IGVC
2009

UNIVERSITY OF
Waterloo

UW
ROBOTICS TEAM

"IOREK" DESIGN OVERVIEW

I, Dr. William Melek, assistant professor in the Department of Mechanical and Mechatronics Engineering at the University of Waterloo, certify that the engineering design presented in this document for the Iorek robot by the current student team has been significant and equivalent to what might be awarded credit in a senior design course.

William W. Melek

Prof. William Melek <wmelek@mecheng1.uwaterloo.ca>

14 May 2009

1. Introduction

The University of Waterloo Robotics Team (UWRT) presents Iorek, a new autonomous vehicle designed and built entirely by undergraduate students. UWRT is proud to make its second appearance at the Intelligent Ground Vehicle Competition in 2009.

2. Innovation

A large portion of the innovation found in this vehicle resides in the design of the control architecture. Aside from taking the prudent step of separating the low-level motor controls from the high-level board via a custom embedded system, several other measures have been put in place to safely degrade the vehicle performance when problems occur. Key software modules on both the high-level board and the low-level controller implement timeouts and checksums to ensure that regular commands are being exchanged as expected. If failures are detected, the vehicle will come to a controlled stop. Hardware drivers are implemented in dedicated threads and the entire system transfers information in an event-based fashion, isolating the various software modules from each other.

Furthermore, the software itself is designed with the vehicle hardware in mind. Wheel acceleration profiles are generated dynamically whenever the vehicle maneuvers, lessening stress on mechanical components and improving the controllability of the vehicle. Parameters ranging from wheelbase to the system's state estimation method can be changed in the field without the need to recompile the code.

There are a number of innovative mechanical design aspects of this chassis, most notably a dropped axle and a nodding LIDAR mount. With six wheels, a concern for this chassis is the drag that occurs while turning. In order to counter this, the center axle is positioned slightly below the front and rear axles. Theoretically, this results in turns being made on four wheels instead of six, thus decreasing the drag acting on the robot.

A LIDAR is positioned at the front of our chassis in order to detect obstacles. In order to increase the efficiency of the LIDAR, the mount on the front of the chassis frame is designed to nod up and down continuously while the chassis is in motion. This enables the LIDAR to determine both the location and vertical profile of obstacles.

3. Vehicle Overview

Iorek is built to be extremely durable as well as versatile. Thanks to UWRT's corporate sponsors, high quality components were purchased for significantly less than MSRP. A detailed breakdown of cost and a look at Iorek's hardware specifications follow.

3.1 Components and Cost

Item	MSRP	Cost to UWRT
NovAtel OEMV-3 GPS System	$10000	$2000
SICK PLS101-112 LIDAR	$6000	$0
Logitech QuickCam Pro 9000	$100	$100
EZ-Compass-3	$1000	$0
Trojan SCS-150 Deep Cycle Lead-Acid Battery (x2)	$360	$310
IFI Victor 885 Motor Controllers (x2)	$400	$360
Mini-ITX SBC86807 Motherboard	$250	$0
Sam-7 Header Board	$50	$50
GreyHill 63R Encoders (x2)	$110	$110
12V Mini bike motors (x4)	$800	$0
Globe motor	$25	$0
Samlex America SDC-23 Step-down	$150	$150
Pololu Motor driver	$50	$50
PicoPSU power supply	$70	$70
Various Mechanical Components	$690	$690
Various Electrical Components	$200	$180
Total	**$23855**	**$4070**

3.2. Specifications

Physical Properties		**Performance**	
Weight	90 kg	**Speed**	2.1 m/s
Height	1.0 m	**Acceleration**	5.25 m/s^2
Width	0.75 m	**Turning Radius**	0 deg
Length	1.0 m	**Payload**	10 kg

Power Consumption		**Sensor Accuracy**	
Drive Motors	40 A	**GPS**	±0.6 m
LIDAR Motor	5 A	**Encoders**	0.703 deg
Sensors	2 A	**LIDAR**	± 50 mm @ 50 m
Electronics	2 A	**Compass**	± 0.25 deg
Total Current Draw	**49 A**	**Webcam**	30 fps
Battery Capacity	74 AH @ 50 A		
Battery Life	approx. 1.5 hrs		

4. Mechanical Design

4.1 Overview of Design

Many criteria were considered in the design process. The first was the ability to navigate effectively off-road. To do this for the IGVC and further applications that would potentially require more demanding off-road capabilities, large pneumatic tires were selected. These are far superior to treads in durability, ease of use and, in most cases, traction. To facilitate excellent maneuverability six wheels are used, with the middle wheel lowered by $\frac{1}{4}$". This allows the center wheel to take the majority of the normal force, and therefore, the frictional force, which makes turning easier, since there is less friction (turning scrub) force to overcome while turning. It also keeps the wheel base long for climbing and stability, without adding extra turning scrub associated with a long base, four wheel skid steer. Most components were selected based on their cost and availability. The machined components are designed to be quick to fabricate and easy to assemble to speed up the manufacturing. Side and isometric views of Iorek are shown in Figures 1 and 2.

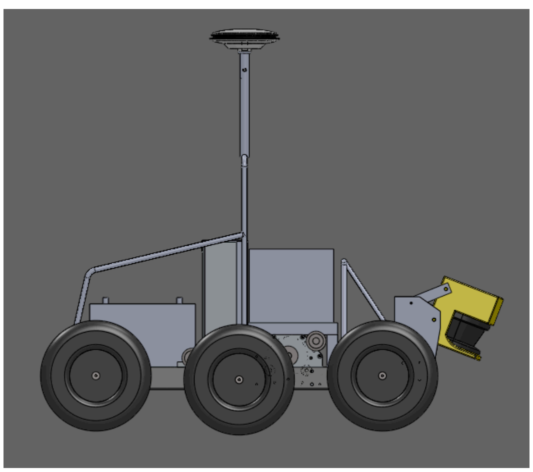

Figure 1: Side View

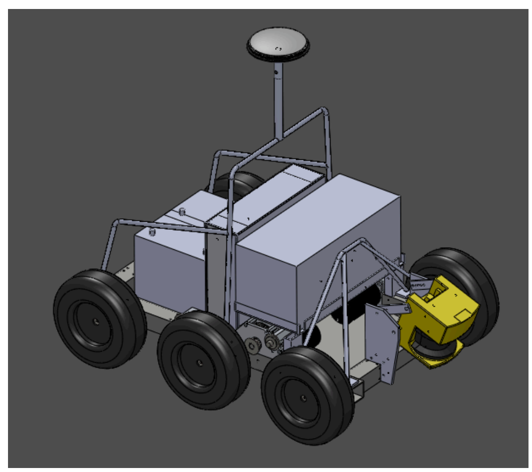

Figure 2: Isometric View

4.2 Drive Train

The drive motors are 12V minibike motors that can provide 270 Watts at maximum power. Since the possible weight of the vehicle is greater than 160 pounds, four motors are used: two powering each side. These will have their power combined in a custom single stage gearbox which uses a 15 tooth, 1.5 module gear on the output shaft of each motor, mated to a common output shaft with a 60 tooth gear (Figure 4). The output of the gearbox is a 14 tooth sprocket, which drives one continuous chain, as shown in Figure 3, that wraps around the 60 tooth sprocket on each of the 3 wheels on that side. This layout allows for a single chain and two tensioners. The combination of the 4:1 Gear reduction and the 60:14 Sprocket reduction combine for a reduction from the motors of 17.14:1. This ratio, with the 2700 RPM motors and 10.5" wheels gives a top speed of 2.1 m/s (4.7 mph), which is just under the maximum for the IGVC competition.

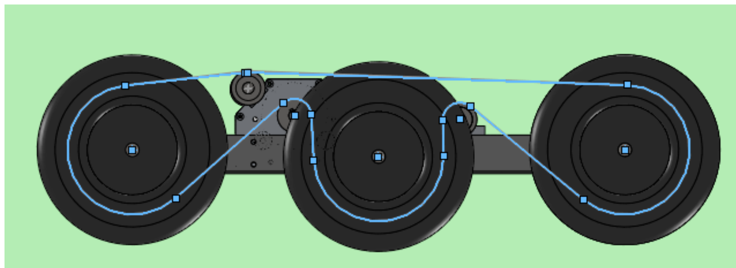

Figure 3: Side View of Chain and Centre Wheel Offset

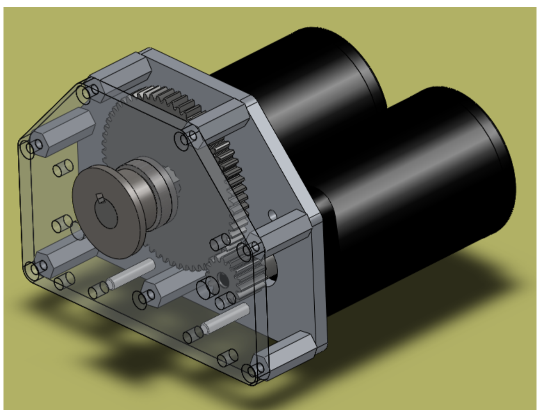

Figure 4: Gearbox Design

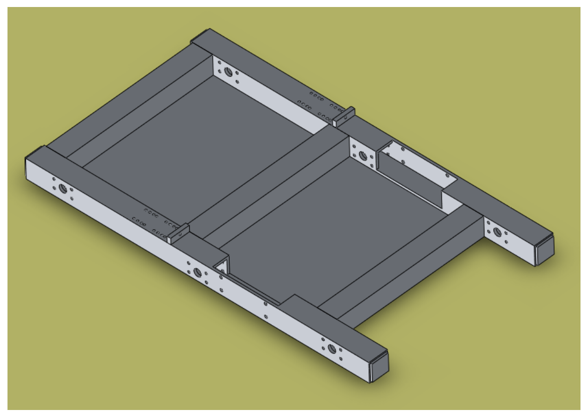

Figure 5: Frame Design

4.3 Frame

The frame is a ladder frame for maximum strength, minimum cost and ease of fabrication. There are two main rails that run the length of the frame and mount most of the components: the dead axles for the wheels, and the gearboxes. These rails are subjected to large amounts of torsion, due to the cantilevered axles, so a cross rail is situated as close as possible to each axle. This layout provides maximum torsional stiffness, especially since boxed tubing is being used instead of channel or angle (Figure 5). The frame is welded together with solid fillet welds and the bottom plate is stitch-welded on. The bottom plate provides additional stiffness, and convenient mounting for the remaining components. The batteries need to be isolated from the vibration from the frame and, as such, sit on a foam pad on the bottom plate with a strap to hold them in place.

Quarter-inch tubes were used for the walls. The reasons for this are that it adds only minimal weight (approx. 8 lbs), and increases the strength greatly, especially at welds. However, the biggest advantage is that the tube can safely be tapped for mounting components.

4.4 Axle Design

The wheel and sprocket assemblies have integrated $\frac{3}{4}$" bearings so they are mounted on dead axles. The axles must simply support the load created by the weight of the vehicle and the tension of the chain. It was determined that stress concentrations on the axles would be quite large and so a hardened 4130 shaft is used. The axles were shrink-fitted into a mounting plate and bolted onto the frame through the close fitting holes milled in the frame. This design is very simple to machine (using undersized reamers) and is very durable.

4.5 Electrical Box Design

The electrical box was designed to hold the boards vertically, which is beneficial for cooling, and to allow ease of access for troubleshooting. The box is made from aluminum which creates a Faraday cage to shield the electronics from electromagnetic interference generated by the high-current electronics. Rubber grommets are used on the fastening holes to reduce vibration.

4.6 LIDAR Mount

In order to increase the amount of area scanned by the LIDAR, it was determined that the sensor's mount should continuously pivot up and down while the vehicle is in motion. In order to achieve this "nodding" motion, a four-bar mechanism is used in a crank rocker formation. The rotational motion of one bar powered by a motor is converted into the rocking motion of the pivoting mount, which ultimately results in a more efficient use of the LIDAR.

5. Sensors

5.1 Positioning Sensors

Encoders

Both drive axles on Iorek have an encoder mounted on them so that the distance travelled by each side can be determined. This can be used to not only determine speed and distance travelled but also rotation of the robot by calculating the difference between the speeds of each side.

Working together with the GPS and compass, the encoders will help determine position as well as heading. Greyhill 63R encoders have two channels at 128 pulses per revolution. Therefore, using quadrature decoding each axle's direction and rotation is known with resolution 360/(128x4) = 0.703^{o}. Given the drive wheel diameter of 25.4cm the encoders are able to detect the robots change in position with (π x 25.4 x 0.703)/360 = 0.156 cm resolution.

Differential GPS

Differential GPS (dGPS) is the only sensor that can localize the robot in the global reference frame. However, dGPS is the least accurate of all the positioning sensors. Therefore, the other sensors are used to augment the dGPS readings. The UW Robotics Team purchased a NovAtel OEMV-3 system and its associated antennas. The package is meant to be used as a pair – one stationary and one mobile. However, given the IGVC rules that prohibit off-board sensors, Iorek is limited to using the available WAAS signals for correction and can therefore only achieve around 0.6 m absolute accuracy [1].

Compass

A compass is included in Iorek's sensor suite to give a reference heading in the global reference frame. The GPS can also be used to determine the robots heading but the robot needs to be moving in order to do this. The compass helps determine heading more accurately and without the need to move the robot. The EZ-COMPASS-3 was chosen for its robustness and integrated tilt sensor. The accuracy on this compass is stated to be +/- 0.25 deg [2].

5.2 Obstacle Detection Sensors

Laser Range Finder

The laser range finder is the primary sensor for obstacle avoidance; it is used detect any and all obstacles with height. The SICK PLS is an excellent sensor for mobile robotics due to its excellent performance benchmarks and durability. The SICK sensor has a 180° view and a maximum range of 50 meters. It measures with 5 centimeter accuracy and angular resolution of 0.5 degrees. This allows the robot to detect and map obstacles well in advance of encountering them.

Webcams

Iorek uses a Logitech QuickCam Pro 9000 webcam for its vision system. The webcam is mounted on the tower above the robot in center and angled downed in such a way that the top edge of the image is parallel with the horizon. The camera is used to detect non material obstacles such as lines and simulated pot holes.

6. Electronics Design

6.1 Power Distribution

Iorek is powered by two 12V Trojan deep cycle batteries, similar in size to car batteries, which are wired in series to produce a 24V supply for the robot. These batteries were chosen because of their very high capacity which is helpful for testing. The drive motors are driven directly off the 24 V line through a pair of Victor 884 motor controllers. There is also a 12V Samlex America step-down converter which is used to power the nodding LIDAR motor. The mini ITX mother board is powered through a PicoPSU power supply and the other electronics are supplied with 5V via a small switching regulator.

6.2 ITX Board

The main control loop, sensor data and path-planning is done on a VIA Mini-ITX computer. The ITX board has a 1.0GHz processor, 1GB DDR2 RAM, a laptop hard drive, and the standard set of peripheral ports. The operating system used is the Ubuntu 8.10 server edition. The LIDAR, GPS and compass are connected to the ITX board through USB to serial converters and a USB hub. The board processes their data and makes movement decisions. The decisions are then passed on to the MCU board using a serial port and custom binary protocol.

6.3 MCU Board

The main control board (MCU) is the robot's low-level control board. The board receives speed commands from the ITX board and uses encoder feedback to control the motor speeds. The MCU sets the speed of each motor by sending a servo pulse to the Victor 884 motor controller. In addition to controlling the robot's main motors, the MCU controls the LIDAR nodding and monitors the E-stop and the LIDAR potentiometer status. The MCU board is designed to have the following functionality:

The MCU board uses an Atmel SAM7 microprocessor (AT91SAM7S64). To simplify the board design and rewiring, an Olimex SAM7 header board is used. The header board has self-contained power management. Instead of soldering directly to the SAM7 microprocessor, the header board connects to two rows of female headers. This makes prototyping and fabrication easier since wiring a pair of headers is much easier than wiring to a 64 pin LQFP package. The header board provides access to most of the SAM7's programmable I/O pins.

7.0 Emergency Stop

All motor power on the robot is routed through relays which are used to cut power when the emergency stop is pressed. There is a physical E-stop which is a large red button located on the rear of the robot as well as a remote e-stop. The remote e-stop is a remote control which signals the MCU to switch off the relays. Both the physical and remote e-stops are in series so either will result in stopping all power to the motors.

8. Software Design

8.1 Architecture

Key in the design of the system is the separation of the high-level state estimation, mapping, and behavior control from the low-level motor control (Section 6.0). This separation allows for faster development, easier debugging, and a greater degree of safety and predictability. The majority of the vehicle's control is done with an application developed with Java 1.6. OOP principles are used extensively, allowing the system to be reconfigured at runtime. Additionally, sensor input is incorporated in an event-based fashion, reducing interdependence between sensor modules.

The vision processing code is written using the open-source OpenCV vision libraries and communicates with the high-level code via TCP/IP sockets. For this system, both programs run on the same computer. However, the use of sockets makes it very straightforward to dedicate a second computer purely to vision processing without needing to change any code. The low-level control code is written in C for an ATMEL SAM7 microcontroller. The open-source FreeRTOS real-time operating system is used to provide rudimentary threading, message passing, and memory allocation. Figure 6 shows a block diagram of the computer architecture, including peripherals.

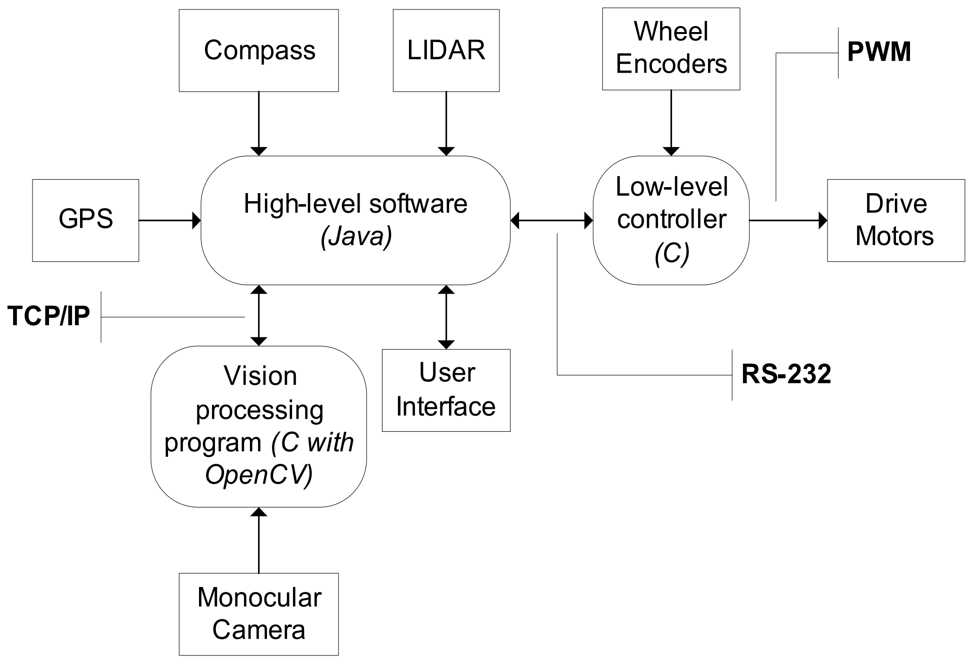

Figure 6: System Architecture

8.2 Verification

The event-based architecture makes the verification of high-level system operation a straightforward task. A simulation environment was developed to provide sensor events and receive motor commands. Because complicated tasks such as localization and trajectory generation are completely dependent on sensor events, they do not need to be modified to undergo these tests. This significantly reduces the time required to move the system from successful operation in the simulator to successful live operation.

The low-level board can also be controlled via a remote control receiver, allowing the electrical functionality to be tested without the high-level software. Not only is this convenient for testing when the high-level system is unavailable, it allows the source of any errors to be quickly isolated.

More rigorous performance verification is made possible by use of the Apache log4j framework. The system is set up to continuously log all sensor events, state estimates, and outputs to a time- and date-stamped log file as they are received. A set of MATLAB files can read this output, allowing for a significant degree of analysis and comparison to be done after each run.

8.3 *Safety & Monitoring*

In addition to the hardware emergency stop systems detailed in Section 6.4, several design decisions were made to increase the reliability of the control systems themselves. First, the high-level software logs system events as they occur, in a similar way to the method used for performance evaluation. Also inherent in the architecture is the safety provided by the event-based architecture. Since each sensor driver is its own thread, failures are isolated to individual threads - the system can keep operating based on its remaining data. On top of this, the high-level kinematics module contains a watchdog timer to ensure that it is continuously being provided with inputs. If this timer expires, the module assumes that a critical error has occurred at a higher level and issues a stop command to the hardware.

This is mirrored on the low-level board. This board requires new commands to be sent at a frequency of at least 1 Hz. If this is not met, it overrides any existing commands and forces the robot to a stop. The serial protocol between the boards implements a CRC-8 checksum, and any commands with an invalid checksum are discarded.

9. Vehicle Strategy

9.1 *Environment Representation*

Vehicle Position

The vehicle position is represented by its global x position, global y position, and global heading. The x dimension is analogous to longitude, and the y position is similar to latitude. Both are measured in meters, with the origin at the intersection of the equator and the prime meridian. Since the vehicle is not traversing long distances, the inaccuracy introduced by assuming the earth is flat will not be an issue. Global heading is laid out in radians according to an RHS coordinate system, with 0 radians corresponding to the vehicle facing east and $\pi/2$ radians meaning north.

Obstacles

The LIDAR maps obstacles into a discrete occupancy grid with a resolution of 10 cm. Each grid cell stores an estimate of how likely it is that the cell is occupied. The usage of a discrete grid makes path planning around obstacles very straightforward task, since the cells can be used as nodes for a graph search.

Boundaries

To accomplish lane following, the OpenCV-based vision system receives real-time video data from the camera. Frames are first processed to enhance the contrast between the white boundary lines and the rest of the course. One approach that works reasonably well uses only the saturation channel of the image for further processing. A threshold is then applied to equalize all areas of the image that are of no concern to the lane following algorithm. Finally, a Hough transform (provided by OpenCV) finds possible lines in the image, and dominant lines likely to be course boundaries are selected from this set.

The boundaries of the Navigation Challenge path are stored as a list of 2-D line segments projected onto the ground plane. To simplify representation and reduce error, the ground is assumed to be flat. As well, only the lines currently visible are used in the decision making process. This removes the need to implement an extensive representation of the path boundaries.

9.2 State Estimation

Though accurate vehicle positioning is not required for the navigation challenge, it is absolutely essential for the navigation challenge. To this end, an extended Kalman Filter (EKF [3]) is used to estimate the position portion of the vehicle state. The drivewheel encoders are used as the control input in the prediction step, while a combination of the GPS and compass data is used for the correction step. Simple differential drive kinematics [4] are used for the control model.

9.3 Initial Trajectory Generation

Initial trajectories are generated simply by offsetting towards the middle of the course any lines picked up by the vision system. Where no lines are visible, the vehicle continues straight ahead.

9.4 Obstacle Avoidance

Once the initial "ideal" trajectories are generated, multiple additional potential trajectories are created by applying a potential field from the obstacles to each ideal trajectory. Different weights are used to create a variety of paths. The final trajectory is selected by applying the following metrics to the set of potential trajectories:

- Distance from obstacles
- Sinuosity of trajectory
- Deviation from ideal path

9.5 Waypoint Navigation

The waypoint navigation process is a straightforward one, based on a similar approach that was taken with an entry into the 2008 RoboGames in San Francisco. A list of available goals is input ahead of time into a configuration file. As the vehicle progresses, an A* search is used on the occupancy grid of obstacles to determine a safe path. The obstacles in the grid are "inflated" to prevent the vehicle from coming too close. The occupancy grid uses the same implementation as that used in the autonomous challenge, reducing development time.

9.6 Controls

The high-level controller for both the autonomous and the navigation challenge produce a desired path for the vehicle to follow. Because of this commonality, a single controller can be tuned for both cases. The approach chosen is known as "pure pursuit," which is suitable for low to moderate speed and directs the vehicle to follow a point on the path l_d m ahead of the vehicle (Figure 7).

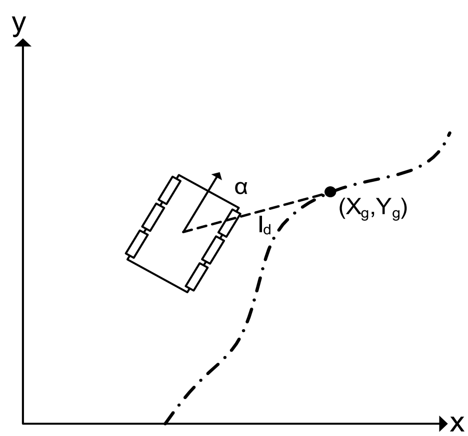

Figure 7:"Pure Pursuit" Strategy

[5] provides a derivation based on a bicycle kinematic model. Since this chassis is slightly different, the equation was altered to

$$\omega = \frac{2K\sin(\alpha)}{l_d}$$

This produces a desired rotational velocity given angle α and a control constant K, which are passed to the low-level board along with a cruising velocity. Once on the board, they are converted into left and right wheel speed profiles based on a predefined linear acceleration.

Dynamically generating a velocity profile in this way lessens mechanical stress on the drivetrain and reduces vehicle slip.

10. System Integration

The design of the system provides for easy system integration. There are three primary phases that will take place:

1. **Mechanical Verification**

   Once the drivetrain has undergone initial assembly, the motors will be run directly from power supplies. This will reveal any issues with the mechanical design (misalignment, for example) before additional parts are mounted to the vehicle.

2. **Hardware Verification**

   At this point, the mechanical operation of the chassis has been verified. The power and control wired will then be routed, the motor controllers secured and the low-level control board mounted. Then, the vehicle's speed control will be tested and tuned by using direct commands from a remote control system. This removes the dependence on high-level software during this phase of testing.

3. **Operational Verification**

   This is the final stage of integration, comprising the testing of the high-level control software with the now fully-assembled chassis. As before, this step builds on systems which are known to work, since the high-level software will have been previously verified in simulation and the rest of the hardware checked during the previous integration steps.

11. Predicted Performance

Despite this being only the second appearance for UWRT at the Intelligent Ground Vehicle Competition, it is expected that Iorek will be successful in both the Autonomous Challenge and Navigation Challenge. Enormous effort has been expended on innovative design with the goals of the competition in mind, and hopes are high that it will pay off! Some expected benchmarks are:

- Cruising speed: 2.0 m/s
- Speed control: Within 0.05 m/s
- Terrain: 20^{o} incline
- Localization accuracy: 0.6 m position, 1 degree heading
- Closest point of approach to obstacles: 0.25 m

12. Design Process

The UW Robotics team tries to maintain a sense of continuity as members graduate and new members join the team. This allows our designs to be iteratively refined over several terms. For example, this chassis is based on a design from September 2008 created for the Canadian Unmanned Systems Challenge, which in turn is a refinement of an entry into RoboGames 2008. The base software architecture dates back to 2007, and it has been continuously improved since then to be compatible with every outdoor autonomous vehicle the team has used.

Because of this strategy, the team is capable of compressing the development schedule of individual robots. We have found that this is a favourable approach, as the co-op system at the university results in very few people being present full-time during the summer, which would be necessary if each design was done from the ground up.

12.1 Team Organization

Team Member	**Academic Department & Class**	**Tasks**	**Hours Expended**
Rabia Aslam	2N Systems Design	Software Development	80
Robert Brooks	2N Mechatronics	Firmware Development	80
Ryan Gariepy	4B Mechatronics	System Design, Software Development	300
Evan Goldenberg	2N Mechatronics	Vision System	80
Kyle Litchfield	2N Mechatronics	Mechanical Design, Fabrication	20
Craig Mackenzie	2N Mechatronics	Electrical Design, Fabrication	120
Pat Martinson	4B Mechatronics	Mechanical Design, Fabrication	80
Megan Pollock	2N Systems Design	Mechanical Design, Fabrication	40
James Servos	2N Mechatronics	Electrical Design, Fabrication	150
Kent Stoltz	2N Mechatronics	Mechanical Design, Fabrication	150
Bryan Webb	4B Mechatronics	Electrical Design	50
		TOTAL	**1150**

12.2 Problems and Resolutions

One of the major problems that the mechanical design ran into was with the chain. The gear box sprocket was too high and the chain was rubbing against itself as it travelled over the top of the gears. In order to prevent the chain from rubbing on itself, idlers were added into the chain's path.

One of the major problems that was encountered with the electrical system was the large amounts of current being drawn from the drive motors which results in a large amount of electromagnetic noise. This noise would result in disruptions in the signal wires and sensors which would be near the motors or the motor wires. In order to prevent this the sensitive sensors such as the compass and GPS were mounted on top of a tower above the robot, far away from any sources of noise. Also the high power relays which are in the electrical box are separated from the electronics with an aluminum plate which acts as a faraday cage and dampens any noise from traveling to the electronics. Also the high power wires and signal wires are routed on opposite sides of the robot to give extra separation.

13. Conclusion

Iorek is a highly robust and capable autonomous vehicle that is the result of tremendous student effort. In addition to being a complete entry into this year's Intelligent Ground Vehicle Competition, Iorek serves as a platform for future improvements and modifications. UWRT looks forward to more exciting opportunities involving Iorek.

14. Works Cited

[1] NovAtel. (2009) NovAtel corporate site. [Online]. http://novatel.com/Documents/Papers/ProPakV3.pdf

[2] Eli Marianovsky, "Advanced Electronic Compass for Mining Applications", Advanced Orientation Systems, Sensors & Transducers Magazine (S&T e-Digest), Vol.43, Issue 5, May 2004, pp.234-239

[3] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," University of North Carolina, USA, Technical Report TR 95-041, 2006.

[4] R. Siegwart and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, 1st ed. Cambridge, MA, USA: MIT Press, 2004.

[5] J. M. Snider, "Automatic Steering Methods for Autonomous Automobile Path Tracking," Carnegie Mellon University, Pittsburgh, Technical Report CMU-RI-TR-09-08, 2009.