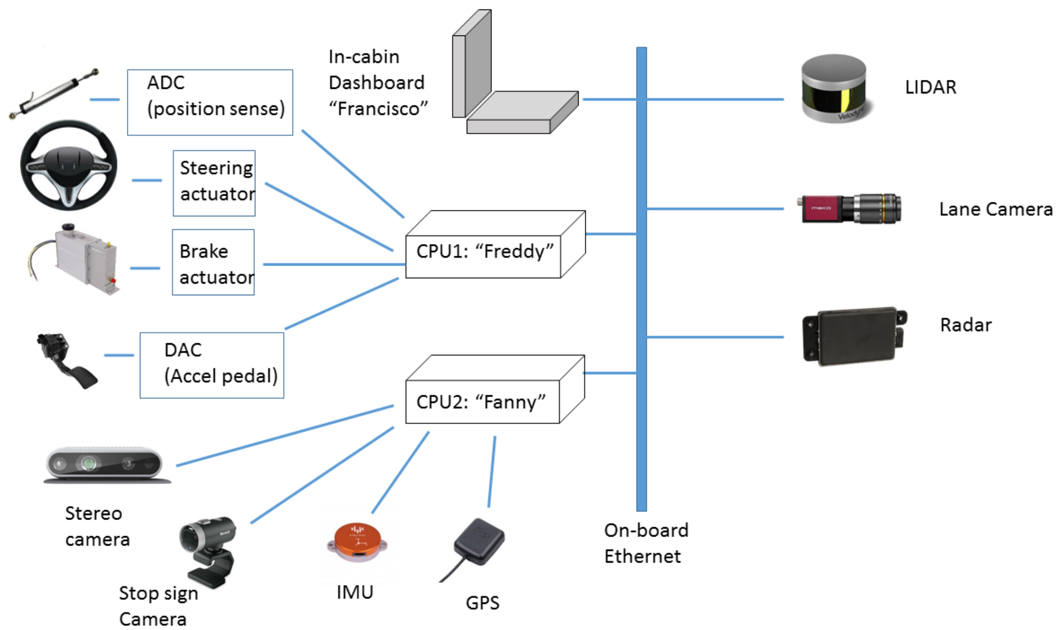

Figure 2. Block Diagram of the Control System.

Power Distribution System

There are two 48-to-12V DC-to-DC converters from which the computers are powered. The computer that processes images, “Fanny,” requires a 12V power supply and draws 10A. All the cameras are connected to this computer. The sensors require 12V power supply and draw 3A. The other computer, “Freddy,” receives feedback from the other sensors and sends signals to the motor controller and actuators which requires a 12V power supply and draws 5A as seen in Figure 3.

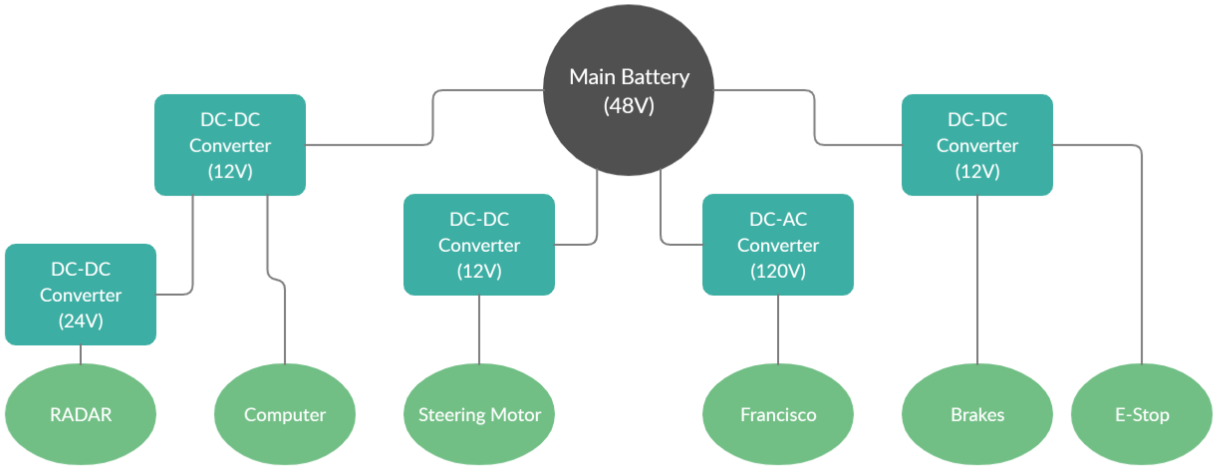

Figure. 3 Power Distribution Diagram.

Electronics Suite Description

Computer Hardware

1. *Freddy* is a LINUX PC that runs the actuation nodes.
2. *Francisco* is a LINUX laptop that runs the high-level RTK nodes and provides a software dashboard in the cab of the vehicle.
3. *Fanny* is a LINUX PC that runs sensors and localization nodes.
4. *Whyme* is a Windows laptop that runs the WMI (Warfighter Machine Interface) and can be used by the vehicle's occupants or a remote operator.

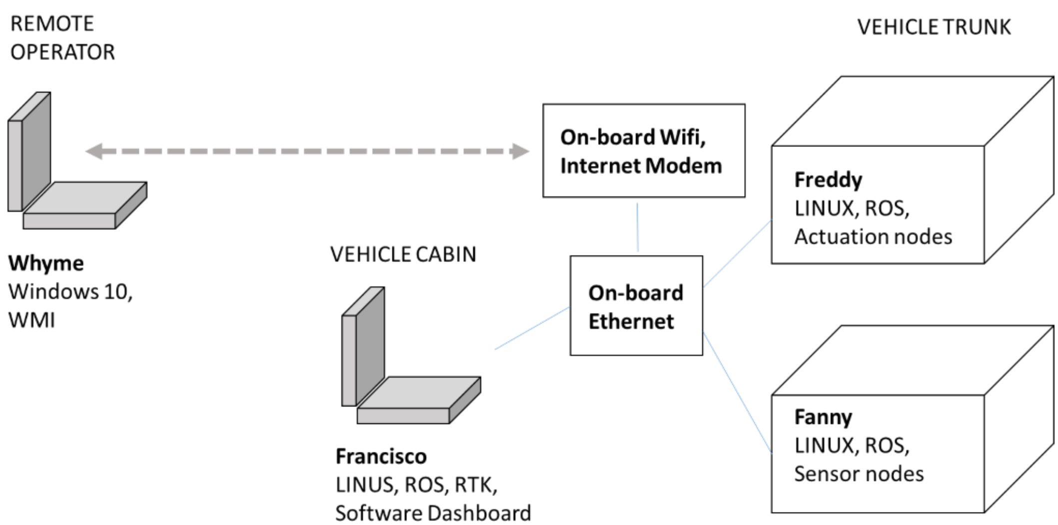

Figure 4. Computer Hardware Connections.

Sensors
1. Cameras
    a. Lane detection camera
    b. Stereo camera
    c. Road sign detection camera
2. LIDAR
3. RADAR
4. Localization sensors
    a. GPS sensor
    b. Odometry sensor
    c. IMU (Inertial Measurement Unit)

Figure 5. Sensor locations in the vehicle.

Safety Devices

Fuses are inserted between the DC-to-DC converters and computers to prevent short circuits. An E-Stop system is also implemented in the case that the vehicle must be shut down immediately. A backup battery is installed in the HydraStar's braking system in the case of failure.

SOFTWARE STRATEGY AND MAPPING TECHNIQUES

Overview

Our software is based on the ROS (Robot Operating System) framework which is an open-source software package maintained by the Open Source Robotics Foundation (OSRF). In ROS each major function is managed by a separate smaller program called a node. The nodes communicate between each other and perform the various tasks needed for our vehicle to function. Then we have RTK which is a compilation of ROS nodes managed by GVSC, the Army CCDC Ground Vehicle Systems Center. A list of custom ROS nodes written by the team can be found on Table 2.

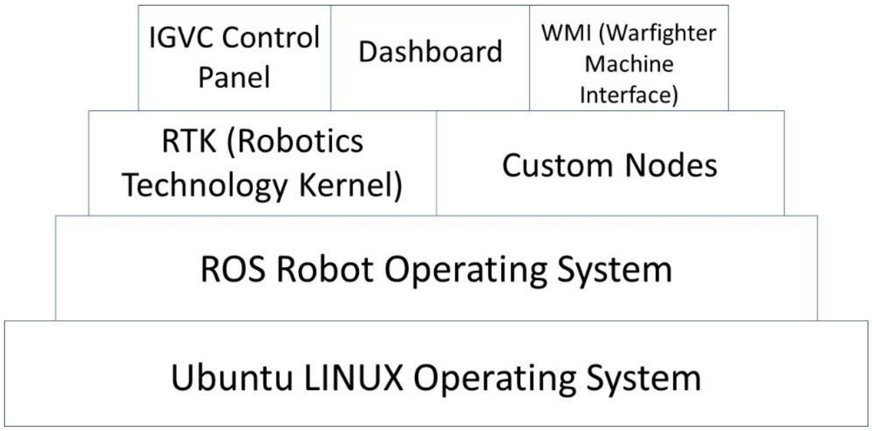

Figure 6. Software Architecture.

Table 2. Custom ROS nodes.

Nodes	Description
steer_drive (Freddy)	Subscribes to /navigation/curvature_setpoint and /navigation/speed_setpoint (RTK) Publishes /steering/position Reads and publishes the steering position sensor. Controls the DAC that creates the pseudo-accelerator-pedal signals.
relay_board (Freddy)	Subscribes to /navigation/speed_setpoint (brakes on speed zero) Controls the hydraulic brake actuator and the warning light on top of the vehicle.
waypoint_drive (Fanny)	Publishes /navigation/speed_setpoint, /steering/steer and /brake Simple waypoint following, alternative to RTK for testing.

igvc_tasks (Francisco)	Human Interface to select IGVC tasks Publishes to RTK to execute various IGVC tasks
dashboard (Francisco)	Publishes /steering/steer and /navigation/speed_setpoint Subscribes to and displays /imu, /gps Allows manual control of all actuators and displays sensor values for testing and debugging
radar_obstacles (Fanny)	Reads obstacles from the RADAR device and publishes them to RTK's world model.
lane_detector (Francisco)	Reads images from the camera on top of the vehicle. Detects lane lines in the image using OpenCV. Publishes obstacle information to RTK's world model to keep the vehicle within the lanes.
stop_sign_detector (Francisco)	Detects stop signs using OpenCV / neural nets Publishes messages to stop the vehicle for the stop signs /opt/ros/kinetic/shared/lib
Localization_transform (Freddy)	Takes the data output by the navsat_transform_node and converts them to a format useable by RTK.

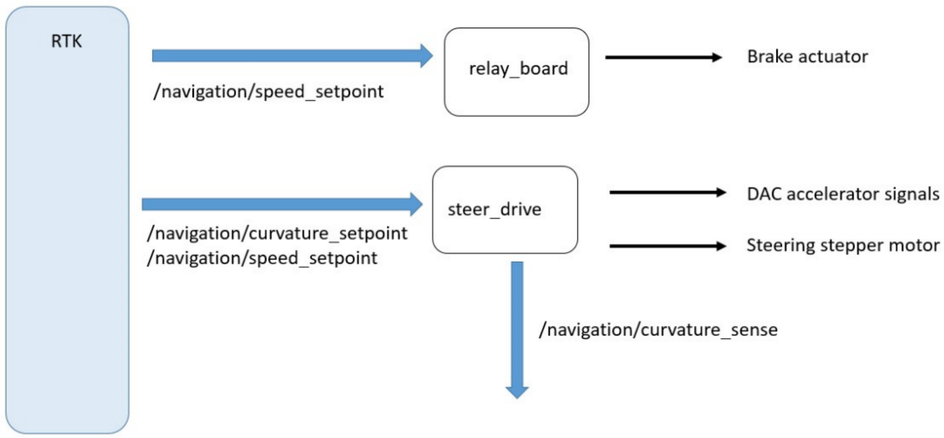

Figure 7. Custom actuator nodes interaction with RTK.

Obstacle Detection and Avoidance

We used three major components for obstacle detection: a stereo camera, LIDAR, and RADAR.

1. LIDAR is used to create a 3D map of the area around the vehicle as seen in Figure 8.
2. RADAR senses certain obstacles in front of the vehicle, for example, pedestrians and other vehicles.
3. The stereo camera detects obstacles that are in front of the vehicle as seen in Figure 9.

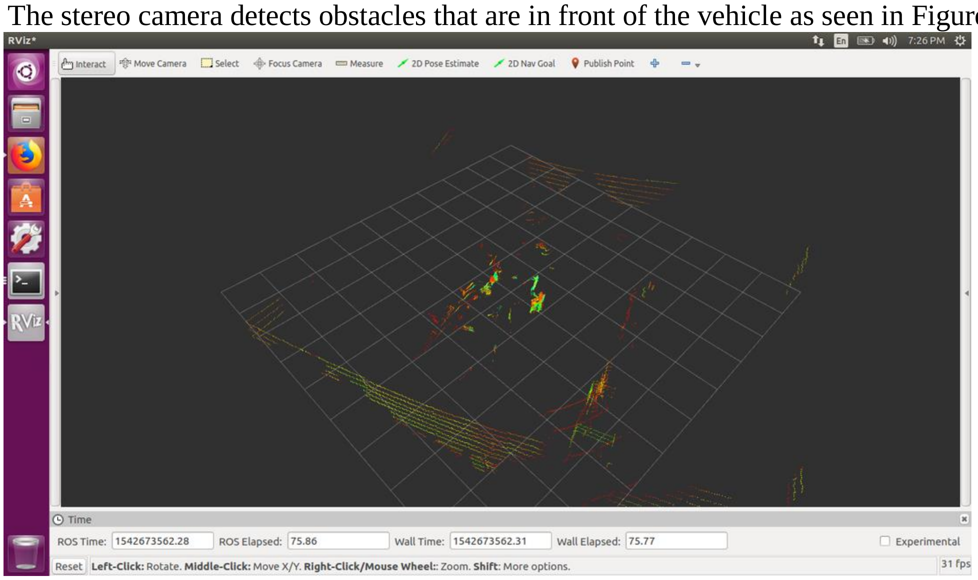

Figure 8. LIDAR sample.

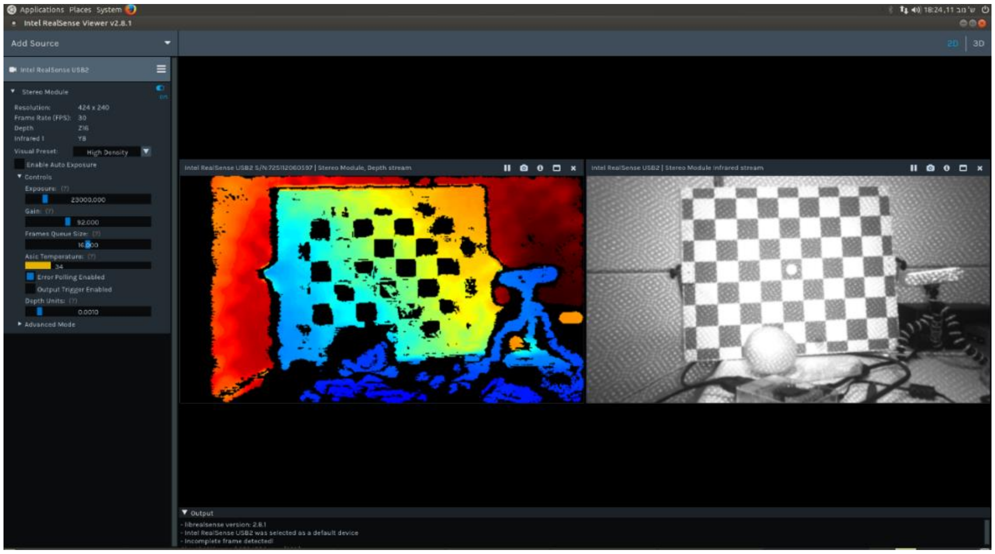

Figure 9. Sample depth image from the RealSense stereo camera.

Lane Following

A camera on the top of the vehicle scans for road marking lines. These marking lines are turned into 3D obstacles and added to the cost map so that the path planning will stay between them.

Map Generation

The world model module of RTK combines data from the LIDAR, stereo camera, and other sensors to generate a map of the world around the vehicle with the obstacles and other parts of the course. This world model then generates the costmap for the entire situation. The generated costmap indicates the riskiness of different paths the vehicle can take. See example cost map in Figure 10.

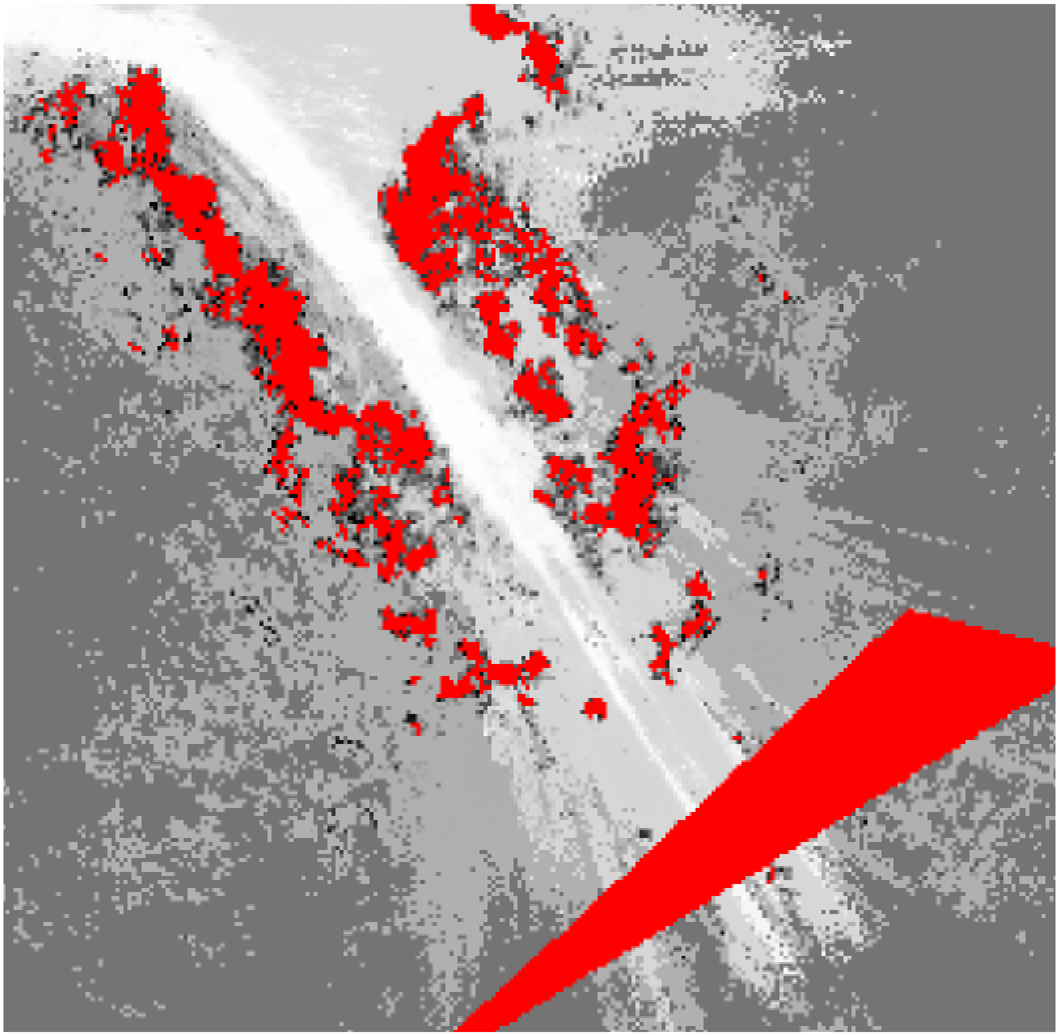

Figure 10. Example of a costmap generated by RTK.

Path Generation

The path planning module of RTK uses the costmap and the A* algorithm to find the path of least total cost. The RTK module uses the sensor information to determine where the vehicle cannot go, like going off the road or crashing, and associates that with a very high cost. The clear road ahead of the vehicle will be assigned a very low cost and that is the path that the vehicle will take. The vehicle will then use this path that RTK plans to drive the motor and steering of Bruin 3. There will be a user interface where the user can provide their desired destination. The vehicle requires GPS to perform waypoint navigation. When the next point is entered or identified, the system then uses the sensors to ensure all obstacles are avoided. The vehicle stays in the lane while the GPS indicates the location to direct the system to head in the right direction.

DESCRIPTION OF FAILURE MODES, FAILURE POINTS AND RESOLUTIONS

Vehicle failure modes and resolutions

If the steering actuator subsystem fails, the vehicle may attempt to drive into obstacles. The obstacle sensors should detect the obstacles. It will attempt to steer around the obstacles, without success, and then stop the vehicle when it becomes clear that a viable path is no longer available. The human safety driver is also responsible to observe the operation of the vehicle and intervene if the path is toward an obstacle.

If the brake actuator subsystem fails, we are dependent on the human safety driver to stop the vehicle using the independent front wheel braking system. In the unlikely event of failure of both brake systems, the parking brake may be used by the human safety driver to stop the vehicle.

If the accelerator pedal actuator fails, the vehicle may accelerate out of control. The obstacle detection systems should intervene and attempt to brake the vehicle. If full acceleration and braking are both actuated at the same time, the brakes will be able to stop the vehicle but at a reduced rate. The human safety driver may need to intervene in this situation as well.

Vehicle failure points and resolutions

If the battery fails, we are dependent on the back-up battery. This battery will continue powering the vehicle estop which will be triggered by loss of power.

If the actuators fail, we are dependent on the e-stop to stop the vehicle, in order to avoid further complications.

If the e-stop communication fails, this will cause an e-stop.

If the communication between the computers fails, the vehicle will stop driving.

All failure prevention strategy

The vehicle health system can detect multiple failure points across the vehicle, and it will stop driving.

The vehicle operation currently requires a human safety driver in the vehicle at all times. The human operator can engage the estop at any time. The operator can also brake or steer the vehicle manually.

Testing

The estop system was tested and the stopping distance was measured at 11 feet at 5 miles per hour. It also engaged when a wire was disconnected from the e-stop system.

The wireless estop system has a specified range of 600 feet, well beyond the IGVC requirement of 100 feet. This range has yet to be tested but will be tested before IGVC.

The torque required to override the steering actuator was measured to be 5 lbs. at 5.5 inches or 2.3 foot-pounds (3.1 Nm) of torque, which is easily achievable by the human safety driver.

The manual brakes were tested and are fully functional in autonomous mode.

The vehicle and our on-campus test track are modeled in the Gazebo simulation environment. The vehicle can be driven in the simulated environment using the same software as the real vehicle. The resulting vehicle trajectories can be compared. See section 8 for simulation details.

The actual vehicle was tested on a grassy field on the BJU campus. On March 12 we successfully demonstrated driving to a waypoint as seen in Figure 11. See section 9 for testing details.

Figure 11. Picture taken on the testing day for the vehicle.

Vehicle safety design concepts

The Polaris E2 vehicle meets all the safety standards for a low speed electric vehicle (LSEV) including headlamps, tail lamps, stop lamps, reflectors, mirrors, a parking brake, a windshield and seat belts. We have not modified any of the safety features except the brakes as described below.

The speed and path curvature are limited by the software to stay within the limits of the vehicle.

The vehicle includes a fire extinguisher as required by the IGVC rules.

Four on-board e-stop buttons and a wireless e-stop provide hardware shutdown of all of the actuators. The estop buttons activate normally-closed switches, so any hardware fault in the system that results in an open circuit causes an estop.

The steering wheel and brake pedal are fully functional in autonomous mode, giving a safety driver capability to control the vehicle manually at all times. With the current vehicle we intend to operate the vehicle only with a human safety driver in the driver's seat.

The HydraStar braking system uses a backup battery to provide positive braking in the event of an e-stop; the vehicle does not coast after e-stop and will stop even in the case of a total loss of primary system power.

The front (manual) and rear (estop) brakes have separate hydraulic systems, so that if either system fails the vehicle can be stopped with the other.

SIMULATIONS EMPLOYED

Simulations in virtual environment

In order to test our software without taking the vehicle out in the field, we are using a simulation tool called Gazebo like the one in Figure 12. Gazebo is packaged with ROS, and it provides a graphical simulation of the vehicle along with a simulation of the sensors and actuators from the real vehicle. For this project, we are using a modified Gazebo simulation of all nine tests used at IGVC. This simulation includes a vehicle similar to ours along with a number of sensors similar to ours.

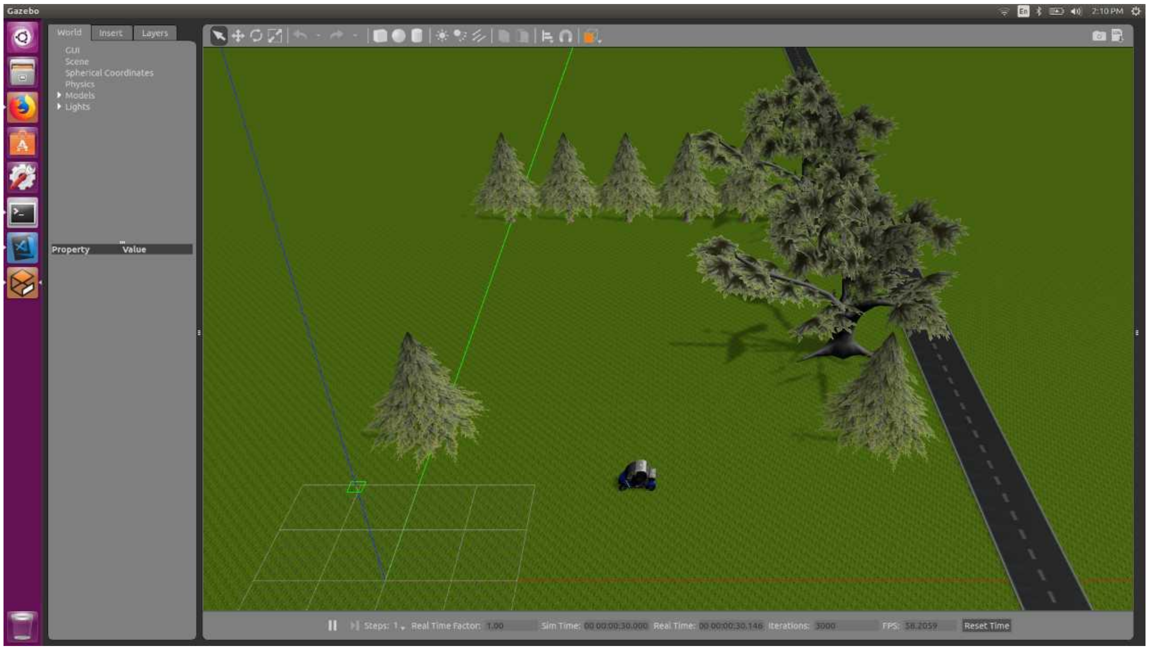

Figure 12. Gazebo simulation example.

Theoretical concepts in simulations

Gazebo

Gazebo runs off a *world file* that describes the vehicle, sensors, trees, and other items in the world being simulated. It uses a system of *plugins* for simulated input and output.

We have two world files, for the test track we have on campus and for the IGVC competition.

PERFORMANCE TESTING

Component testing, system and subsystem testing, etc.

The vehicle has a specified battery range of 20 to 30 miles. In a worst-case battery life test (hilly terrain, high speed stop-and-go driving) the vehicle reached a “low battery” level after 12 miles. This lowers the range between 8 to 18 miles but is more than adequate for IGVC.

The vehicle is able to climb a 6-degree (11%) slope easily.

All sensors have been successfully tested in ROS.

The Dashboard program provides feedback on sensor status to assist in testing and debugging.

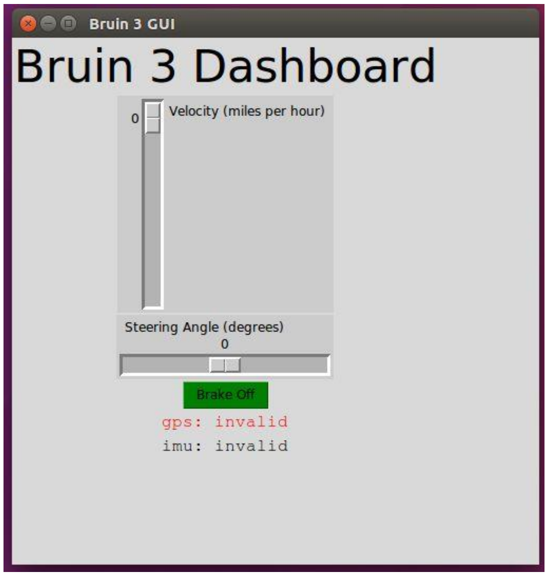

Figure 13. Dashboard sample.

INITIAL PERFORMANCE ASSESMENTS

At the time of completion of this report, the vehicle is capable of simple drive-to-waypoint behavior but is not yet capable of performing the full set of IGVC tasks.