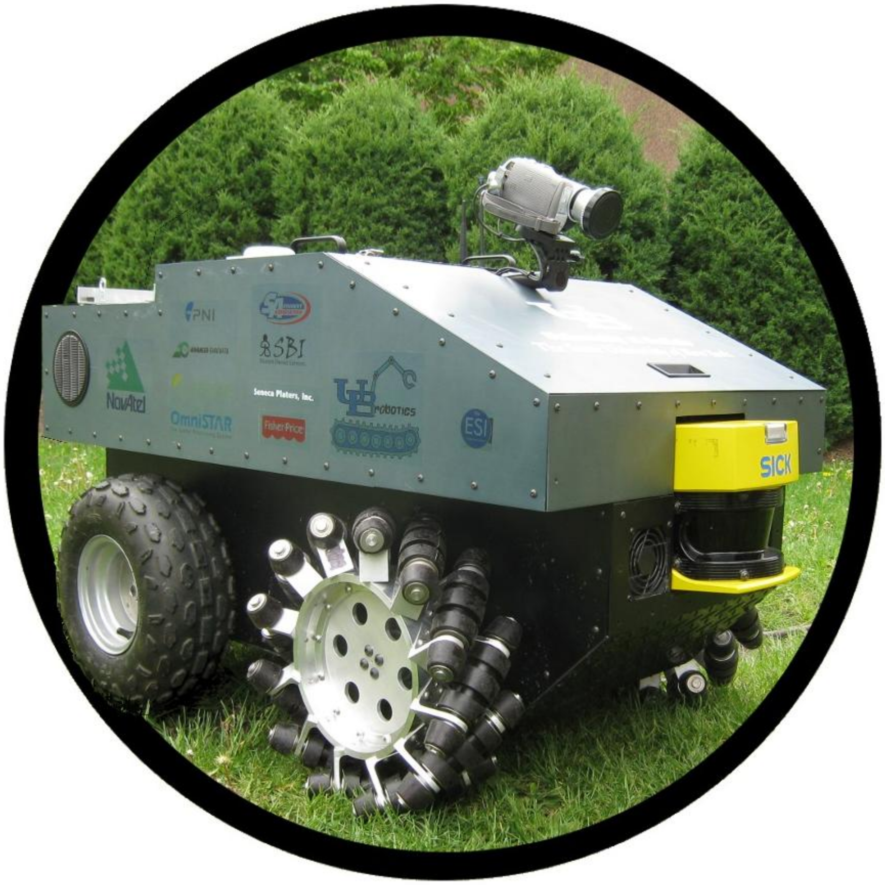
Big Blue



2010 Intelligent Ground Vehicle Competition

Dominic Baratta, Brett Bowman, Dylan Conway,
Ben Deuell, Colin Lea, Christian Nugent, Nathan Ohmit, Bich Vu

I certify that the engineering design of the vehicle described in this report was done by the current student team and has been significant and equivalent to what might be awarded in a senior design class.

Dr. Jennifer Zirnheld
Department of Electrical Engineering
University at Buffalo

1 Overview

UB Robotics, an undergraduate student-run organization at the University at Buffalo, presents substantial revisions to **Big Blue**, a robot that was first introduced in the 2009 Intelligent Ground Vehicle Competition. Efforts have been made to enhance the previously created unmanned ground vehicle in order to leverage the countless hours already dedicated to the platform. At the 2009 competition, Big Blue placed 12th overall and successfully completed the Interoperability Challenge. The goal for 2010 was to become more competitive in both the autonomous and navigation challenges.

Aims for the 2009-2010 school year stem from feedback received from the previous competition as well as problems seen in the exhaustive analysis and review process. Notable changes have been made to the drive mechanisms, electronics, software algorithms, and safety mechanisms for operation. The entire platform will be documented and major changes will be noted with a ★ symbol.

1.1 Team Structure

After the 2009 competition, many key seniors graduated making UB Robotics a relatively young team. Current members range from freshman to juniors pursuing their undergraduate education. Many new areas were investigated and the club's recent accomplishments represent a comprehensive understanding of the field. The IGVC team structure is as follows:

Project Leader *Colin Lea, ME ‘11*	
Hardware Leader *Ben Deuell, ME ‘12*	**Software Leader** *Dominic Baratta, CS ‘12*
Christian Nugent, EE ‘12	Colin Lea, ME ‘11
Brett Bowman, EE ‘12	Bich Vu, CSE ‘13
Dylan Conway, ME ‘12	
Nathan Ohmit, CSE ‘12	

ME = Mechanical Engineering, EE = Electrical Engineering
CSE = Computer Engineering, CS = Computer Science

Figure 1 Team Structure

1.2 Design Process

Figure 2 represents the two-year design process that UB Robotics has employed. The flow represents an iterative approach emphasizing simulation and testing. When possible, physical prototypes are tested before spending large amounts of time and resources manufacturing full-scale components. Simulation is used in all domains whenever possible; CAD for mechanical design and software simulation for algorithm development.

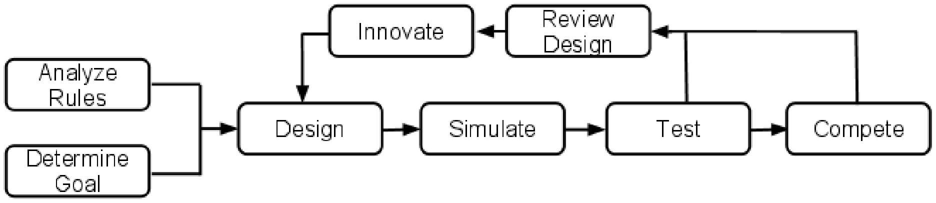


Figure 2 Overview of the Design Flow Implemented by UB Robotics

UB Robotics feels that outreach and dissemination of information is important for promoting the field of robotics as well as self-reflection. By developing tutorials and workshops on tools useful to the competition, students not only develop a deeper understanding of the content they are teaching but are able to help others learn valuable skills. This also leaves a legacy, which aids in documentation and assists new members in climbing the learning curve. Tutorials are available in both written and video format on the UB Robotics website [1]. Additional demonstrations have been done this year at the Buffalo Public Library, Buffalo Museum of Science, and regularly at University at Buffalo.

1.3 Focus Areas

At the 2009 competition, Big Blue faced critical hardware problems that almost stopped it from competing. The custom motor controllers had problems due to large current draws causing some of the transistors to overheat and malfunction. Too much traction on the tires caused the motors to stall. These problems were not experienced in preliminary testing and resulted in the loss of important practice time at the competition. Two solutions were investigated to limit the amount of current being drawn. One is in the introduction of Mecanum wheels and the other is encoder-based feedback control.

The previous UB Robotics software team built a hardware abstraction layer with many of the features necessary for robust autonomous navigation. Significant work for the 2010 competition has been done in advancing motion planning and path generation to ensure success. Efforts for standardization using ROS (Robotic Operating System) were pursued but ultimately not used.

IGVC 2010 Focus Areas

Hardware

- Feedback Control
- Wheel Design
- Circuit Diagnostics

Software

- Path Planning
- Motion Planning

It is estimated that over 2500 voluntary man-hours have been put into Big Blue over the past year without class credit or monetary compensation. Weekly meetings are held to discuss updates and open hours are hosted regularly to facilitate active membership. Integrating the hardware and software teams is important for physical development and implementation, thus joint weekly meetings are usually held.

2 Mechanical Design

An in depth background of Big Blue's chassis and general hardware design can be found in the 2009 IGVC Technical Report [2]. In this document, we will focus on technical details and highlight recent innovative efforts. All hardware designs were first developed using Computer Aided Engineering tools such as Autodesk Inventor and PCB Artist. A complete test platform was developed to prototype the new additions before the full-scale models were manufactured.

2.1 Chassis and Drive Train

The design goal of Big Blue's chassis and drive train was to establish a rugged, reusable platform capable of navigating diverse outdoor terrain. A four wheel direct drive scheme was used to increase speed capabilities and provide zero point turning. Zero point turn is especially important for software control in order to simplify the motion planning process. Additional consideration was placed on keeping a low center of mass and making components easily accessible.

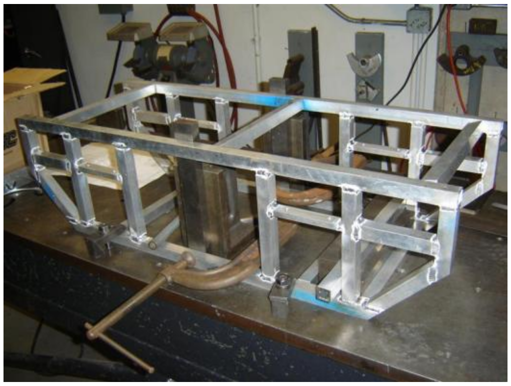


Figure 3 Bottom Portion of the Chassis

The chassis was developed with an upper and lower half. Heavy parts such as motors and batteries are placed in the bottom half, and circuit boards, sensors, and the computer are placed in the top portion. The welded frame was manufactured using 1" square tubing. Finite element analysis within Autodesk Inventor was used to confirm structural integrity [2].

Big blue uses four NPC Robotics T64 brushed DC motors running on 24V with an output of over 0.7 horsepower. Experimental results show the vehicle can travel at speeds up to 10 miles per hour. This is restricted in firmware to 5 miles per hour for competition.

2.2 Mecanum Wheels

In order to navigate a curve a four-motor differential drive system requires wheels to slip. This causes localization issues, puts added stress on the motors, and requires greater amounts of current to navigate. Problems such as these were not fully taken into account during the original design of Big Blue. These issues have been addressed with the introduction of custom Mecanum wheels.

Mecanum wheels have a series of rollers that are placed along a wheel hub at 45 degree angles, which allow the vehicle to move forward and laterally [3]. Recent publications demonstrate vehicles with Mecanum wheels attached to all four motors allowing movement in any direction [4]. Note that the goal of using these wheels was not to develop a non-holonomic vehicle, but to turn with greater efficiency and control. Putting them only on the front motors increases mobility and decreases current draws on the motor controllers. Slipping is no longer a problem, meaning the wheel encoders provide significantly more reliable localization data.

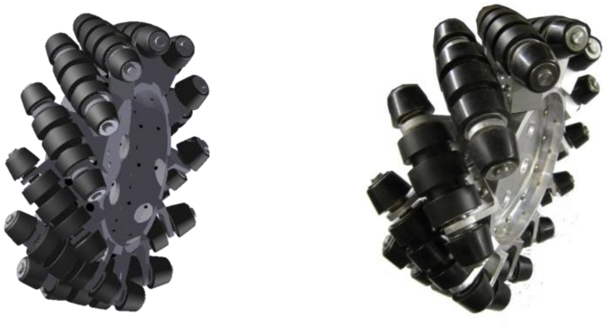


Figure 4 Mecanum Wheels (a) CAD Design (b) Final Product

The size and ruggedness of wheels required for Big Blue are unavailable through commercial-off-the-shelf (COTS) solutions, thus the design was developed and manufactured in-house by UB Robotics. Considerations were placed on ruggedness and durability. The rollers on the COTS wheels investigated are continuous and are meant for smooth, indoor surfaces. Grooves in the UB Robotics design provide greater traction for the competition's outdoor environment. Each wheel has twelve rollers equally distributed on 8.5 inch rims. Rollers are made of a two part urethane cast in a silicone mold that was molded around a custom aluminum master part created on a CNC lathe. The design was developed in Autodesk Inventor (figure 4a) and then imported into FeatureCAM to generate the tool paths and code for the lathe.

2.3 Sensors

Big Blue houses a suite of differential and absolute sensors used to determine its location, vehicle motion, and objects on the course. A Novatel ProPak-V3 differential GPS is used to track the global position. The GPS is WAAS-enabled and obtains high accuracy data with three standard deviations of 10 centimeters using the support of an Omnistar HP subscription. A PNI 3-axis digital compass with pitch/roll compensation is used to determine the current heading with resolution of 0.1°. Each motor has a US Digital E4 Wheel Encoder which provides a resolution of 2560 clicks per revolution at the wheel. The sensors are filtered with an Extended Kalman Filter which is discussed in section 3.6.

Objects are detected on a 2D plane using a SICK PLS101 laser rangefinder (LIDAR) which offers 50 meter range over a field of 180° with resolution of 0.5°. For competition the scan range is reduced to 4 meters for computational reasons. At this range the LIDAR has an accuracy of approximately 7 cm. For lane detection in the autonomous challenge a Panasonic 3CCD video camera with a 37 mm wide angle lens is used.

2.4 Power Supply

A custom power supply board was previously created to supply power to each component. Four rails distribute power at 24, 12, and 5 Volts. The 24V rail is an unregulated source connected to the motor controllers. The rest of the rails are regulated and are enabled with individual channel switches. A soft-start circuit charges the capacitors before starting the main system. A keep-alive circuit was created for the GPS to eliminate the need to reconnect to satellites every time Big Blue restarts.

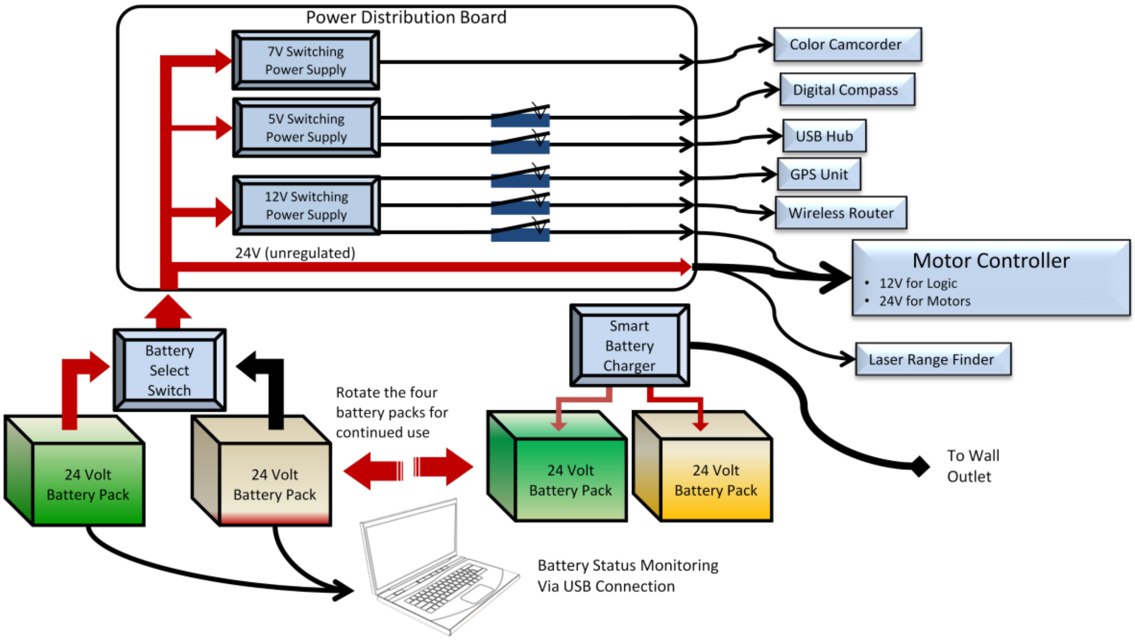


Figure 5 Power Distribution

2.5 Batteries

Two 24V battery packs are installed in the vehicle at a time. Each pack contains two 12V lead acid batteries in series. Power can be switched between packs without shutting the system down. A total of four battery packs are rotated successively in order to age them at the same rate. A battery monitor was custom designed to monitor the voltage levels in order to maximize performance.

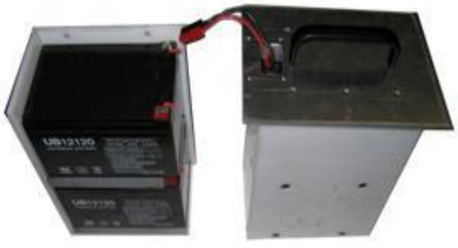


Figure 6 Uncovered and enclosed battery packs

2.6 Motor Controllers

A custom controller board was designed for the 2009 competition in order to implement Big Blue's drive train. It features interchangeable H-bridges capable of driving each motor up to 50V and 30A. ★This year temperature and current sensing capabilities were added as a safety precaution to prevent chips from overheating and malfunctioning. Motor commands are received from the computer and encoder data is transmitted back over a USB interface. As a safety measure the emergency stop cuts power to the motor drivers via logic gates rather than through firmware.

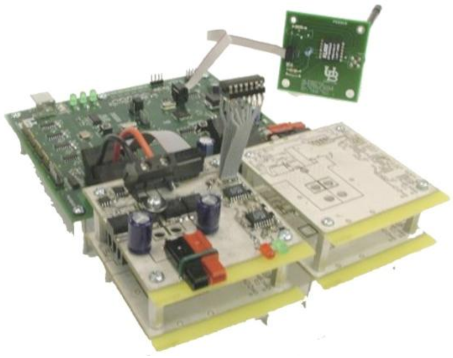


Figure 7 Motor Controller boards

2.7 Remote

A custom rapid-prototyped remote was previously developed for wireless communication with Big Blue. All information including the emergency stop, joystick positions, and menu buttons are sent as packets to a UART interface on the motor controller which uses a 418MHz RF Transceiver. The packets are filtered to check for errors in transmission resulting in a more reliable method of communication.

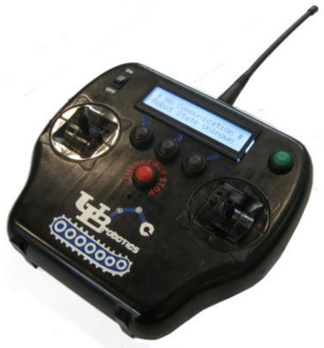


Figure 8 Wireless Remote

2.8 Power Consumption

The majority of power consumption comes from the motors. Figure 9 details the breakdown for components in Big Blue.

	Voltage (V)	Power (Watts)
Motors	24	480
LIDAR	24	17
GPS	12	2.8
Compass	5	0.1
Camera	8	5.5
Circuitry	5	0.6
USB Hub	5	2.5
	Total:	**508.5**

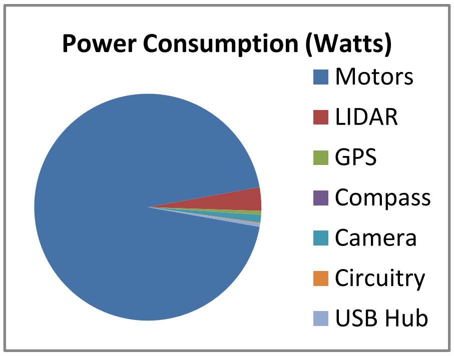


Figure 9 Power Consumption

3 Software Design

Significant work in the Fall semester was spent attempting to redesign the way UB Robotics handles software control. ROS (Robotics Operating System) and Player/Stage were evaluated in efforts to standardize on a design. The theory was that a robust open source architecture would be a more sustainable choice for long-term use. By using one of these solutions, focus could be spent on testing algorithms instead of writing low level code. Introductory documentation was written to facilitate learning for new members [5].

Despite these efforts, UB Robotics ultimately decided to switch back to RobOS, the custom software architecture written for the 2009 IGVC competition. Working with ROS in C++ was above the capabilities of the younger software team members who start off with formal training in Java. Additionally, there were problems interfacing with some of Big Blue's sensors which would have required substantial low-level work.

RobOS was developed targeting the Java SE 6 Development Kit to run on the vehicle's Dell Latitude Laptop with a dual core Intel processor and 2 GB of memory. All planning algorithms, driver support, and image processing has been created in Java and SVN repositories were used for multi-user development.

3.1 Autonomous Challenge ★

A local path planning algorithm has replaced the previous implementation of A* for the Autonomous Challenge. The new planner, Vector Polar Histogram Plus (VPH+), relies on a local map from the vehicle and attempts to find the safest and farthest direction in which to move. The algorithm is based on a published implementation [6] and is modified to suit the needs of this challenge.

3.2 VPH+

VPH+ works by transforming a set of data in polar form into a binary histogram. The data used is merged from the current LIDAR data and line boundaries marked by the camera. A function is used to find a "safe" distance in each direction, as determined by equation 3.2.1. This value is compared to the distance from the vehicle to the nearest object at each angle. Parameters V, θ_i, a, and D_{safe} are used for velocity, target angle, deceleration rate, and safety distance.

$$D(\theta) = \frac{V^2 \cdot cos^2(\theta_i - \frac{\pi}{2})}{2a} + V + D_{safe} \qquad \text{(eqn. 3.2.1)}$$

Target directions are determined by free spaces indicated by a "1" in the binary histogram. Targets are filtered based on an angular safety distance, eliminating choices that are too close to hazardous objects. In situations where there are less than a nominal number of targets an artificial point behind the robot is chosen. This requires the robot to turn around and search its environment.

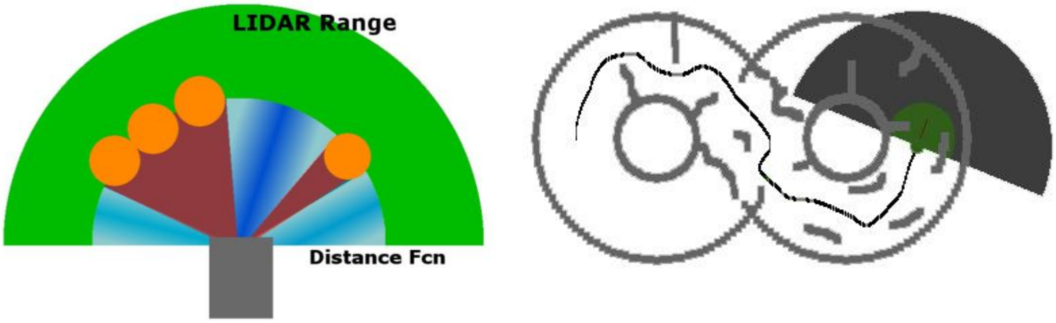


Figure 10 (a) VPH+ Diagram (b) Simulation using VPH+
Shades of blue indicate cost where darker blue is larger in value. Red denotes an invalid direction.

Points are grouped into different objects based on their proximity to other nearby points. If the distance between two sequential angles is less than a certain value it is concluded that they both belong to the same object. Directions encompassed by closer objects are eliminated from the target directions. This is the main advantage to the VPH+ algorithm over its predecessors VPH and VFH.

A cost function determines the final direction to move towards. Cost is developed with the idea that there is not a predetermined goal. Big Blue should move forward as far as it can while minimizing turning and maximizing safety. The safety factor is based on the angular distance to the closest "closed" angle. The cost function is shown by equation 3.2.2. The final direction is based on the maximum cost. Parameter θ_i refers to the target angle and K_s and $K\theta$ are tunable coefficients reflecting the weighting of the safety and heading factors. Figure 10 depicts the target directions and their calculated cost.

$$Cost(\theta_i) = K_s D_s - K_\theta \cdot (\theta_i - \frac{\pi}{2}) \qquad \text{(eqn. 3.2.2)}$$

This algorithm has proven to provide safer navigation over Big Blue's previous A* based method. Simulation in figure 10 shows the robot navigating a course without hitting anything. The closest obstacle comes 0.4 meters away from the robot on its side.

3.3 Lane Detection

Previous efforts finding lines on the ground in the autonomous challenge proved successful, thus Big Blue's image processing algorithm has not changed for the 2010 competition. In the Autonomous Challenge a series of filters is used to extract lines from the ground and transform them to Big Blue's mapping system. Figure 11 shows crisp white lines denoting the course boundary. Cones also get marked as boundaries; the bottom of the orange barrel and the legs of the sawhorse are visible in the post image. Based on this and subsequent images the vehicle chooses to take the path in between the sawhorse and barrels.

The problem of missing lines along the course has been investigated. Using VPH+, the vehicle searches for the direction that optimizes safety and resists change in direction. In all cases analyzed from previous competitions the vehicle follows the correct path. It is theorized that Big Blue could go off course only if there is a stretch of over 6 feet without lines and there are obstacles in its path directing it outwards.

Image Processing Procedure:

1. *Crop the top of the image*
2. *Brightness threshold*
3. *Remove green pixels*
4. *Grayscale filter*
5. *Highlight remaining pixels*
6. *Blur the image*
7. *Remove noise with a blob filter*

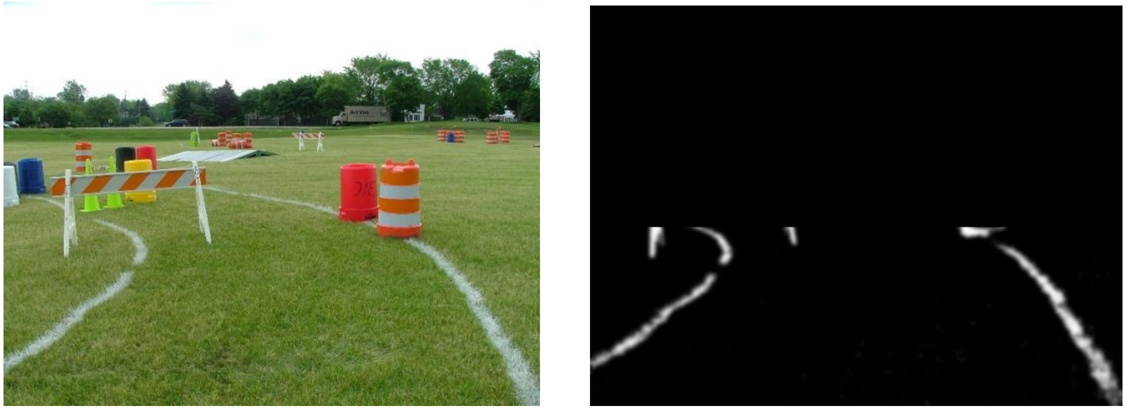


Figure 11 Lane Detection: (a) from camera (b) post processing

3.4 Navigation Challenge

Planning for the navigation challenge is done using both local and global path planning techniques. On a global level the A* algorithm creates paths connecting each of the waypoints while avoiding known obstacles. A* provides an optimal closed set path from the robot to the goal [7]. The path is augmented using VPH+ as described in section 3.2. For this challenge the cost function in VPH+ is adjusted to follow the direction of the line provided by A* rather than attempting to keep the current heading. Combining A* and VPH+ provides a safer way to avoid obstacles and also generates a smoother path.

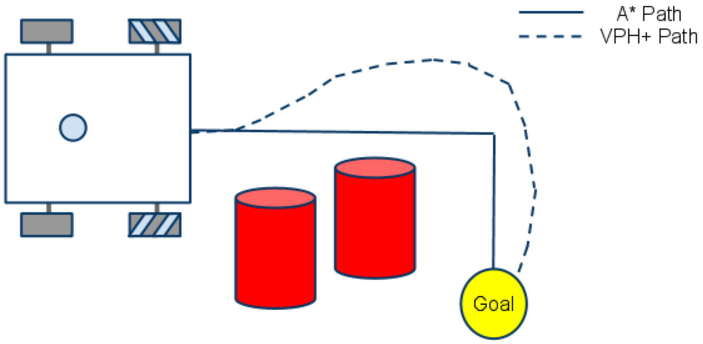


Figure 12 Comparison of A* and VPH+ using A* for guidance

3.5 Mapping

In both challenges the world is only partially observable. This is a greater problem in the Navigation challenge because Big Blue has to remember obstacles it has previously seen. At every time step a local map replaces information in the global map. This replacement technique is used instead of addition to reduce map smearing due to localization issues. The global map uses Java's Hashmap data structure to store objects. The Autonomous challenge relies on a local map including LIDAR data and lane boundaries from the camera. This is stored in a polar map used for VPH+.

3.6 Localization

An Extended Kalman Filter (EKF) is used to provide refined localization using GPS, odometry, and compass sensor data. The EKF is a Gaussian-based filter that linearizes the vehicular model with Taylor series expansion using the state model seen in equation 3.6.1 [8]. Covariance is calculated with respect to sensor measurements and the predicted state which is used to weight each input differently during the update phase. Redundancy in sensors by means of differential and absolute measurements provides more accurate localization data. For example, when the vehicle is not moving higher weighting is put on the encoders due to random deviations in GPS data. However, when the vehicle is turning the GPS is weighted more heavily because the encoders provide a less accurate motion model.

$$X = [x \; y \; \theta \; \dot{x} \; \dot{y} \; \dot{\theta}]^T \quad \text{(eqn 3.6.1)}$$

3.7 Vehicle Kinematics

The implementation of Mecanum wheels on two of the motors changes Big Blue's kinematics. To find the output velocity based on wheel encoders the inverse kinematics translating individual wheel rotational velocity to vehicle translational velocity must be formulated. Previous work has been done deriving kinematics for a design using four Mecanum wheels [4], but no known papers exist on vehicles using only two of these wheels.

The pose with respect to the translational and rotation velocities is defined as $\dot{X}_v=[\dot{x}\ \dot{y}\ \dot{\theta}]^T$ with wheel speeds $\dot{\theta}_w=[\dot{\theta}_{11}\,\dot{\theta}_{12}\,\dot{\theta}_{21}\,\dot{\theta}_{22}]^T$ where T is the notation for transpose. The parameters L, W, and R_w are used for length from the centroid to front wheels, width from the centroid to side wheels, and wheel radius. *a* is the value *a*=W+L.

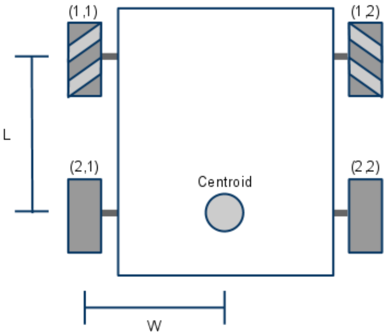


Figure 13 Kinematic Notation

The output velocity is calculated with respect to the Jacobian in the form $\dot{X}_v=J\dot{\theta}_w$ (eqn. 3.7.1) reflecting the relation between the individual wheels and current rotational velocity. The Jacobian is developed with the form:

$$J=\begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & 0 & 0 \\ -\frac{1}{a} & \frac{1}{a} & -W & W \end{bmatrix}\cdot\frac{R_w}{4} \qquad \text{(eqn. 3.7.2)}$$

3.8 Control Feedback

A Proportional-Integral-Derivative (PID) controller is used to govern Big Blue's wheel speed. It is assumed the motor varies linearly with voltage input. The output is dependent on the current error, rate of change in error, and accumulation of error as calculated by equation 3.8.1.

$$u(t)=k_d\dot{e}(t)+k_pe(t)+\int k_ie(t) \qquad \text{(eqn. 3.8.1)}$$

Closed loop control allows the vehicle to follow a trajectory with greater accuracy. It also prevents the motors from stalling and improves response time. Figure 14a overlays images of Big Blue tracing a circle over time. The red line is superimposed for visualization.

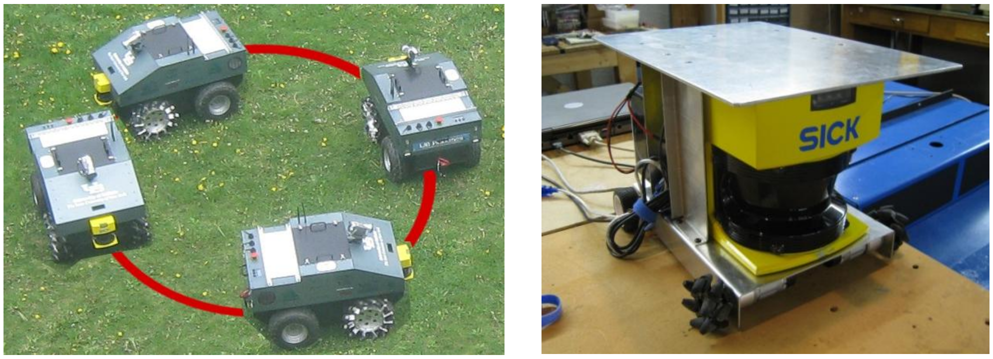


Figure 14 (a) Trajectory Tracing (b) Big Blue Test Platform

3.9 Testing and Simulation

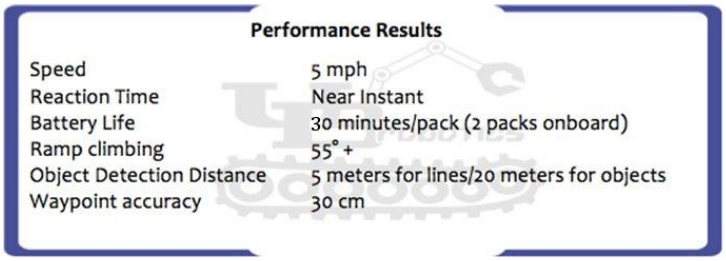
Before completion of Big Blues new hardware, testing was done in simulation and on a prototype robot. A test platform was constructed and evaluated with off-the-shelf Mecanum wheels. The kinematics were evaluated to test correctness and a LIDAR was employed to test path planning algorithms. A laptop running RobOS is placed on top of the vehicle seen in figure 14b.

3.10 Interoperability Challenge

Big Blue completed the Interoperability Challenge for the 2008 and 2009 competitions by implementing the JAUS protocol. In both occasions JAUS was tested using simulation code on the software platform. Additions this year include new request IDs to meet the updated specification and integration with Big Blue's complete hardware platform. This has enabled the JAUS subsystem to be able to issue commands to Big Blue in addition to sending out data.

4 Performance

Big Blue has exceeded expectations in regards to ruggedness and response. The vehicle can travel at upwards of 10 miles per hour and has ascended hills with an angle of over 55 °. The speed is capped at 5 mph by the motor controller microprocessors for purposes of competition. Additionally, Big Blue can climb six inch curbs with relative ease. It takes less than 1 second to go from stopped to full speed and about 0.5 seconds to go from active to stopped. Big Blue's response and speed come with drawbacks. Each battery pack lasts about 30 minutes. Thus, with its two on-board packs the total battery life is 60 minutes. Note that the introduction of Mecanum wheels has increased battery life by a factor of 1.5.



Performance Results	
Speed	5 mph
Reaction Time	Near Instant
Battery Life	30 minutes/pack (2 packs onboard)
Ramp climbing	55° +
Object Detection Distance	5 meters for lines/20 meters for objects
Waypoint accuracy	30 cm

Figure 15 Performance Results

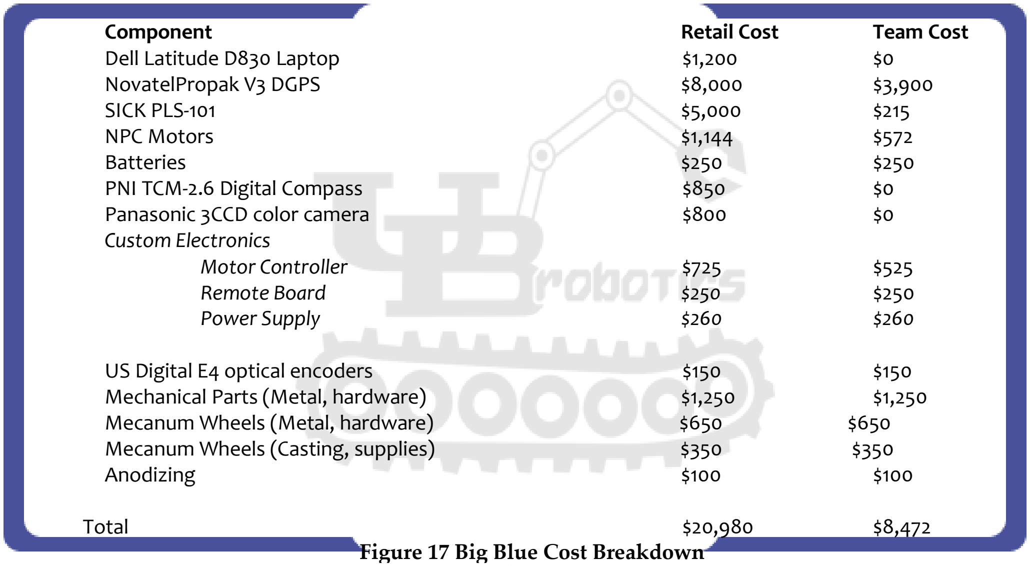
4.1 Course Complexities

Small complexities in both challenges make the courses significantly more difficult. UB Robotics has put consideration into all specified problems. Detection of potholes is dependent on the color of the hole. Big Blue's transforms filtered camera image data to a map, thus if the pothole is a color other than green it is added as an obstacle.

The implementation of VPH+ running on Big Blue compensates for dead ends. In safe situations the algorithm can always find multiple target travel directions. If there are less than a small specified number of targets then the vehicle turns around and detects an open path. Using a local path planning algorithm eliminates the goal seeking problem that global planners have with switchbacks. Because VPH+ resists turning (while optimizing for safety) it does not have this problem. Simulation (figure 10) shows that Big Blue successfully traverses switchbacks.

4.2 Cost

Big Blue is considered a research vehicle, thus its cost is substantiated by its high-accuracy sensors, well manufactured parts, and custom electronics.



Component	Retail Cost	Team Cost
Dell Latitude D830 Laptop	$1,200	$0
NovatelPropak V3 DGPS	$8,000	$3,900
SICK PLS-101	$5,000	$215
NPC Motors	$1,144	$572
Batteries	$250	$250
PNI TCM-2.6 Digital Compass	$850	$0
Panasonic 3CCD color camera	$800	$0
Custom Electronics		
Motor Controller	$725	$525
Remote Board	$250	$250
Power Supply	$260	$260
US Digital E4 optical encoders	$150	$150
Mechanical Parts (Metal, hardware)	$1,250	$1,250
Mecanum Wheels (Metal, hardware)	$650	$650
Mecanum Wheels (Casting, supplies)	$350	$350
Anodizing	$100	$100
Total	$20,980	$8,472

Figure 17 Big Blue Cost Breakdown

5 Conclusion

The addition of Mecanum wheels, Control Feedback, and more robust planning algorithms represents substantial change to Big Blue. UB Robotics is confident in their efforts and believes that these additions will ensure success in the 2010 Intelligent Ground Vehicle Competition.

Acknowledgments

UB Robotics would like to thank all its supporters who provided sponsorship and donations in the form of hardware, services, and financial support making Big Blue possible: Novatel, Omnistar, PNI, Advanced Circuits, Sunstone Circuits, Seneca Platers, Inc., UB Student Association, Sub Board-I, UB Engineering Machine Shop, and the Energy Systems Institute (ESI) at the University at Buffalo. Special thanks go to the club adviser, Dr. Jennifer Zirnheld, for her continued support, as well as the rest of the ESI staff and UB Robotics alumni for all of their help.

References

[1] *UB Robotics*. Available: http://www.eng.buffalo.edu/ubr

[2] B. D. D. Baratta, M. DiSanto, C. Lea, T. Montgomery, D. Muffoletto, C. Nugent, M. Tjersland, et. al. , "Big Blue: University at Buffalo's Entry into the 2009 Intelligent Ground Vehicle Competition " May 2009 2009.

[3] B. Ilon, "Wheels for a course stable self propelling vehicle movable in any desired direction on the ground or some other base ." U.S. Patent 3 876 255, 1975.

[4] S. Y. A. Akira, P. Viboonchaicheep, and K. Samura, "Mecanum-wheel Vehicle Systems Based on Position Corrective Control," in Proceedings of the Annual Conference of the IEEE Industrial Electronics Society, 2005.

[5] C. Lea. (2009, An Introductory Lesson on ROS. Available: http://sites.google.com/site/ubrobotics/ros-documentation

[6] Y. D. J. Gong, Y. Man and G. Xiong. , "VPH+: An Enhanced Vector Polar Histogram Method for Mobile Robot Obstacle Avoidance," presented at the Proceedings of the IEEE International Conference on Mechatronics and Automation, 2007.

[7] S. Koenig, Likhachev, M., Liu, Y. , Furcy, D., "Incremental heuristic search in AI," AI Magazine, 2004.

[8] W. B. S. Thrun, and D. Fox, Probabilistic Robotics. Cambridge, MA: The MIT Press, 2005.