

REDBLADE: A MULTIFUNCTIONAL AUTONOMOUS VEHICLE

Students: Ryan Wolfarth, Steven Taylor, Aditya Wibowo, Brandon Williams, Vincent Gatto
Advisors: Dr. Jade Morton, Dr. Peter Jamieson
Miami University

ABSTRACT

RedBlade is a multi-functional autonomous vehicle with two seasonal configurations which allow it to plow snow in the winter and mow grass in the summer. This vehicle participated in the Institute of Navigation's 1st Autonomous Snowplow Competition at the St. Paul Winter Carnival in Minneapolis where it won second place. It will again compete in the 8th ION Robotic Lawnmower Competition in June 2011. This report presents the design and implementation of the RedBlade mechanical system, sensor components, software architecture, control algorithm, and safety systems.

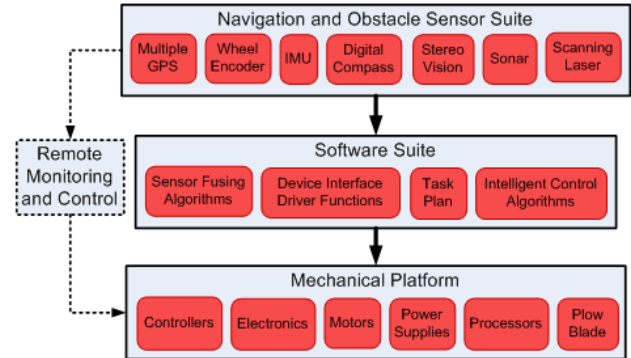


Figure 1. RedBlade three-layer system abstraction.

1. INTRODUCTION

Autonomous vehicles capable of performing many functions with accuracy and reliability in a timely manner are highly desired in modern society. RedBlade is designed as an expandable host to perform in multiple roles. It represents the next stage evolution of an autonomous lawn mower used in the ION Robotic Lawn Mower Competition. Since its inception as an in 2004^[1], RedBlade has been enhanced with the additional function of plowing snow. This paper describes RedBlade's mechanical, sensor electronics, control algorithms, and safety mechanisms required by the competitions.

The RedBlade system architecture consists of three layers as abstracted in Figure 1. The top layer is the navigation and obstacle avoidance sensor suite. The current generation of the RedBlade navigation sensor suite includes a Topcon Hiper Lite Plus GPS receiver, a MicroStrain 3DM-GX2 inertial sensor, and two optical wheel encoders as part of the integrated motor drive system. The current obstacle detection sensor candidates are touch sensors, ultrasonic sensors, and a scanning laser. A stereo vision system will be developed this summer for next year's competition. A second GPS unit on-board the rover will also be deployed to assist vehicle heading computation and to introduce redundant position solution for enhanced robustness for future competitions. The middle layer is the collection of software that provides driver functions for the sensors, sensor fusion algorithms, path planning, and vehicle motion control algorithm. The bottom layer is the mechanical platform, electronics hardware, including the motor controller RoboteQ, safety systems and power supplies, and processors that carry out the software functions.

RedBlade's immediate objective is to compete in the 8th Annual ION Robotic Lawnmower Competition to be held in Beavercreek, Ohio in June 2011^[2]. The problem statement is defined by the rules of the competition: RedBlade must be able to mow grass in two different 150 square meter fields of play^[3]. The first course is an open lawn with one randomly placed static obstacle which should be autonomously avoided. The second course contains a number of additional static obstacles and one moving obstacle that may appear at certain points on the map. This second scenario is depicted in figure 2.

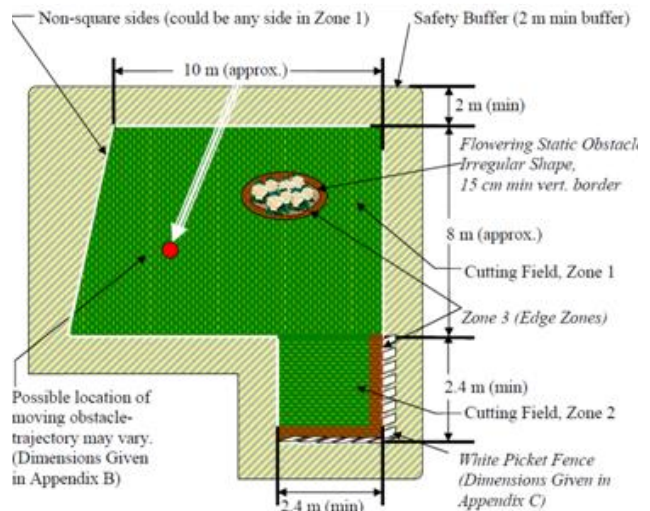


Figure 2: Illustration of the second course in the lawnmower competition (www.ion.org)

RedBlade utilizes the three navigation sensors (GPS, IMU, and tachometer) to determine its position, heading, and velocity (PHV). The vehicle PHV information along with its predetermined destinations is fed to an on-board computer that implements a Proportional-Integral-Derivative control algorithm to adjust the vehicle heading. The lawnmower blade and engine were taken from a Black & Decker CMM1200 electric mower. Both remote and on-board emergency kill switches allow an operator to stop all robotic motion.

The remainder of this report is organized as follows:

- Section 2: Mechanical Platform.
- Section 3: Electrical Components.
- Section 4: Control Algorithm Design and Implementation.
- Section 5: Path Planning.
- Section 6: Obstacle Detection and Avoidance.
- Section 7: Failure Modes and Recovery Actions.
- Section 8: Conclusions and Future Work.

2. MECHANICAL PLATFORM

RedBlade's mechanical platform consists of two drive wheels, three caster wheels for stability, and a metal chassis that houses the electrical systems. An overview of the mechanical platform for the wheels can be seen in Figure 3, a side view and a top view of the vehicle with mounted plow and all the components' physical dimensions can be seen in Figures 4 and 5.

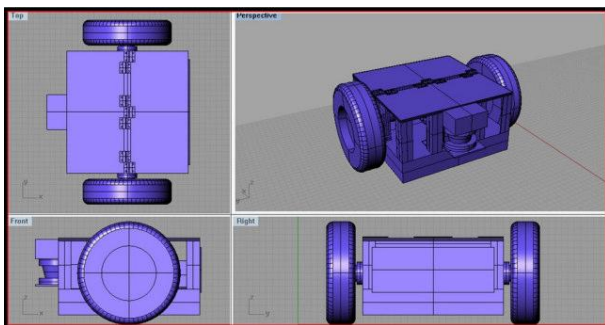
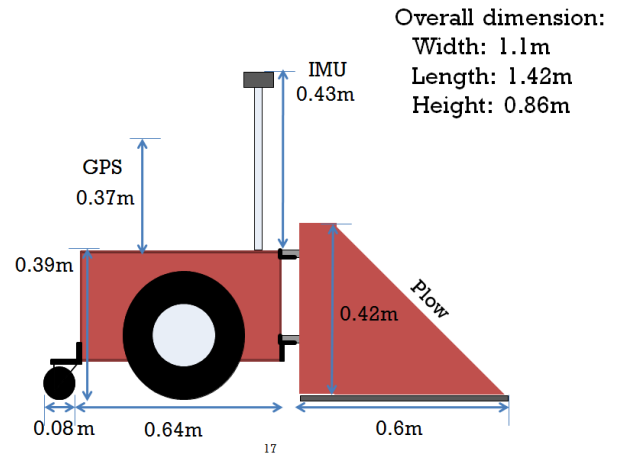


Figure 3: Drawings of the wheels and casing.

Figure 4. Side view of the mechanical platform with



mounted snowplow.

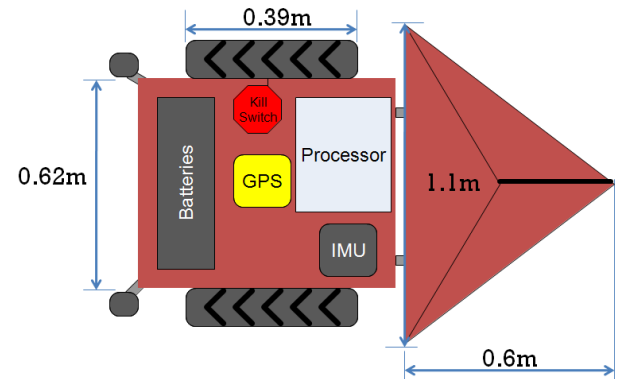


Figure 5: Top view of the robot with mounted plow.

The robot is driven by two 24 volt electric motors that each output 1.5 horsepower through a 20:1 reduction gearbox^[4]. Four spring-loaded casters mounted on the four corners of the casing were inherited from the previous RedBlade lawn mower platform, but testing showed that these caster wheels were problematic when the robot encountered small ledges, tree roots, and other terrestrial obstacles. Our research led us to the compromised, minimal modification solution of replacing the front two caster wheels with a single, more robust mount and wheel attachment in the front-center of the robot. The rear two spring-loaded casters were retained. Field tests indicate that this change afforded the robot increased mobility, but many soft terrain surfaces (mulch, depressions, etc.) continue to hinder performance.

The original drive wheels proved to provide too little traction in some dry and all wet conditions. These wheels were replaced with auger-style wheels which can be seen

in Figure 6. Increased traction was proven through experimental testing.



Figure 6. New auger-style wheels.

3. ELECTRICAL COMPONENTS

The RedBlade platform houses a number of electrical and electronics components including batteries, safety switching circuits, a motor controller, and the entire navigation sensor suite. This section presents details of the electrical system.

3.1. Power Supplies

Three banks of lead-acid batteries provide power to different sections of the system. Two 24V banks provide power to the drive wheels and blade mower respectively, while a third 12V bank powers our computer and other sensor components which do not have an onboard battery.

To prevent voltage arcs between the chassis and the battery leads, a piece of non-conductive plastic was placed between the battery housing and the controller, which was the nearest conductive component. This arcing was too common when changing batteries and was extremely dangerous. The addition of the non-conductive plastic eliminated these chance arcing events.

Early experiments showed that some of the electronics operated erroneously when the supply voltage fell below a certain threshold. An additional set of batteries was purchased to facilitate prolonged testing and run times.

3.2. Processors, Controllers, and Hard Drives

The main system control is operated by a PC running a Linux installation. We communicate with this device via direct connection or through an on-board wireless router.

Because RedBlade also functions as a snow plow, weather-proofing was required to ensure safe and reliable operation. A standard hard drive contains small, weak motors which are likely to freeze in lower temperatures. RedBlade uses a solid-state device (SSD) to mitigate this risk. In addition to having better temperature endurance, the SSD is known for being able to withstand much higher degrees of vibration and impact compared to that

of a physical disk. Power consumption is also greatly reduced. The typical SSD consumes no more than 1.7 watts while idle, with many as low as 0.5 Watts^[8]. This is much lower than our previous hard drive which consumed up to 20 watts^[9] to spin up from idle.

A RoboteQ AX2550 controller is used to drive the motors^[10]. This controller is capable of directing 6 times the power that we require and has several built-in protection modes. We limit the controller's output to 20Amps in order to protect the 14 gauge wire that is used for power transfer to the motors.

3.3. Safety System

The RedBlade platform has two emergency shut-off options: remote control and an on-board stop button. Our tests show that the stopping distance from a maximum speed of 2 m/s varies depending on the type of surface tested:

- Icy Surface: 0.5m
- Concrete: 0.3m
- Rough Brick: 0.2m
- Dry Grass: 0.3m
- Wet Grass: 0.4m

3.4. Navigation Sensors

A MicroStrain 3DM-GX2 IMU is used to determine the vehicle heading^[11]. It has an adjustable data rate to facilitate interfacing with different clients. Our tests in an environment where there were no major ferromagnetic materials present show that we can achieve approximately 2° accuracy over a 3 meter distance. Calibration for the device can be completed using the Hard Iron Calibration tool in MicroStrain's Data Acquisition and Display software^[12]. While this particular guide is for the older GX1 model, the process is similar. On site calibration is necessary to mitigate errors from the IMU while the robot is in the field of play.

The Hiper Lite Plus is a survey grade dual-frequency differential GPS system by Topcon. Field tests near Miami's Engineering Building with masking angle at 30° on one side shows location accuracy within 2cm as specified by the device manufacturer^[13]. The raw geodetic coordinates given by the Hiper Lite Plus receiver are converted to an ENU local coordinates system before being sent to the control algorithm. The origin of the local coordinate system is the beginning of the path (where the robot is to begin its job), while the robot's initial heading points to the local y-axis. Custom driver software was written to allow the GPS receiver to send its position measurements to the on-board processor.

Two US Digital E7MS quadrature optical encoders are installed on both left and right wheels of the vehicle. Each encoder sends its signal on two different channels with 90 degree offset. By using two channels it is

possible to determine the direction of movement if there is no slippage. When the robot is moving forward, one channel emits a pulse before the other. By counting the pulses sent from each encoder, we are able to determine the number of revolutions, therefore allowing us to determine the distance traveled.

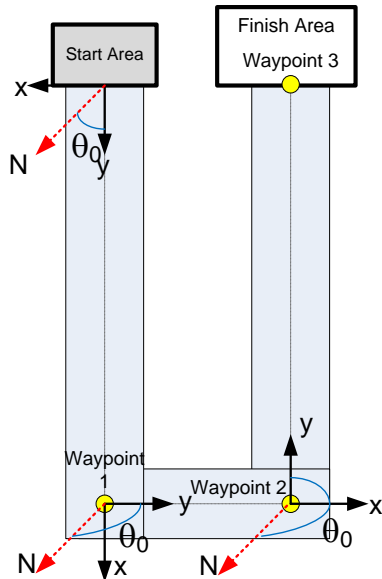
In order to assign a control constant to the wheel encoders we must first be able to determine how much drift the encoders have relative to each other. This is accomplished by comparing the output from each encoder after turning each wheel an equal distance. Comparison between each wheel output allows us to assign a control constant based on how much drift is observed.

The wheels encoders are dead-reckoning devices that will not provide as precise a position estimate as the DGPS unit. However, when combined with GPS measurements, these encoders will provide measurements to detect slippage and mitigate.

4. CONTROL ALGORITHM

A PID-based feedback control algorithm is used in both RedBlade configurations. The algorithm accepts a waypoint vector as its inputs. We designate the end of each straight path as a waypoint. For example, in this “U” shaped path (encountered in the plow competition), there are three waypoints as shown in Figure 7. We will use this “U” shaped path as an example while explaining our control algorithm.

Figure 7: Shows the three waypoints and the local coordinate



system associated with each path segment

We divide the “U” shaped path into 3 segments:

- Segment 1: starts at the start, ends at waypoint 1.
- Segment 2: starts at waypoint 1, ends at waypoint 2.
- Segment 3: starts at waypoint 2, ends at waypoint 3.

For each segment, we define a local coordinate system whose origin is located at the starting point of the segment and the local y-axis is along the path direction, while the x-axis is perpendicular to the y-axis as shown in Figure 7. The waypoint vector for this “U” shaped path is defined as:

$W = \{[\theta_0(1), L(1)]; [\theta_0(2), L(2)]; [\theta_0(3), L(3)]\}$, where θ_0 is the path direction relative to true north, and L is the path length. At each segment of the path, the vehicle’s objective is to reach the point $[0, L]$ while trying to maintain a heading direction of θ_0 . Clearly, this W vector is expandable if a more complicated path is required.

Figure 8 explains how this objective is achieved. While traveling along a segment of the path at time t , the GPS input shows that the vehicle is located at some intermediate point (x, y) . We can compute the desired vehicle heading based on the current location (x, y) and the end point $(0, L)$:

$$\theta(t) = \theta_0 + \Delta\theta(t) \tag{1}$$

$$\Delta\theta(t) = \tan^{-1} \frac{x}{L - y} \tag{2}$$

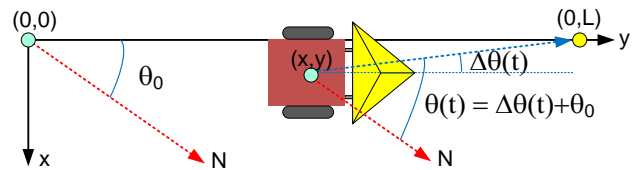


Figure 8: Schematics explaining the relative vehicle heading dependence on vehicle location and target location.

The $\theta(t)$ value is the desired set point value of the vehicle heading in order for the vehicle to reach the desired destination. The IMU measures the actual vehicle heading. The difference between the computed set point and the IMU measurement is the error input to the PID loop. By properly selecting the K_p , K_i , and K_d coefficients, the PID loop generates a signal that drives the two wheel motors in a way that minimizes the error, thereby forcing the vehicle stay on the path. Figure 9 shows the block diagram of the PID feedback loop.

The above algorithm applies to each of the three segments defined earlier. Therefore, the entire procedure is repeated for each waypoint vector component. Figure 10 shows the top level block diagram that cycles through each of the waypoints and executes the PID for each path segment. Any number of path segments can exist in the

field of play, so the path planning can be abstracted to allow navigation around more complicated paths.

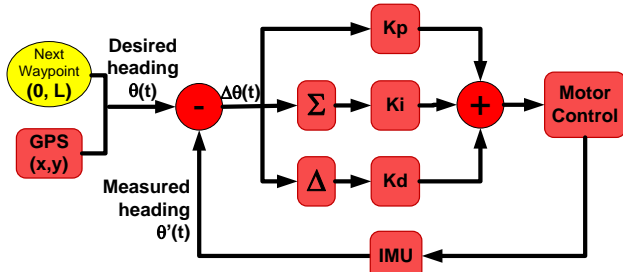


Figure 9: PID feedback control block diagram.

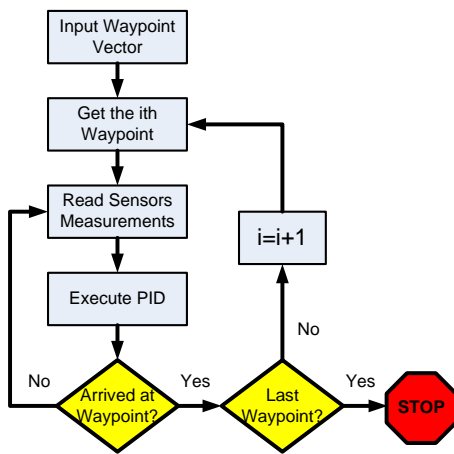


Figure 10: Block diagram of vehicle operation over the entire path.

System stability and response is dependent on the selection of the aforementioned control constants. We experimentally derived these constants through the use of the Ziegler-Nichols tuning method. This method requires that the K_P constant be set such that the system is put into sustained oscillation without becoming unstable. Herein, this maximum K_P will be referred to as K_U . The period of these sustained oscillations is defined as T_U . The three control constants are derived from the equations listed below:

$$K_P = 0.6 \cdot K_U \quad (3)$$

$$K_I = \frac{2 \cdot K_P}{T_U} \quad (4)$$

$$K_D = \frac{K_P T_U}{8} \quad (5)$$

Figure 11 shows the data plot of IMU heading versus time that was used in deriving our control constants. The Fourier transform of this data set gave the data shown in Figure 12. The same plot is visible in Figure 13 with only the area of interest shown.

It should be noted that there are two peaks in Figure 13 that suggest two possible values of T_U . The value for T_U gotten from each peak was experimentally tested and we determined that the most effective control constants were derived from $T_U = 2.14$ seconds.

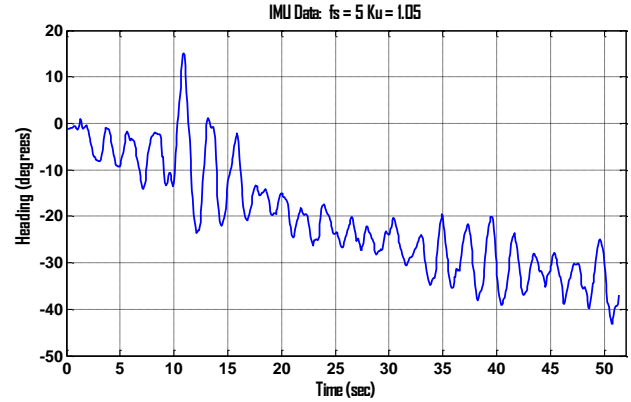


Figure 11: IMU heading versus time. This data was used to empirically derive the control constants.

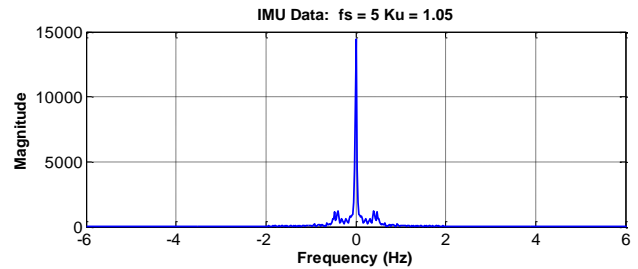


Figure 12: Fourier transform plot of Figure 11 data set.

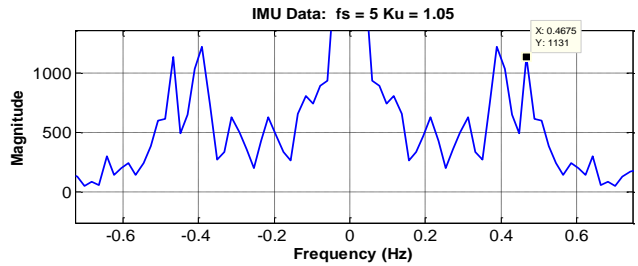


Figure 13: Peaks of interest from the Figure 12 plot. Notice the two peaks that suggest two possible values of T_U .

5. PATH PLANNING

Our current path planning strategy consists of these steps:

- Mow through zone 1 towards zone 2
- Completely mow zone 2
- Continue mow zone 1 while avoiding flower bed

As stated in the rules of the competition, teams may only start from safety buffer zone. Referring to Figure 2, our plan is to have the robot start from top right of the buffer

zone and enter zone 1 from the top right corner. Then, the robot will move straight all the way down to zone 2.

Given that zone 2 provides higher score relative to other zones, we set zone 2 as our priority to mow as early as possible.

The last part of our run that demand the most amount of time is completing the zone 1 with the presence of the flower bed. As mentioned in the later section that we may not have our LIDAR to be able to detect the flower bed, we approach this issue by mapping the location of the flower bed within zone 1. As stated in the rules, there will be a minimum of 2m between the edges of flower bed and edges of zone 1. In addition that the dimension of the flower bed will not exceed 5m, it is then possible to estimate the location of the flower bed in our path planning strategy.

6. OBSTACLE DETECTION AND AVOIDANCE

As stated in the rules for the ION Autonomous Lawnmower Competition [3], the advanced category will have several obstacles to avoid: a fence, a flower bed, and a stuffed poodle mounted on an R/C car. The first two fall into the category of static obstacle, and have the goal of mowing around them, but not touching/moving them. The last obstacle falls into the category of dynamic obstacle, with the goal being simply to avoid it.

The current rules state that the dynamic obstacle will not be brought out until the mower is at least 2 meters away from another obstacle. Since the location of the fence obstacles is acquired when mapping the field, we know exactly where they are. The flower bed, however, will not be placed until 5 minutes before the run begins, and must be mapped during the run. If we can find and map the flower bed immediately, then we know where an obstacle will be detected, and if there is a hit somewhere else on the map, then that must be the dynamic obstacle.

6.1 Mapping the flower bed

Mapping the flower bed requires a proximity sensor of some sort, and there are many ideas currently in testing. LIDAR offers the advantage of a 180 degree scan, but the materials used may not reflect the infrared light used well enough for our purposes. LIDAR is also very sensitive to the noise caused by grass sticking up in front of it. Ultrasonic proximity sensing is also affected by grass, but not as much as LIDAR, since the grass would probably not reflect the sound well enough to be detected as a hit. Ultrasonic also offers the advantage of being the only

other method of mapping which does not require the mower to touch the flower bed.

More “hands-on” mapping could be done with whisker sensors, or spring loaded potentiometers. While a typical whisker sensor provides only an on/off output, there are flex sensors which provide an analogue output with respect to how much they are bent. As the mower gets closer to the flower bed, the flex sensor will be bent further, indicating in a predicable manner how far away the mower is. The downside to a flex sensor is the possibility of noise being introduced through contact with the grass, or bumps in the lawn as the mower is moving.

Similar to the flex sensor is the membrane potentiometer. This sensor is a long, flexible strip, similar to the flex sensor in shape, however it requires pressure at a certain location to register a reading. This scales linearly with respect to where pressure is applied. Since the sensor can be bent or touched lightly without detecting a hit, noise from moving/impacting grass would be reduced. Unfortunately, it is unclear whether the sensor will be sensitive enough to detect when it has contacted the flower bed.

The final thought on mapping the flower bed is an arm/skirt under the mower which is spring loaded and attached to a potentiometer. By adjusting the spring, optimum stiffness can be obtained so as not to falsely detect grass, and not move the flower bed while mapping. The problem here is where to put the rather large device so it won't interfere with the blade or wheels of the mower.

We are currently performing extensive testing and evaluations of these sensors. Our final selection and implementation will be based on the evaluation outcome.

6.2 Detecting dynamic obstacles

The dynamic obstacle (stuffed dog) in the competition will enter the field perpendicular to the mower's velocity vector while the mower is moving straight, and at least two meters away from another, static obstacle. This solves the problem of needing to detect the dynamic obstacle while also detecting a static obstacle, and gives a narrow window to need to detect the dynamic obstacle.

In order to stop in time to not hit the dynamic obstacle, non-contact detection, such as LIDAR or ultrasonic needs to be used. The advantage to ultrasonic detection is the simplicity of the data processing which needs to happen. With LIDAR, there are 361 measurements to process, while with ultrasonic, there is only one. Unfortunately,

the background that the ultrasonic sensor picks up is noisy, and somewhat hard to deal with. Since the stuffed dog is made of a soft, non-uniform fabric, the ultrasonic sensor may have trouble detecting it as well.

LIDAR solves the problem of background noise, since there are more than one reading to look at. We would be able to filter the noise out by looking at measurements taken around a possible false positive. However, the LIDAR also may not be able to reliably detect the fur of the dog, leading to the dog getting hit.

Ultimately, if space and processing power permit, it would be ideal to use multiple sensors for each type of obstacle, to be certain of where the obstacle is, or to have a backup in case another sensor is unable to perform on competition day. As with the IMU, tachometer and GPS, these different sensors would complement each other, and could all factor into an all-encompassing proximity sensor driver on the software side. We have ordered the same poodle dog used in the competition and will perform extensive testing using both Lidar and ultrasonic sensors.

7. FAILURE MODES AND RECOVERY ACTIONS

Two failure modes are identified. The first failure mode is current overload. Though the motor control limits steady state current to 20A, it is possible that current can surge to as high as 80A. This is due to the inductive nature of the motors. This problem is handled by a 20A circuit breaker which interrupts the motor power supply when the overload occurs.

The second issue is speed control. The competition regulations dictate a maximum speed of 10m/s. This has been accomplished by a separate thread that constantly polls the vehicle velocity through both GPS and wheel encoders. This thread interrupts motor operation if the system is in excess of the maximum speed.

8. TESTING AND COMPETITION SIMULATION

Vehicle testing was divided into three stages:

Stage 1: Indoor testing of PID control using IMU input only. Testing period: October-November 2010. During this time, the objective was to fine tune the PID control and to obtain a feeling for the tuning direction. Short segments of a straight path inside the Miami University Engineering Building were used to perform the test.

Stage 2: Outdoor testing using both IMU and GPS inputs. Testing period: December 2010. During this period, students evaluated GPS accuracy, learned to convert GPS solutions to local coordinates, tested driver functions on

the processor on the platform, and tested the vehicle to travel a straight line over longer distances.

Stage 3: Outdoor testing under snowy conditions using IMU, GPS, and wheel encoder inputs. Testing period: January 2011. During this period, we had several major snow storms which provided natural condition for our testing. During the testing, we uncovered major vehicle platform problems. The plow often got stuck in the snow and vehicle slippage occurred. We modified the plow mount, added weights to the vehicle, and updated the control algorithm to include slippage detection and handling. Additionally, we tested the vehicles ability to plow a "U" shaped path.

Stage 4: PID tuning on grass and simulator conception. Testing period: February 2011. During this period, the software was refactored and the PID constants were tuned to operate on grass. During this period, work began on a RedBlade simulator tool.

Stage 5: Advanced path planning LIDAR troubleshooting. Testing period: March 2011. During this period, path planning was designed for much more advanced course in the lawnmower competition. LIDAR driver was found to be more troublesome than expected.

Stage 6: Path planning continues and simulator is realized. Testing period: April 2011. During this period, path planning for the lawnmower functionality was refined.

Stage 7: Obstacle avoidance. Testing period: May 2011. During this period, a variety of touch sensors, ultrasonic sensors, and laser are tested and evaluated for both static and dynamic obstacle detection.

9. CONCLUSIONS AND FUTURE WORK

During the past eight months, a group of undergraduate students at Miami University designed, built, implemented, and tested an autonomous vehicle which can plow snow in the winter and mow grass in the summer. This vehicle has demonstrated the ability to move in a straight line similar to the distance specified by the 1st ASC rule and can make turns as required by the competition. This ability has been achieved through a navigation sensor suite, including a survey grade DGPS receiver, a MEMS IMU, and wheel encoders, a PID-based control algorithm, and a home-made mechanical platform. Several failure modes have been taken into consideration and recovery actions have been implemented after more extensive testing. Extensive obstacle detection sensors including ultrasound, laser, and touch sensors are currently under investigation.

The more long-term impact of this project is the valuable learning experience gained by the students working on the

team. Students learned trouble shooting, managing deadlines under a tight schedule, and interfacing with parts and supply sources. They also learned organizational skills through this complicated project and improved their communication capability by preparing for the presentation and the written report.

ACKNOWLEDGMENTS

The RedBlade team would like to thank the Institute of Navigation Satellite Division for sponsoring the autonomous lawn mower competition and for the ION Dayton Section for organizing the competition. The team received the funding support from Miami University Office for Advancement of Research and Scholarship, School of Engineering and Applied Science Dean's Office, the Department of Electrical and Computer Engineering, and the Department of Computer Science and Systems Analysis. Additionally, the team appreciates technical guidance and support from Drs. Wouter Pelgrum and Frank van Graas of Ohio University and Mr. Jeff Peterson from Miami University.

REFERENCES

- [1] McNally, B., M. Stutzman, C. Korando, J. Macasek, C. Mantz, S. Miller, Y. Morton, S. Campbell, J. Leonard, "The Miami Red Blade: An Autonomous Lawn Mower," *Proc. 2004 ION Annual Meeting*, P538-542, Dayton, OH, Jun. 2004.
- [2] <http://www.automow.com>
- [3] <http://www.ion.org/satdiv/alc/rules2011.pdf>
- [4] Newstadt, G., K. Green, D. Anderson, M. Lang, Y. Morton, and J. McCollum, "Miami RedBlade III: A GPS-aided autonomous lawnmower," *J. Global Positioning Systems*, 7, No.2, P115-124, 2008.
- [5] http://www.engineeringtoolbox.com/friction-coefficients-d_778.html
- [6] Casassa, G., Narita, H., & Maeno, N., "Shear cell experiments of snow and ice friction. *Journal of Applied Physics*," 69(6), p3745, 1991.
- [7] Kuroiwa, D., "The kinetic friction on snow and ice," *Journal of Glaciology*, 19(81), p141-152, 1977.
- [8] http://www.kingston.com/ukroot/ssd/v_series.asp
- [9] http://www.pcpower.com/technology/power_usage
- [10] Roboteq, Inc., "AX2550 AX2850 Dial Channel High Power Digital Motor Controller User's Manual". Roboteq. 2007.
- [11] <http://www.microstrain.com/3dm-gx2.aspx>
- [12] <http://www.microstrain.com/pdf/3DM-GX1%20Hard%20Iron%20Calibration.pdf>
- [13] <http://www.topconpositioning.com/products/gps/geodetic-receivers/integrated/hiper-lite-plus.html>