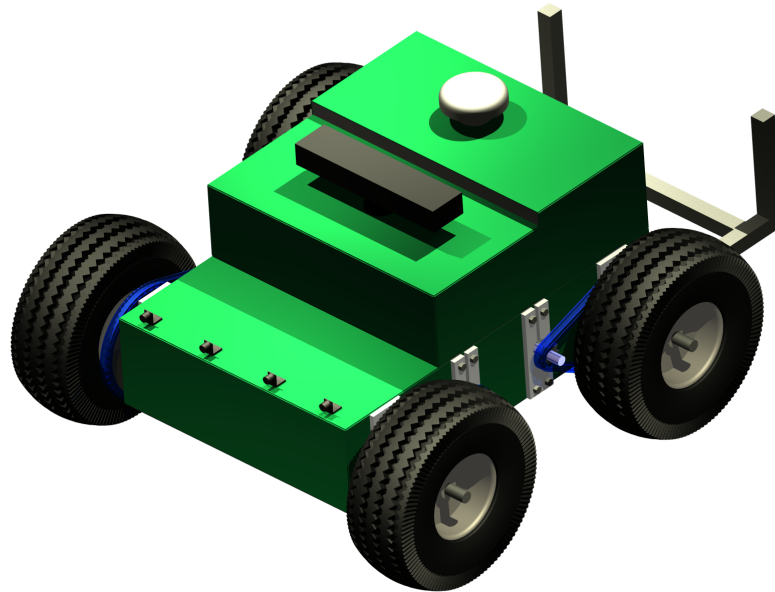
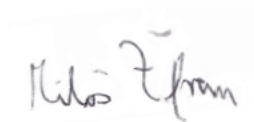


The University of Illinois at Chicago
Kratos
2011 Autonomous Vehicle

Matthew Byrne, Steve Piasecki, Mayur Patel



The engineering design in this vehicle by the current student team has been significant and equivalent to what might be awarded in a senior design course. Faculty Advisor Certification Signature:



1 Introduction

The Engineering Design Team of the University of Illinois at Chicago is pleased to introduce Kratos, an 2011 entry for the Intelligent Ground Vehicle Competition (IGVC). Kratos is a newly-engineered platform designed to be small and fast; the chassis size meets the minimum requirements for the competition and the electronic systems and software are able to deliver a quick response time under any circumstance. The small size and simplified software flow combine to create a vehicle that minimizes the chance that a collision with course obstacles will occur. Kratos was also designed with budget constraints in mind, forcing us to be creative in development with cheap but effective solutions to each problem.

Kratos is a unique platform that includes many components that were designed and built in-house as well as proprietary components such as an XBOX360 Kinect camera. With a few veteran team members, Kratos takes features from many of the entries from previous years' competitions, including our other entry into IGVC. The development of Kratos has only lasted a year but it is continuing process that will continue for years to come.

2 Design Process

2.1 Team Overview

Composed of students from every field of engineering, the Engineering Design Team is a group which is rooted in years of experience with autonomous and remote controlled vehicles. With our large size and past successes we have been granted the equipment and experience which allows nearly our entire vehicle to be constructed in-house. From the circuit boards to the drive trains, components are designed and constructed in small teams led by a team captain. These captains communicate with each other in meetings every week as well as on a day-to-day basis with the IGVC leader. Since we have the manpower to make several people responsible for each component, every task is analyzed by at least two people, at least one of whom was not responsible for actually completing that task. This redundancy not only allows for a more robust vehicle, but one that can also be built much faster.

2.2 Design Process

Our team follows the Engineering Method which can be seen in Figure 1 on the next page. This method is generalized, but it is also proven and is a great tool not only for introducing students to the design process but also for standardizing the process. The central task which occurs each year is the task of brainstorming possible solutions and

analyzing those solutions. It takes up months of our time, months that we spend documenting, researching, and communicating our ideas. Our group concentrates on opening communication between all of our members so that problems with possible solutions can be identified early, documented, and either discarded or improved upon. In order to make this task as coherent and expedient as possible much care is taken to document everything; this means that previous ideas can be looked at for inspiration or to identify past problems that might have relevance to current ideas.

Before this task takes place, a team of officers and captains are responsible for problem identification. This includes looking at the vehicle from last year and identifying major issues that appeared, looking at the modified rules and identifying any problems that might arise, and identifying scheduling issues for the coming year. By analyzing the vehicle that competed in last year's Intelligent Ground Vehicle Competition during problem identification, we eliminate issues that would otherwise come up in the next year's proposed solutions.

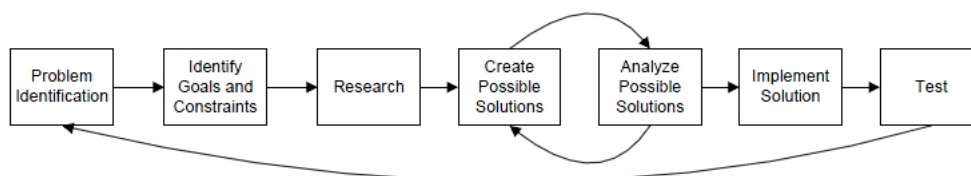


Figure 1: The Engineering Method, The Infinity Project: Engineering Our Digital Future. Upper Saddle River, NJ: Pearson, 2004.

3 Mechanical

3.1 Mechanical Design

3.1.1 Chassis

The chassis is divided into two sections: The bottom chassis and the top chassis. The bottom chassis houses the drive train and electronics. It is made of welded steel tubing in order to provide structural support to the drive train- which can produce significant forces during operation. The top chassis is made of aluminum T-slot which houses the batteries, small computer, and sonar array. The T-slot allows for quick and efficient design, fabrication, repair of the chassis structure, and quick access to electronic components. The ease of modification of the T-slot also allows for the number of batteries and the size of the computer to be increased in the future.

3.1.2 Drivetrain

The drive train of Kratos is a differential drive, which allows the vehicle to rotate in place. The rotation of the wheels is geared down by a factor of 1/8 multiplied by motor rotation. Due to Kratos' compact configuration, the gearing was designed for speed rather than power because it is significantly lighter compared to other vehicles. Each motor drives two wheels per side of the vehicle.

3.2 Mechanical Innovation

- The drive train is of differential drive. Therefore the vehicle is able to rotate in place without having to move forwards or backwards.
- The bottom chassis is made of welded steel to provide structural strength and to hold the forces of the drive train. The top chassis is made of aluminum T-slot which allows for easy modifications or additions. Panels slide along the T-slot allowing quick access to components and part replacement. Also, if any pieces of the T-slot are damaged, it can be replaced very efficiently.
- Filters are in place to prevent outside particles from entering the chassis. Compartmentalization of components simplifies the airflow to electronics and to the computer.

4 Electrical

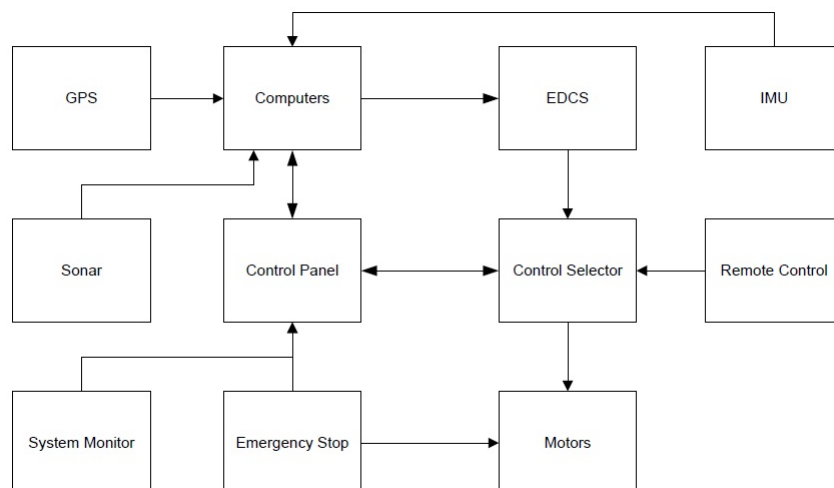


Figure 2: Electrical Flow Diagram.

4.1 Electrical Design

Kratos contains an innovative and robust electrical system rooted in the tenets of reliability, safety, and efficiency. These tenets are realized by designing all components to safely handle and recover from a multitude of error conditions. To increase reliability, each electrical component is packaged in a durable case with clearly labeled ports and of modular design. Furthermore, all microcontroller systems enter idle mode when they are not doing work, thereby reducing power consumption and increasing efficiency. The following sections highlight the electrical components of Kratos.

4.1.1 Power System

Kratos is powered with a set of two 12 volt 26 amp hour sealed lead acid batteries. The electronics voltages are regulated using efficient switching regulators, and the computer is powered with an DC-DC ATX pico power supply.

4.1.2 E-Stop

The emergency stop system (E-stop) is a system designed to stop the vehicle quickly in emergencies. The user may activate the E-stop in two ways, either by a wireless remote control or by pressing an onboard switch. The E-stop is composed of two modules: a handheld transmitter unit that wirelessly transmits a signal to the vehicle and a receiver unit that is onboard the vehicle. The transmitter unit has a highly visible red pushbutton switch that when pressed, activates the E-stop. The behavior of the E-stop is that if either the onboard switch or the wireless transmitter provides the STOP command, the vehicle will stop moving. The vehicle will resume movement only when both sources provide a GO signal. The radio module uses spread spectrum technology to ensure data encryption and prevent interference or jamming from external sources, thus improving system reliability. In addition, the radio module has a range of 5 miles in open-air line of sight.

4.1.3 Embedded Drive Train Control System (EDCS)

The EDCS is a system designed to interface the software controls system to the motors that drive the vehicle. The system consists of a module for each side of the vehicle. Each module is responsible for measuring the wheel speed and controlling the power level of the motor for its side. A microcontroller is the heart of each module and communicates with the software controls system via RS-232 serial communication. The communications protocol between the EDCS and the computers is designed to inherently detect communication errors, leading to improved system reliability and safer operation.

4.1.4 Sensors

Kratos has several types of sensors that provide data to the computer software. 8 sonar units are used to measure approximate distances to objects; four sonar units are placed in the front of the vehicle, and four in the back of the vehicle. The sonar units produce a pulse train with frequencies linearly dependent on the distance to the object. The sonar units then send the pulse train to the sonar control module which measures the pulse width and sends it to the computer. An Inertial Motion Unit (IMU) is used to track the vehicle's movements in three dimensions. The IMU control board measures the readings from the IMU module and communicates the data to the computer. A stereo vision camera is used for line detection as well as object detection. The camera is interfaced to the computers via a firewire connection.

4.1.5 Signal Multiplexer

The Signal Multiplexer allows the user to choose the manner in which Kratos is controlled. This system has four modes of operation that are easily controlled by the user through switches on the Control Panel. The first mode is a safety mode which causes the vehicle to remain stationary; it is also the default mode to ensure safety by not allowing the vehicle to move until a user deliberately allows it. The second mode allows the vehicle to be exclusively autonomously controlled. The third mode allows the vehicle to be exclusively controlled by remote control. The final mode allows the vehicle to switch between autonomous and radio control by the switch of a button on the remote control. In addition the system recognizes the absence of a signal from the remote control and stops the vehicle when no signal is detected.

4.1.6 Control Panel

The Control Panel interfaces with every electronic system on the vehicle through the I2C interface, allowing it to send commands and receive data. In this sense, the control panel acts as a medium between devices, as it is the only device capable of controlling others; if one device needs access to another, it must indirectly access it through the control panel. The secondary function of the control panel is to gather status information from devices and forward the information to an LCD screen, where the user can view it, and to the computers for feedback. This feedback greatly increases vehicle reliability and safety since it informs the software of possible hardware system errors. Status information includes Signal Multiplexer status, E-stop status, EDCS data, battery status, and fan status. One can adjust settings for the Signal Multiplexer and fan operation using buttons located on the Control Panel. Other features include ports for connecting monitors, keyboards, and mice to the computers, permitting quick access to the computers.

ENYO Control Panel				
SYSTEMS	EDCS 1 EDCS 2	STATUS		
ESTOP	GOOD	80% 2m/s -20% -1m/s	WiES	GO
SigMu	GOOD		OnES	STOP
SysMc	ERR1	SONAR 1	SigMu	Auto
EDCS1	GOOD	500cm 206cm 198cm 514cm	CB1	GOOD
EDCS2	GOOD	SONAR 2	CB2	POP
Son1	ERR2	600cm 600cm 600cm 600cm	12V	GOOD
Son2	ERR2		24V	CRIT

Figure 3: Example of the control panel display.

4.1.7 System Monitor

The System Monitor is responsible for monitoring the systems of the vehicle and detecting malfunctions. If malfunctions occur, the System Monitor notifies the Control Panel, which in turn notifies the software and the user. One key task of the System Monitor is to continuously measure remaining battery life and warn if the battery capacity is becoming low. The system has two types of battery warnings for both the 12 volt and 24 volt systems, and another type of warning for each motor. Based on these warnings, the user or software can take action to correct these faults, allowing the vehicle to run in the safest manner possible.

Event	Electrical Response	Software Response
Battery Low	Msg: Change Batt. Soon	Limit Speed
Batt. Critically Low	Msg: Change Batt., Switch on E-stop	None
Motor Pwr. Lost	Msg: Motor Pwr. Lost	Wait

Figure 4: System Monitor Warning Chart

4.1.8 Computer Systems

The vehicle contains one small form-factor computer to do image analysis, sensor data analysis, and control the vehicle. The computer is powered with 12V from the batteries. It has an Intel Atom Z550 CPU and an NVidia Ion2 graphics processor that has CUDA support. An 80GB laptop hard drive is used to store data. Laptop hard drives were chosen for their resistance to vibration damage during vehicle operation.

4.2 Electrical Innovation

- The electronics are powered through switching voltage regulators that efficiently regulate the 12 volts from the batteries down to a clean 5 volts. The motor con-

trollers receive 12 volts which is capable of supplying copious current of torque to the motors.

- The EDCS is capable of adjusting the speed of the wheels according to the commands that are issued by the computer software.
- The Emergency Stop is capable of remotely triggering a relay to cut off the power to the motors at a range of up to five miles.
- Kratos has several different types of sensors that together make for an excellent source of feedback for the control software: there is an inertial motion unit (IMU), a kinect camera, and sonar sensors.
- The control panel is readily accessible as it is located on the side of the vehicle. It displays the state of all of the electrical components and all vital data to a Liquid Crystal Display. One can also monitor the battery level and the state of any protection circuits located in the vehicle.

5 Software

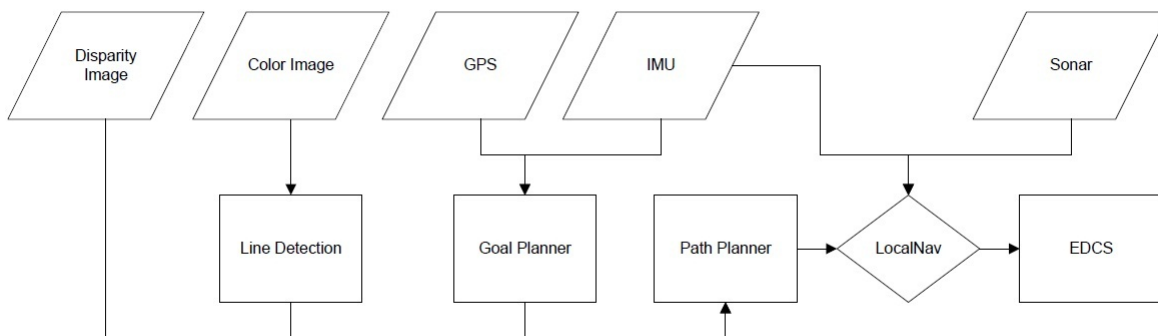


Figure 5: Software Flow Diagram.

5.1 Software Design

5.1.1 Goal Planner

At IGVC the course is dominated by objects and lanes which can be used to help guide the vehicle through it. By taking advantage of this fact the vehicle can navigate by choosing to stay close to lines and obstacles. By choosing a goal that is toward lines and obstacles it guarantees the vehicle is motivated to stay within the boundaries of the course. This goal has a bias toward an immediate opening near the vehicle. By looking

at the map of the local area around the vehicle a waypoint is generated by looking at the density of objects and lines and matching that vector with the direction the vehicle has been travelling.

5.1.2 Path Planner

Obstacles and lines on an occupancy grid are grown to the size of the vehicle in order to determine paths that the vehicle can safely traverse. Waypoints are then generated by creating a scaffold of points that are linked by edges from objects and lines. This scaffold simplifies the possible routes a vehicle can take through the local environment and a path through these nodes and quickly and easily be generated using a heuristic search algorithm like astar.

5.1.3 LocalNav

Once a set of waypoints to a goal has been defined, the speed commands to each motor can be defined by creating a map of vectors which defines the path of least resistance and by calculating the optimum turn radius for each waypoint. This component also receives range data from a sonar unit on the front and avoids objects that may have not been detected by the vision system or whose location was estimated incorrectly. Power to the motors are regulated by observation of data collected from the IMU, which is adjusted incrementally in order to quickly adjust for errors.

5.1.4 Line Detection

Lines are extracted from a color image which is acquired from the Xbox360 Kinect camera. A pyramid segmentation algorithm is used to distill important features from the image and get rid of noise and unimportant data. Edges are then found in the image and lines are generated using a Hough transformation. Lines that do not match the vector to the next waypoint within a given threshold are discarded and the lines that remain can be connected to fill in gaps. Eliminating lines that do not match the direction of the next waypoint keeps noisy data from interfering with mapping.

5.1.5 Stereo Vision

Objects are extracted from a disparity image which is acquired from the Xbox360 Kinect camera. Disparity is calculated on-board the camera from an infra-red emitter and camera. Objects can be extracted from this image and mapped in linear time allowing for as little computation as possible to take place on the main CPU.

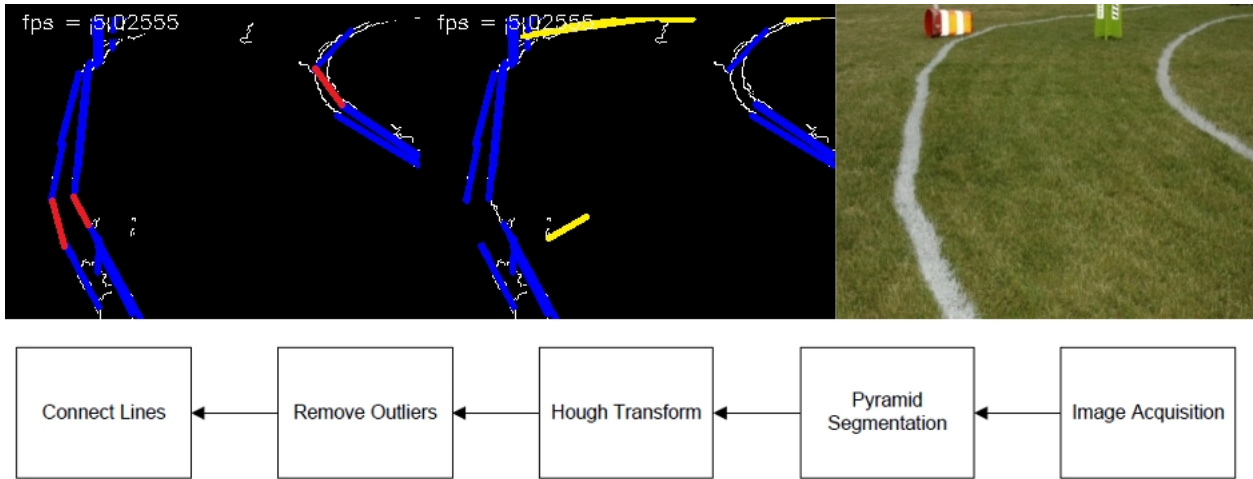


Figure 6: Line Detection Algorithm.

5.2 Software Innovation

- Disparity calculation is handled on-board the camera, freeing up valuable CPU cycles.
- A simply waypoint-generation algorithm that keeps the vehicle inside the course allows for decisions to be made quickly.
- The local histogram plus utility uses sonar data to avoid objects that were not correctly or completely detected by the stereo vision system.
- Line detection eliminates lines that are not relevant to the vehicle, allowing for less errors.
- The Xbox360 Kinect camera provides object detection and line detection in a single unit and is a fraction the price of a laser range finder without sacrificing too much accuracy.

6 Safety

Our vehicle excels at being designed to operate safely and without incident. During the design process time is taken to add functionality to the vehicle which specifically relates to safe operation. Each system is double-checked by several members during the design process to ensure that interaction with any other system does not cause any safety concerns for vehicle operation. Also these safety features are tested multiple times upon completion.

- The E-Stop provides a way of disabling the vehicle with the press of a button from up to five miles away. If signal is lost from the E-Stop, the vehicle ceases to function

until a signal is re-established.

- The speed of the vehicle is limited mechanically to around 9 miles per hour. Additionally the software limits the acceleration of the vehicle during operation.
- Feedback from the control panel allows the operator to know through text and through lights which mode is currently selected and whether the E-Stop is enabled or not.

7 Testing

7.1 Testing Process

The testing process for our vehicle begins in the design phase of our vehicle wherein the constraints of each subsystem are tested in simulation; the total output of our drivetrain under the estimated weight of the final vehicle under different conditions is tabulated, the different electronic components can be simulated virtually, and the navigation subsystems can be tested with fake sensory information. Unit testing is done periodically when each component is finished by using stubs for software testing, computer interfacing with electronic components, and running the drivetrain under controlled conditions. Subsystem testing is then used when many components are finished via simulating different conditions by creating fake sensory information for the other subsystems; for software subsystems that means using archived data from previous years' competitions, for electronic subsystems that means interfacing with incomplete subsystems and slowly increasing the functionality of each subsystem, and for mechanical subsystems that means testing the vehicle with its full weight and in different conditions. Integration testing follows and is the last remaining step before the vehicle is ready to compete.

7.2 Testing Results

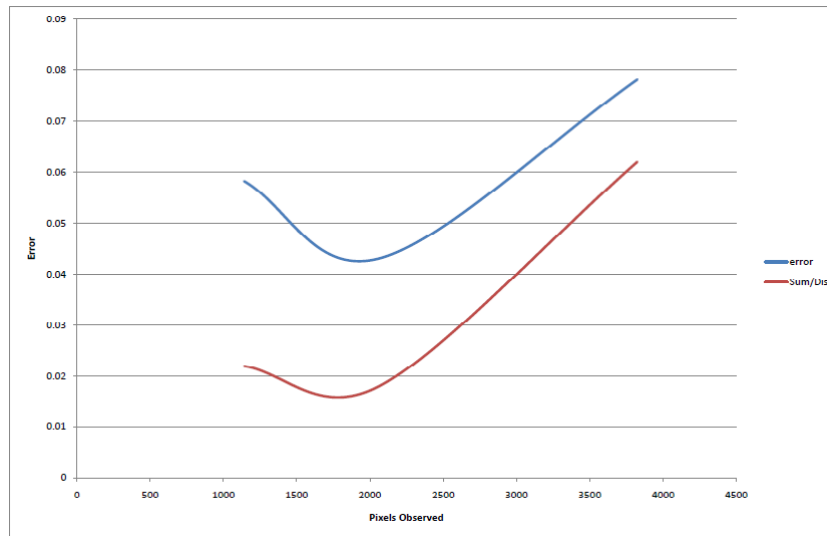


Figure 7: Line detection results show the error of detected pixels against actual line pixels. This shows that the algorithm used is more accurate for images with a medium amount of data. Since three tests were performed, each with a different size image, the middle image resolution of 340x220 provides the most reliable results.

8 Performance Summary

	Speed	Battery Life	Max Incline	Avg. Obstacle Detection	Waypoint Accuracy	E-Stop Range
Predicted Performance	9 mph	30 mins	<10°	15 ft	± 3 ft	5 mi

Figure 8: Performance Chart.

9 Cost

Component	Type	Retail Cost
Sonar		\$240
Batteries	Lead-Acid	\$300
IMU		\$40
GPS	Hemisphere	\$1600
Camera	Xbox360 Kinect	\$150
Gears, Sprockets, Wheels, and Chains		\$280
Raw Materials (Aluminum, Steel, and Plastic)		\$95
Electronic Components and Wire		\$1150
Motors		\$1100
Remote Control		\$350
Computer Parts		\$1250
Total		\$6555

Figure 9: Cost Summary

10 Conclusion

Kratos is an exceptionally quick, cheap, simple, and compact vehicle. By utilizing compartments, the vehicle is able to fit more into less space. By forcing the price of the vehicle to be low, novel solutions were developed that would otherwise have been left uninvestigated. The low price also leaves more resources for updates to the vehicle. The simplicity of the design of most of the vehicle's systems allows for less to go wrong and for the vehicle to have faster response times than other, larger and more complex vehicles. The small size allows for more room for error when navigating the course and for a higher max speed and acceleration. Our team has spent a significant amount of time and resources on this vehicle and hope that it shows in this report and through its success in the Intelligent Ground Vehicle Competition.