The City College Autonomous Vehicle Team

Proudly Presents

BeaverBot

George Giamos • Luis Pazmino • Vibol Chav • George Ragousis • Wai Khoo • Ruddy Ostin • Ruben Martinez • Roberto Talavera • Pavel Danilotchkine • Rhonda Ramlogan-Burke • Christina Valdez • Safwan Tanveer • Andrew Assing • Martin Holly • Alexis Tejada

Faculty Advisor Statement:

I certify that the engineering design on the BeaverBot by the current student team has been significant and was awarded credit in a senior design course.

Jizhong Xiao
Department of Electrical Engineering
The City College of New York

1. Introduction

The Autonomous Vehicle Design Team at The City College of New York (CCNY) is proud to present BeaverBot, an autonomous outdoor vehicle exclusively designed to compete in the 15th Annual Intelligent Ground Vehicle Competition (IGVC). BeaverBot represents the joint effort of senior-level students in electrical and mechanical engineering, who aspire to make this competition a tradition for future students interested in the field of robotics at CCNY. BeaverBot inherits its name from the school mascot, which embodies the qualities of the engineering profession. BeaverBot is a three-wheeled differential drive vehicle developed with Commercial off-the-shelf (COTS) hardware and software developed on C++, which reduce the computing time significantly. BeaverBot features a reliable power distribution system, an inexpensive custom-made wireless e-stop, and reactive artificial intelligence algorithms. The team believes that the design and manufacturing of BeaverBot will provide a robust, safe, and durable platform to perform well in the IGVC.

2. Design Process

The design process used by CCNY's Autonomous Vehicle Design Team is shown in Figure 2.1. The process is mainly focused on producing a deliverable which will meet the requirements of the competition, on time and within the assigned budget. In order to ensure quality, feasibility and reliability, the project was subject to reassessment in every stage. The process followed depends strongly on research and testing of the ideas proposed by the members. The approach was our best solution given the inexperience of all the members in the field, and the time given to complete the task. The stages of the project are explained in detail in the following paragraphs.

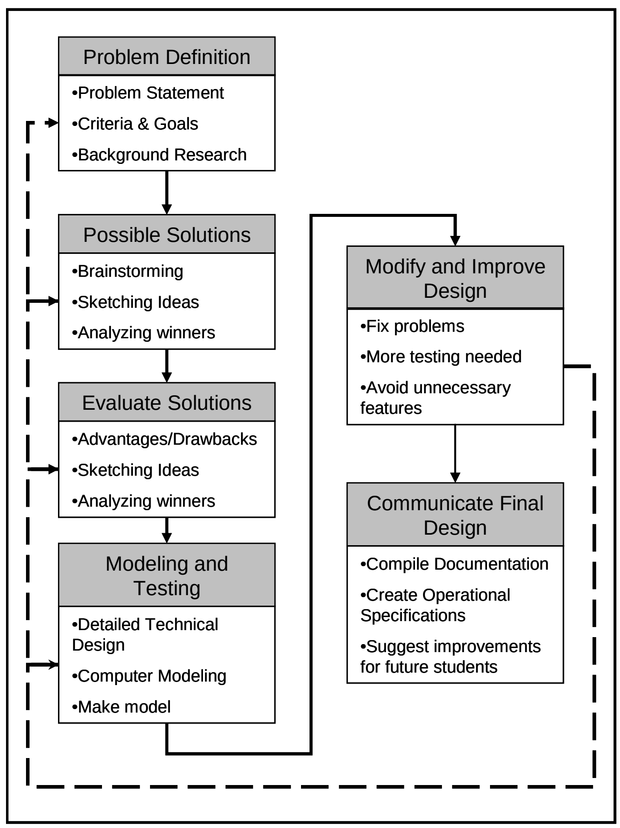

Figure 2.1 – Design Process Used

2.1 Problem Definition

The design process started by defining a clear goal and making sure all the members of the team understood the concept as a whole and their role in the development of the product. The problem statement was clearly denoted in the IGVC rules which were carefully analyzed by each team member. The first step afterwards was to define our priorities to this project given our resources. It was established that it is primordial to be able to compete and gain experience for future generations. The team also wanted to create a durable robot that could be used in future years, with a very reliable electro-mechanical system. Having this in mind, the team researched successful designs from previous years, and also new alternative designs that could provide reliable motion.

2.2 Possible Solutions

The team conducted several brainstorming sessions where everybody contributed with their expertise to the selection of a possible robot structure. The main points of interest were (i) a robot that could rotate on a zero-radius, (ii) the use of multiple sensors to avoid relying heavily on a single sensor, (iii) the size of the robot should be as small as possible for efficient maneuverability, and (iv) it was important to fundraise and administer the monetary resources to efficiently be able to acquire the necessary equipment and devices. Lastly, the designs of winning teams from previous years were studied in detail.

2.3 Evaluation of Solutions

The evaluation of solutions took place after our budget was complete and we had enough cash flow to make decisions regarding the design. Each possible solution was weighted with advantages and disadvantages to establish the most suitable approach. Among our evaluation was the use of custom-made electronics or commercially available ones. By a unanimous consent, the team decided to purchase COTS to avoid delays if the hardware was to fail. The team also tried its best to ensure that these components can be interfaced with one another. With the guidance of faculty and some robotics hobbyists, we were able to narrow down our selection to widely recognized peripherals, which had technical support in case of failure or incompatibility. The decision to pursue bigger companies was based also in the probability that some will provide sponsorship to our team.

2.4 Modeling and testing

The modeling stage included computer aided design and simulations of mechanical components using MSC.Nastran® and electrical components using Cadence® OrCAD®. The models provided the team with tools necessary to verify the theory of design, as well as to provide insight on the limitations of the vehicle. The simulators helped improve the design by allocating the mechanical components in efficient locations, as well as limiting noise from electronic components. Once the simulation parameters fitted the specifications required for the competition, the frame and circuitry were assembled.

2.5 Modify and Improve Design

The next step was to improve the physical model in order to fit our parameters if there was any deviation. The team was able to do so by calibrating all the components using approximated polynomials functions from data containing the desired results vs. the actual results. In most cases, the percent error was a steady repetitive function. For example, the motors behaved differently even though the same voltage was provided. This physical constraint was analyzed and corrected. The team also realigned the wheels and used fuses to protect the valuable hardware from any abnormal electronic behavior.

3. Innovations

BeaverBot features a power distribution system that has a built-in configuration for charging the batteries with the simple flip of a switch. This feature is very useful because it eliminates the need for rewiring and reconfiguring the electrical layout and allows the team to charge all the batteries simultaneously.

Another feature of BeaverBot is the cost-effective wireless E-stop that the team devised, which performs very consistently in the range specified by the rules. A wireless garage door opener is used to trigger a relay which cuts off power to the motor driver.

The team also decided to diverge from using the SICK LMS family of laser range finders considering it too sophisticated and expensive for the task at hand. The Hokuyo URG-04LX instead, was used for obstacle detection for its compact and lightweight design. This significantly decreased the size and weight of the robot. Another feature that interested the team is the low power consumption of this device, which significantly improved the battery life.

The artificial intelligence component of the vehicle relies on a simple bug algorithm, which uses an occupancy grid that represents object boundaries scaled to the proportions of the robot. The robot is then represented by a cell and the path planning becomes a graph search problem.

4. Team Organization

The design team was divided into various groups with individuals from different disciplines assigned to specific tasks. The mechanical engineers had the task of designing BeaverBot's frame while electrical engineers and computer scientist worked on sensor integration and programming. In addition to this, the team had help from volunteers with the organization and management of miscellaneous tasks. To foster future improvements on the vehicle, the team founded a student organization in charge of developing autonomous navigation. The team organizational chart shown in Figure 4.1

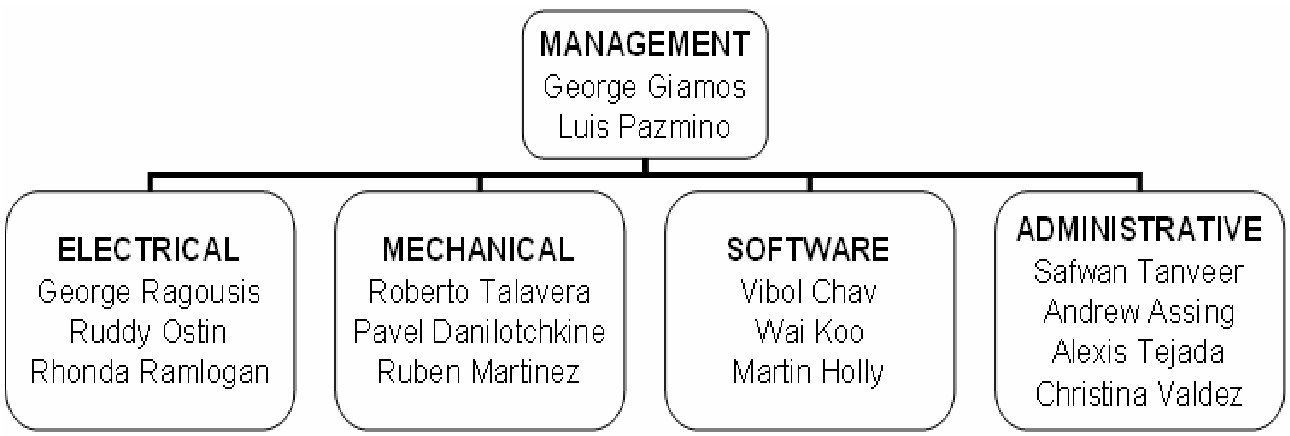

Figure 4.1 – Team Organizational Chart

5. Mechanical Design

The initiative behind the design was to optimize the turning radius, stabilization, maneuverability and making sure that the wheels could support the total weight of all the components on the robot. Through intensive simulation and testing, this frame has proven to meet the demands that it will be subjected to during the competition.

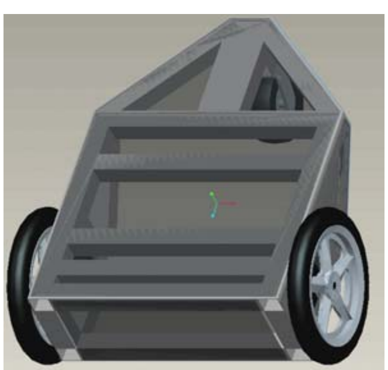

Figure 5.1 – Vehicle chassis

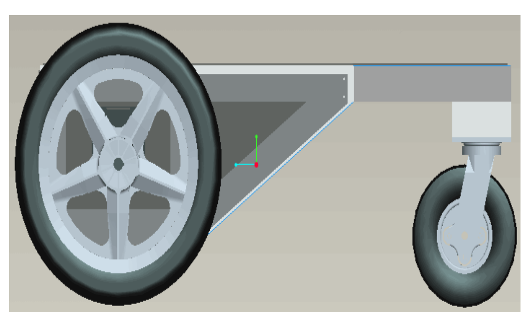

Figure 5.2 – Vehicle chassis – side

5.1 Vehicle Chassis

The chassis of BeaverBot, shown in figure 5.1.1 has undergone intense simulations and testing. The material chosen for the frame is 1/8" square tube aluminum 6061. Aluminum reduces total weight, is strong enough to withstand the stresses generated within the structure and is easy to work with during construction. This allowed us to reduce cost and weight substantially and provides a means for joining other components via screw fastening attachments. The total dimensions of the robot including the wheels are 37" length x 33" wide x 70" high. It weighs approximately 132lbs and is able to carry up to 180lbs.

5.2 Vehicle Drive Train

The drive train constitutes the most important aspect of the robot mechanical design; it will determine the maneuverability, speed and acceleration of the robot. Because of budget, simplicity and maneuverability, the team decided to use a three wheel differential drive with a rear caster wheel. Having a rear caster wheel design provides the vehicle with translational motion in any desired direction and allows the vehicle to perform a complete (360 deg) rotation about its center. In other words, this configuration allows the vehicle to turn with a radius that is half of its length if it rotates about its axis of symmetry and a radius that is its width if it rotates about the left or right wheel. To translate the motor rotation into vehicle motion, the team decided to use a simple sprocket-to-sprocket, motor-to-wheel interface using chains. This simple motor to wheel interface helps alleviate stress coming from the motor shafts and allow a larger ground-to-vehicle clearance.

6. Electrical Design

6.1 Power Distribution

BeaverBot uses a total of three 12 volt sealed lead acid batteries rated at 26Amph each (see Figure 6.1.1). The batteries are located on the rear of the robot to shift the center of gravity. Each battery is protected with an inline fuse rated at 15 Amperes attached to every positive terminal while the negative terminals are connected to the vehicle aluminum chassis as ground. One battery is used to output 12 volts contributing power to the GPS, the DC to DC converters (discussed on the following section) and a small fan for the control box. The other two batteries are connected in series providing a total of 24 volts to the motor driver and motors.

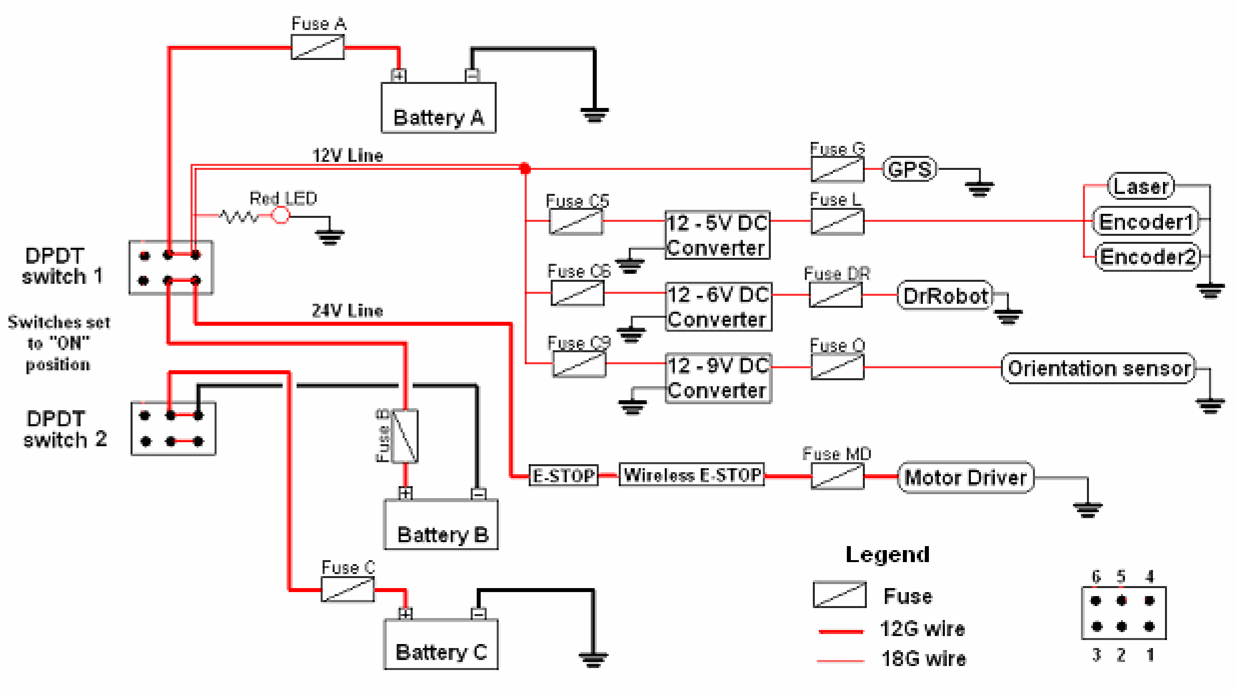

Figure 6.1 – Electrical Layout for battery configuration

The main power switch consists of 2 DPDT (double pole, double throat) rated at 20 Amperes, located in the front left of the robot. The switches are twined up (coupled) with a metallic plate forcing them to have the same position. ; Both ON or both OFF. Two DPDT switches are required to achieve the desired 12V & 24V configuration circuitry in addition to “easy-charging” discussed in the following section. The figure below shows the circuitry when the main power switch is ON. There is a red LED signifying the ON position directly below the switch.

6.2 Voltage Regulation

A total of three DC to DC Converters are used to regulate and step down the 12 Volts from battery A to the desired value according to the input voltages specified for the following components:

- A 12V to 5V converter is used for the LRF and the wheel encoders
- A 12V to 6V for the motor controller Dr Robot
- A 12V to 9V for the Orientation sensor according to component specifications.

All converters are mounted on the outside of the control box (discussed on the next section) and are protected with fuses rated 2 Amperes.

6.3 Charge System

To charge the robot simply set the main switch to "Charging" position and place the positive (+) lead from the charger to the bracket located to the front right of the robot. Place the negative (-) lead from the charger anywhere on the chassis ground. The following figure shows the charging circuitry using the 2 DPDT switches to reconfigure the three batteries from their existing configuration to an all-parallel one. This is referred to "easy-charging" accomplishing convenient battery charging without having to rewire batteries B and C in order to place them in a parallel configuration. A green LED is signifying "Charging" mode directly located above the main power switch.

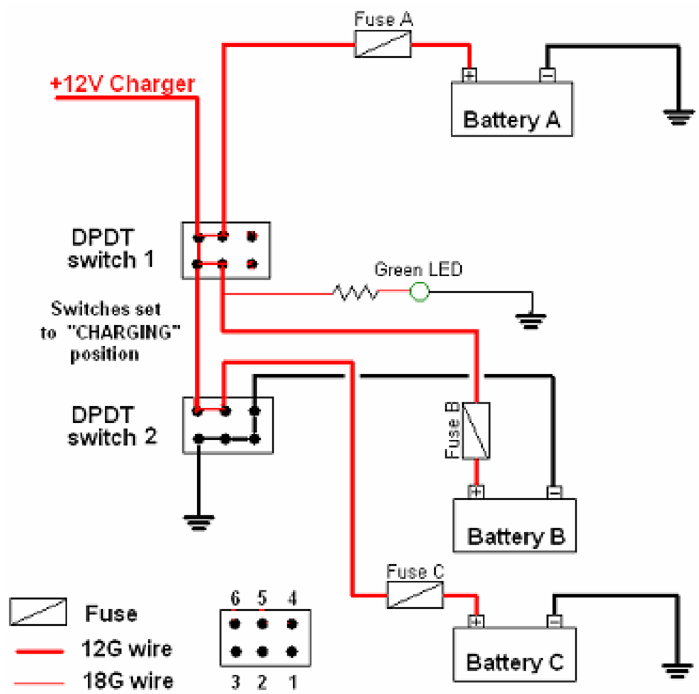

Figure 6.1.2 – Electrical Layout for easy charging mode

6.4 Control Box

The Control Box is the black box located above battery A in the rear end of the vehicle and it contains Dr. Robot motion controller circuit board, the Sabertooth motor driver and a small PC fan to keep the box cool in case of overheating. This box is essential to protect the uncovered circuit boards necessary in our design.

6.5 Emergency Stops – Mechanical E-Stop

The E-stop button must be a push to stop, red in color and a minimum of one inch in diameter. It must be easy to identify and activate safely, even if the vehicle is moving. It must be located in the center rear of vehicle at least two feet from ground, not to exceed four feet above ground. Vehicle E-stops must be hardware based and not controlled through software. Activating the E-Stop must bring the vehicle to a quick and complete stop.

6.6 Emergency Stops - Wireless E-stop

For the Emergency Stop requirements a 12 Volt DC relay is used to deactivate the motor driver and disconnect power to the motors forcing the vehicle to a complete stop. The relay is triggered from both a manual red switch located in the back of the robot, and also from a wireless secondary relay mechanism used to open garage doors remotely. The schematics are the following.

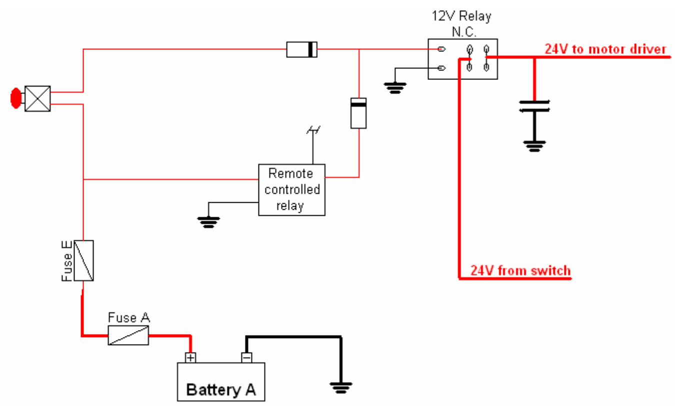

Figure 6.1.3 – Circuit schematics for mechanical and wireless e-stop

7. Sensors

The Laptop is the brain of BeaverBot where all the vital components are connected to it through USB or Serial. The Hokuyo Laser, Novatel GPS, Logitech Quickcam, and Dr Robot motor controller all connect to the Laptop’s four USB ports, while the Microstrain orientation sensor connects the laptop’s serial port. The Dr Robot Motor Controller was the only component that the team needed to convert its serial communication to USB. The laser and GPS both featured built in serial to USB configurations that eliminative this integration condition. In addition to the components discusses above, BeaverBot also features two Bison 24V DC Motors, a Sabertooth Motor Driver, two 2-channel optical encoders. The Motor connects directly to the motor driver, which is controlled by the Dr Robot Motor Controller. The optical encoders additionally connect to the motor controller, where that information is used as feedback to precisely control the motors speed.

Components and Sensors	Description
Dell Latitude D820 Laptop	The D820 Laptop features an Intel Core 2 Duo 2GHz Processor, 2.0GB of Memory, and 80GB Hard Drive.
Hokuyo URG-04LX Scanning Laser Rangefinder	The URG-04LX is a small and accurate laser able to report ranges from 20mm to 4m (1mm resolution) in a 240° arc.
Novatel ProPak-V3 GPS Receiver	The ProPak-V3 is a durable, high performance receiver, capable of receiving sub-meter positioning accuracies.
Logitech Quickcam Pro3000	The Quickcam Pro3000 utilizes a 640 x 480 resolution with a 30 frames per second refresh rate.
Microstrain 3DM-G Orientation Sensor	3DM-G is capable of output an orientation with a resolution of 0.1˚ and an accuracy of ±2˚ in dynamic conditions.
Dr Robot PMS5005 Motor Controller	The PMS5005 controller features the ability of integrating a wide range of sensors and can provide precise PID control.
Dimension Engineering Sabertooth 2x10 Motor Driver	The Sabertooth is a versatile motor driver that can provide 8A of continuous current per channel while running 24V motors.

Figure 7.1 – Summarization of sensors used

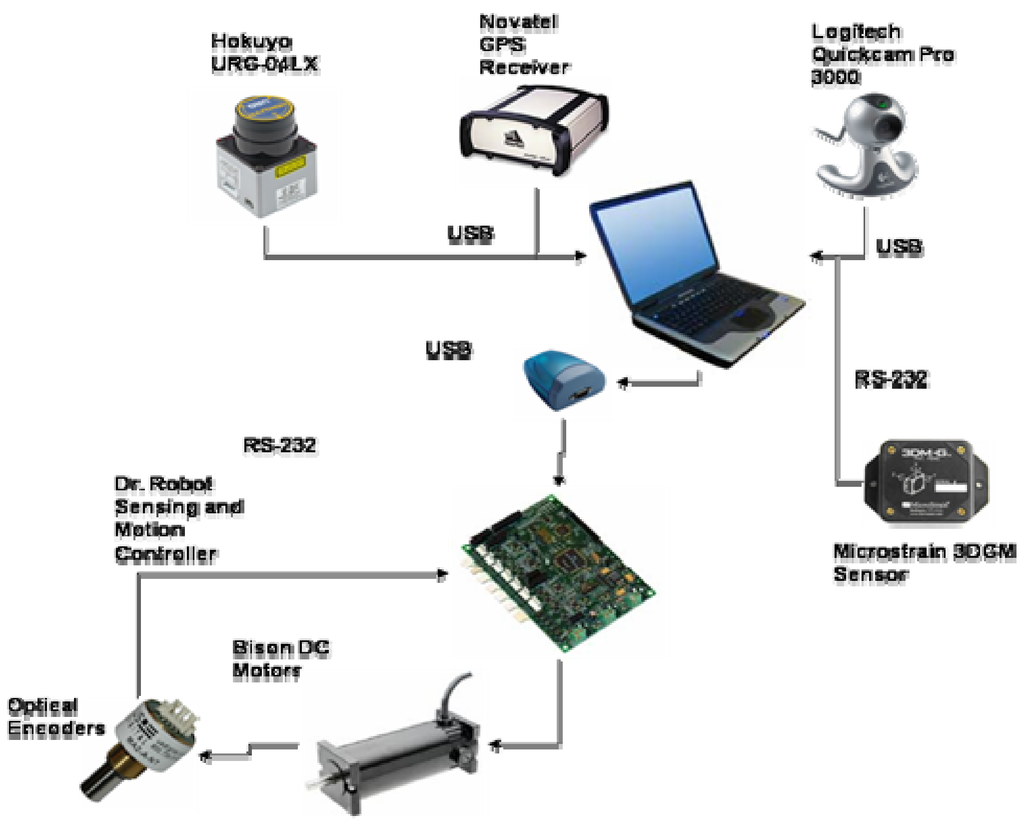

Figure 7.2 –Integration diagram

8. Software

The autonomy of the robot is the most important aspect of the design, so the algorithms used are vital to the performance of the robot. To increase productivity and for easier reconfiguration, the team decided to use popular software that have already built in libraries with algorithms used in mobile robots already. The team considered among Labview, Matlab, ARIA software, and Evolution Robotics. Among the reasons to choose the software were: processing power, ease of programming, ability to integrate with the hardware, and platform compatibility. Our final decision encompassed a windows-based platform with fast processing of information and a robust library, which could save programming time. Evolution Robotics was able to comply with all these prerequisites because it is C++ based. It also has many APIs that can be used with a command line and save processing power. At the same time, it presents a user-friendly interface with remote control, built in navigation and pattern recognition algorithms.

The algorithm starts by initializing all the communication ports, and then it obtains the desired heading by processing the image in front of the robot. The robot is rotated to match the heading relative to the camera. The laser range scanner is then called to map the front of the robot. An occupancy grid is generated with the data, and through iteration the best path is calculated.

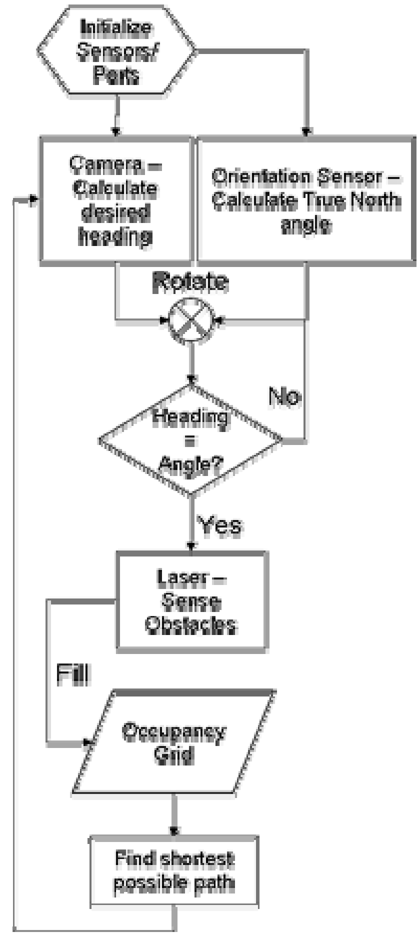

Figure 8.1

8.1 Autonomous challenge lane following

A vision algorithm designed by Andrew Reed Bacha from Virginia Polytechnic Institute was implemented for the QuickCam Pro 3000. These algorithms can analyze camera images and extract information that can be used to recognize an object, to detect motion, or to detect skin (for detection of people).

The algorithm was expected to perform a number of tasks. First, the camera was to take a picture. Once this picture is taken, it is scaled down, and then the image is cut in half. Then, the image is converted into special grayscale (mixed channel method) to eliminate any brightness on top of the image taken.

The vision algorithm also had the task of finding the brightest pixel on each row and column of the images. To tie this all together, the Hough transform is used. For each brightest pixel on each row and column, the Hough transform algorithm found the distance from the origin and the angle with respect to the x-axis. The direction for the robot to move is determined using a decision tree based on lines detected by the Hough transform and the slope of the line.

8.2 Obstacle detection and avoidance

The obstacle avoidance algorithm relies on the Hokuyo Laser Range Scanner to map the environment up to 9 feet in front of the robot with a range of 180º. The map is then converted to a grid using 3x3 inch squares. Each point detected is expanded on all directions to the size of the robot. By doing this the robot is represented by a single cell, and the path planning becomes a graph search problem.

The graph search algorithm goes straight until it reaches a cell with "Occupied value". Once this happens, it examines the neighboring cells to circumvent the obstacle. The process is repeated various times, each time taking a different predisposition, either left or right to create different paths. Each path is given a cost value that determines the shortest path to follow. Once the shortest path has been established, the program breaks it down to commands for each wheel. The function which processes the commands uses time derived from the number of cells in each direction and the constant velocity provided to the motors.

8.3 Navigation

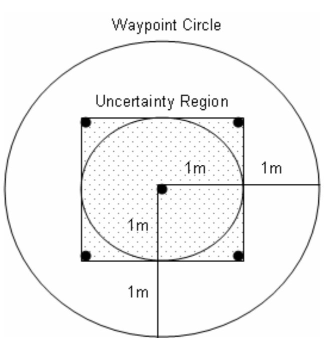

Figure 8.2

The BeaverBot uses a Novatel Propak-V3 to receive GPS coordinates. The ProPak-V3 features integrated L-band corrections from geosynchronous satellites such as OmniSTAR and CDGPS. With these corrections, sub-meter accuracies can be achieved. The Autonomous Vehicle Team decided to use CDGPS satellite network during IGVC competition because of its accuracy and proximity to Canada. If the network is unavailable or inaccessible BeaverBot will switch to the Wide Area Augmentation System (WAAS). The 95% accuracy of the WAAS is approximately 1 meter. Given this inaccuracy, the BeaverBot will verify that is at a given position by validating that the uncertainty region is at most within 2 meters of the desired coordinate. Figure 8.1 shows this verification process. The uncertainty region is created by expanding the coordinate's standard deviation given by the receiver. Once the uncertainty region is created BeaverBot examines the four corners of the region to determine if it is in the prescribed location.

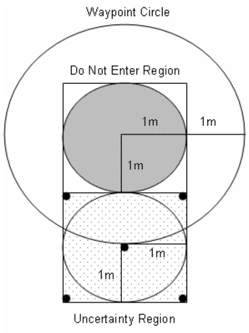

Figure 8.3

The Navigation Challenge requires the robot to avoid entering the 1 meter range of the last GPS coordinate. After careful analysis of this requirement, it was determined that, given the inaccuracy of our GPS receiver, there was less than a 50% chance BeaverBot would make it to that location. Figure 8.2, illustrates this problematic scenario.

8.4 Navigation Algorithm

In the 15th annual intelligent ground vehicle competition (IGVC), the navigation course will be split in half by a fence with a space large enough so that the vehicle can travel between the two areas. GPS coordinates will be placed above and below this fence. To tackle this particular course, BeaverBot will use the following algorithm

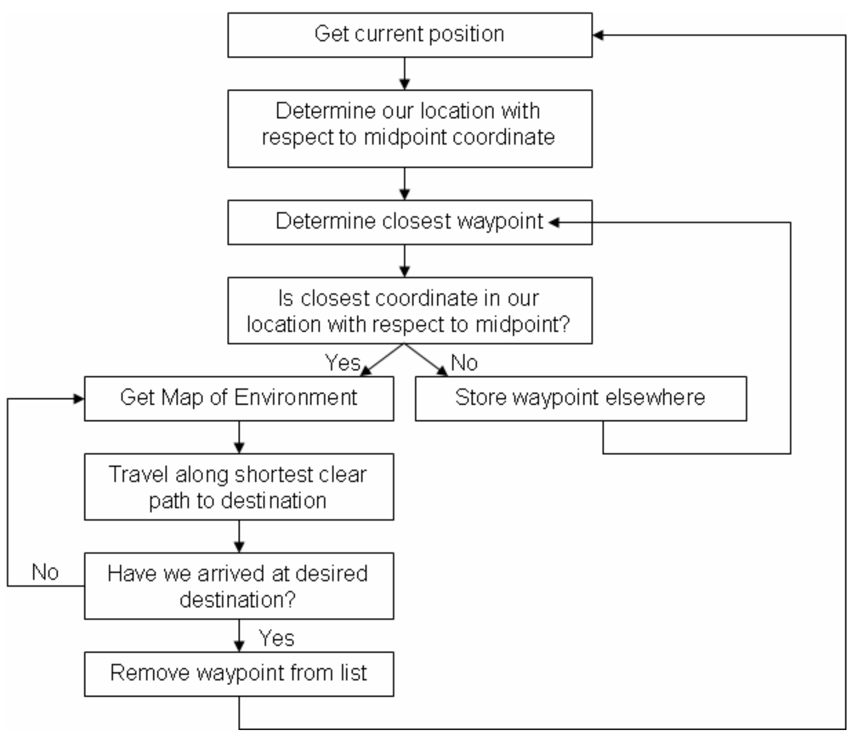

Figure 8.4 – Flow chart for navigation algorithm

9. Specifications

9.1 Safety

Sealed lead acid batteries are used to reduce the risk of hazardous material spillage during operation of the vehicle. The whole robot is enclosed in a metal sheet that will protect bystanders from any fire. The design of the wheels partially covers the chains so that no object could be caught in the chains. The e-stop was tested thoroughly to make sure that it would not fail under sever conditions.

Speed	Maximum speed: 4.5 MPH
Obstacle Detection Distance	Maximum: 15 feet Optimal: 9 feet
Reaction Time	500 milliseconds
Ramp Climbing Ability	A gradient up to approximately 20%

Figure 9.1 Battery Life

The table below lists the vehicle's components and power requirements. These estimates are determined from the maximum power consumption specification. From the data listed, the team estimates a runtime of over 3 hours.

Components	Voltage (Volts DC)	Supplied by:	Current (Amperes)	Max Power (Watts)
Laptop	N/A	Laptop Battery	N/A	N/A
Hokuyo Laser	5	DC Converter	0.5	2.5
Dr Robot Motor Controller	6	DC Converter	1	6
Sabertooth Motor Driver	24	Batteries B&C	.5	12
Camera	N/A	USB	N/A	N/A
Novatel GPS	12	Battery A	N/A	2.5
Orientation Sensor	9	DC Converter	0.05	0.5
Encoders (x2)	5	DC Converter	.05	.25
Bison Motors (x2)	0-24	Motor Driver	5	240
Total Power				347.625

Figure 9.2 – Vehicle components and power requirements

10. Vehicle Cost

The team was able to raise the majority of the budget through donations from the student government, faculty and school of engineering.

Component	Retail Price	Cost to Team
Novatel GPS	$6,670	$2,733
Hokuyo Laser Sensor	$2,703	$2,703
Dell Latitude D820 laptop	$1,478	$1,478
Microstrian Orientation Sensor	$1,495	$0
Bison 24V Motors (x2)	$616	$616
Dr Robot Motor Controller	$485	$0
Mechanical Accessories	$433	$433
Aluminum Frame Tubing	$303	$303
Electrical Accessories	$234	$234
Motor Encoders (x2)	$202	$202
12V 26Ah Batteries (x3)	$114	$114
Sabertooth Motor Driver	$89	$89
Battery Charger	$69	$69
Logitech Quickcam Pro	$65	$65
DC to DC converter (x3)	$58	$58
16in wheels (x2)	$38	$38
Wireless Relay	$23	$23
8in caster wheel	$19	$19
USB to Serial adapters	$15	$15
Total	$15,109	$9,192

Figure 9.3 – Component cost breakdown

11. Conclusion

BeaverBot has been a great learning experience for the team and for CCNY at large. This is the first time a team from CCNY will compete in this competition and we hope to foster an event for future students interested in robotics to undertake. The design process was very difficult due to the team's inexperience in robotic design. A project of this magnitude is never fully realized until one is faced with it. The team encountered many problems in the implementation and integrations process. However, with hard work and perseverance we were able to rise to the occasion and produce a prototype of CCNY's first autonomous vehicle.

A simple yet reliable vehicle was fabricated thanks to the collaborative efforts of the many disciplines at CCNY. The fundamental mechanical and electrical components have successfully been designed and fabricated. Remote operation was implemented for testing before any autonomous tasking. The programming team was comprised of mostly electrical engineers and had very little programming experience making the programming task a rather long research. The team hopes that future students will be able to devote their time on testing and improving their autonomous algorithms on a pre-built robotic platform.