CALCULON

Faculty Statement:
I certify that the work done by all students on this project is consistent with a senior design course and that the vehicle has been significantly modified for this year's competition.

_____
Dr. Fernando Gonzalez
fgonzale@pegasus.cc.ucf.edu

1. Introduction

This paper proudly presents the newest addition to the University of Central Florida's (UCF) ground-based robots, Calculon. Calculon will be competing in the 13th annual Intelligent Ground Vehicle Competition (IGVC) in June of 2005. This vehicle is the culmination of many hours of work by all team members built upon experience from UCF's previous entries in the IGVC. Calculon has been designed and constructed to excel at all three competitive events at the IGVC and to address all problems encountered by previous entries.

Our goal with Calculon was to achieve an intelligent and able robotic platform which leverages an incremental design process, modular software design and use of commercial off-the-shelf products (COTS) for construction.

2. Design Paradigm

Through experience and researching robotic systems design and implementation, we found that strict and unyielding design processes such as The Design Life Cycle or the Spiral Model were not well suited to our group, lab structure or our approach to this competition. Instead, we have followed an
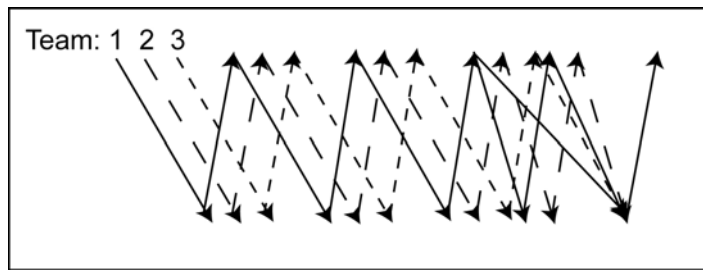

Figure 1: Parallel team development with the 'W' paradigm

incremental design and development process, which allows for more flexibility when needed as well as for parallel development from various sub-groups which, was essential to our team.

Our incremental design flow followed the 'Validation V'. This design flow follows the same path as many other design paradigms, namely requirements stage, system design, construction, testing and completion. With just this simple flow, it is hard to see how this paradigm is incremental. However, by connecting the 'Validation Vs' together we see that we get a 'W' formation which represents the synchronization of multiple teams working together. With the use of COTS products during construction along with the use of standardized hardware and software interfaces and protocols, the teams were able to attain modularity to the
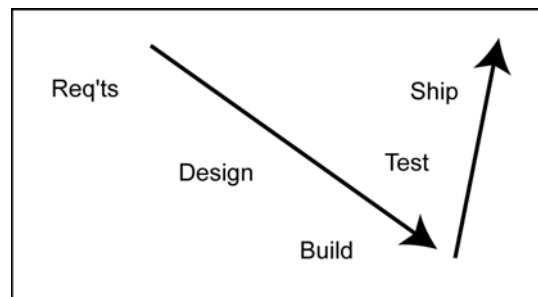

Figure 2: Illustration of the Validation 'V'

entire project that allowed many of the components to be built or worked on simultaneously and in parallel. The 'W' paradigm works exceedingly well with this level of modularity. During project pre-planning, we found this design paradigm to be most realistic for our group in terms of the actual

implementation. Many others seemed as though they may work well in theory, but for a group of our small size and project such as an IGVC vehicle, the 'Validation V' and 'W' concepts were appropriate.

2.1 Team Organization

The Calculon build team consisted of students from various disciplines at UCF including Computer, Electrical, Mechanical Engineering and Computer Science.

Name	Responsibility	Class Standing	Hours Spent
Daniel Barber	Team Captain, System Design, Software Systems Design, Lead Programmer, Sensor Integration, Obstacle Avoidance, Steering and Speed Control, Path Planning, Electrical Systems, Chassis Fabrication, Artwork and Logo Design	Graduate Student, CpE	750
Tim Roberts	System Design, Power Systems, Electrical Systems, Chassis Fabrication, Path Planning, Funding	Graduate Student, EE	380
Brian Becker	Vision System Design, Software Integration	Undergraduate, CpE	400
Anton Kiriwas	Vision System, Dead Reckoning	Graduate Student, CS	200
Chase Hansel	Original Vehicle CAD Design	Undergraduate, ME	20
Rick Porter	Chassis Modification, Cover Fabrication and Design	Undergraduate, ME	40
Matt Ganci	Chassis Modification, Cover Fabrication and Design	Undergraduate, ME	40

3. Design Innovations

To create a winning vehicle over previous UCF entries, the team made use of a COTS platform and high quality sensors for obstacle detection and vehicle control. Using an electronic wheelchair as the base reduces construction time, and guarantees chassis reliability. Use of a more accurate DGPS and digital compass also improve vehicle control and stability over previous designs.

In previous years, the primary source of obstacle detection was machine vision, this year a SICK LMS Laser Range Finder has been added to the sensor suite, combined with a high-quality 3CCD digital video camera. In past designs, the use of standard 1CCD cameras created poor input data for vision processing. The high color and resolution images produced by a 3CCD camera make accurate vision processing a reality in varied lighting conditions.

3.1 The Discover Vision Engine

Our final innovation focuses on the creation of rapid application development tools. The design plan called for the creation of tools in order to better facilitate the vision engine's development time. The need for these tools arose from the length of time necessary to modify C++ source code. To change the order of the filters used by the vision engine or tweaking various parameters, a developer would need to close the program, make the changes, wait for a long C++ recompile time, linking, and re-running of the program. To combat this time-consuming method of testing and experimenting, we developed a rapid

prototyping tool called the Discover Vision Engine. The aim was to be able to dynamically see the results of modifying filter parameters, running new filters, and testing vision related algorithms with minimum latency in between edits.

The Discover Vision Engine is modeled after existing tools such as MatLab, except aiming to present an interface capable of easily manipulating images instead of algorithm development. The GUI interface has two main components: a code editor and an image display. The code editor incorporates syntax highlighting and line numbers. The image display uses two modes: a single pane for showing
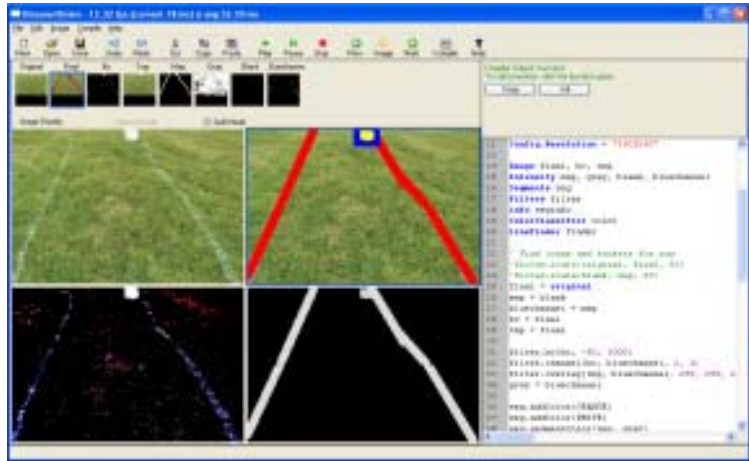

Figure 3: Discover Vision Engine screenshot

the full details of an image, or four panes, each pane containing a separate scaled down image, allowing multiple filtered images to be viewed simultaneously in real time. This is extremely useful to determine how affective a specific or combination of filters can be. The Discover Vision Engine can load and process images from a number of sources, including image files, video cameras, and saved videos.

A simple scripting language was developed for the tool in order to interface between the GUI and the C++ backend. The scripting language currently supports over ten classes and nearly a hundred member functions and is easily extensible. The scripting language supports overloaded functions as C++ does and also offers the ability to create user defined functions that can be called with a click of a button. To increase speed, the script is compiled into byte-code for the interpreter to run. Each 'recompile' happens without any interruption to the programs execution, meaning you can immediately see the difference on the images, even in the middle of processing a video stream.

A user can open, save, and create new scripts, allowing different aspects of the vision system to be tested individually. Furthermore, the compiled scripts can be exported as well. Because of the modular design, the interpreter can be separated from the GUI and used in another framework without difficulty. The Intelligent Client application that controls autonomous behavior uses the interpreter module. This allows us to develop and test a script using the Discover Vision tool offline using existing images or video, and then load it into the client application to serve as the vision system. The client application is further described later in this paper.

4. Mechanical System

In the continuing effort to use COTS components, a Ranger II Storm Series[tm] motorized wheel chair was chosen as the base chassis. The wheel chair fit**s** all of the projects needs since it provides a stable mobile platform with powerful motors, integrated motor controller, and large rugged wheels.

The wheel chair uses a differential drive system making it very maneuverable and easy to control. In previous projects, an Ackerman drive system was used which made path planning and control more difficult. The small turning radius provided through differential steering lets the vehicle navigate around all obstacles easily.

Two large motors control the 12.5 inch diameter front wheels. Two small, caster wheels, allow for smooth motion and in-place turning. In addition to the wheel chair frame, additional mechanical components were added to facilitate the robotic platform. This includes the center pole which supports the digital compass housing, the digital video camera, as well as the DGPS unit. Also added was a front mounting bracket to attach the laser rangefinder and a hood to protect it from the environment as well as to shield from

Figure 4: Calculon Solid Edge CAD drawing

any ambient infrared light that could interfere with operation. Finally a drawer was added to the back of the platform. This drawer houses all electronics necessary for operation as well as power distribution units. The front panel of the drawer contains power switches, battery charging port and electronics reset. On top of the draw is a cork-padded resting place for the main computer, which is described in more details in the electrical systems section of this paper. A fiberglass cover was also constructed to provide environmental protection against unforeseen conditions as well as the daily dust that would accumulate on the robotic components.
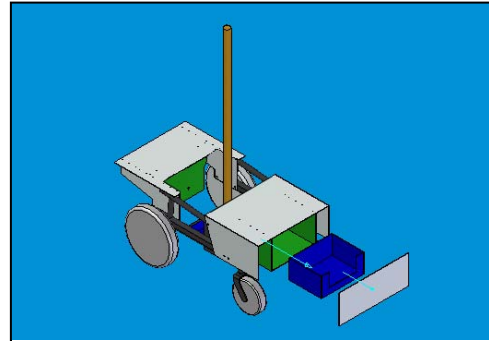
5. Electrical System

5.1 Power Distribution

Calculon's power is provided from two 55Ah 12-volt batteries connected in series to provide a single 24-volt power supply. The 24-volt source powers the wheelchair platform's motors and motor controller. The platform's motor controller also provides a 5-volt power source for custom electronics needed to interface the motor controller to the computer system. This 24-volt power source is also connected to a DC-to-AC inverter which supplies power for the main computer and the digital video camera. The 24-volt source also connects to a 24V-to-12V and a 24V-to-24V converter. The 24-to-24 converter functions as a regulator to guarantee a 24-volt source for the SICK LMS 291 laser ranger finder

and Trimble DGPS. The 24V-to-12V converter provides a regulated 12-volt source to power the magnetic encoders, digital compass and cooling fans within the electronics drawer. Two LEDs on the front of the electronics drawer provide instant verification that electrical systems are functioning properly.

5.2 Electronics

A PIC 16F876A MCU and a D/A converter are used to interface with the onboard motor controller of the wheel chair. Through this interface the MCU can vary the speed and steering of the wheel chair. A second PIC MCU is used to capture signals from the magnetic encoders mounted on each wheel. An RS232 connection from each PIC to the main computer allows it to get encoder information and control the speed and steering of the vehicle. Two individual connections were used to prevent communication saturation. This lets the MCU connected to the encoders continuously transmit information while not blocking drive commands to the MCU interfaced to the motor controller.

If one were to assign a 'brain' to Calculon it would no doubt be the Alienware Area-51m 7700 notebook. With an Intel Pentium 4 3.4 GHz Processor with Hyper-Threading Technology and 1.0 Gigabytes of RAM, the computer controls driving, path planning, vision systems and interacts with the rest of the robot's systems via multiple RS-232 serial interfaces, RS-422 serial interface and FireWire. The computer uses the Microsoft Windows XP operating system.

When not in autonomous mode a Logitech Wireless Joystick is used to drive Caluclon. This joystick's receiver is connected to the main computer over USB, and the server program uses information from it to send speed and steering commands to the onboard electronics. The server program is further described in the software section of this paper. This wireless joystick has an operational range of 40 feet, and is used to switch the vehicle between manual and autonomous drive modes.

6. Sensors

In order to create a winning vehicle, highly accurate and efficient sensors are used by Calculon. A SICK brand LMS Laser Range Finder provides two-dimensional information 1 to 9 meters in front of the vehicle in a full 180° sweep. This information is transmitted back to the main computer by RS-422 at 500kbps giving exact placement of obstacles relative to the vehicle. The 500kbps data rate gives real time obstacle locations which are placed into a local map which is further described in the software section of

Figure 5: SICK LMS 291 Laser Range Finder this paper.

A Sony DCR-HC1000 3-CCD Mini-DV digital video camera is used for machine vision. This camera provides a clear 720x480 resolution image. On previous vehicles 1CCD web-cams were used which do not

Figure 6: Sony DCF-HC1000 DV Camera

6

handle varied outdoor lighting conditions representative of the IGVC. The 3-CCD chips of the Sony digital camera provide high quality color and white balance reduction in various lighting conditions.

Figure 7: Trimble DSM 132 DGPS Unit

For location of DGPS waypoints, a Trimble DSM 132 DGPS is used. This provides sub-meter accuracy with an update rate of 10Hz. The Trimble is connected to the main computer over an RS-232 connection at 19200 bps. The Trimble was chosen because of its high accuracy and reasonable price. Other systems provided more accurate information, but at a much higher price. The extra level of accuracy of more expensive units was not necessary for the 2 meter radius required of the IGVC. Latitude and Longitude information from the Trimble is converted to Universal Transverse Mercator (UTM) System. UTM provides a cylindrical projection in meters which is highly accurate for use in waypoint finding.

A Coral AHRS digital compass provides heading information. An internal Kalman filter keeps the compass accurate within 1 degree. Heading information updates at a rate of 100Hz with filtering, providing almost instantaneous response to physical vehicle movement. Like other sensors in the system, this sensor connects to the main computer via RS-232 at 56 kbps. The Coral AHRS was chosen over others because of its fast update time and accuracy, reducing oscillation when controlling vehicle heading.

Two Baumer Electric MDFK 08G2101 magnetic sensors are placed on each drive wheel. Each reader uses a 256 pulses ring magnet with 16 pole-pairs. The pulses from these sensors are used to calculate vehicle speed and position. This information is used in speed control and dead reckoning. The output is connected to a PIC MCU which transmits the information to the main computer via RS232. These magnetic sensors are easy to integrate with the COTS chassis and give fast and accurate information.

7. Software System

The overall goal for Calculon has remained the same since work began; to leverage COTS components in order to build a functional, easy to use, easy to maintain, and modular robotic platform. On top of this platform is the intelligence of the system, the software. After exploring various alternative programming methods such as pre-built robotic control libraries and visual based programming tools such as LabVIEW, it was decided none of these options afforded the programming team both the modularity and the low-level control they wanted. Instead, the software was implemented in C++. Since the software team was already knowledgeable in C++, an additional training phase would have been needed to learn a different programming tool. This would have delayed construction of the software systems. Using C++ provided the team with the flexibility and low-level control that we needed, without any delay in development. To avoid some of the typical complexity problems associated with C++, a strict object-

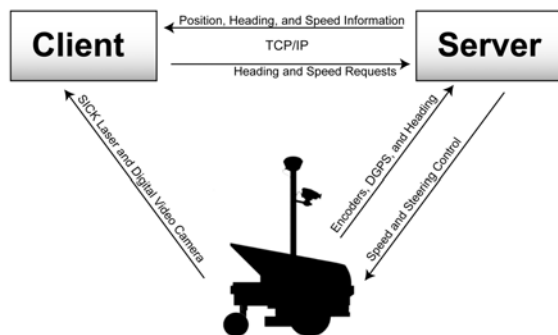oriented development guideline was instituted resulting in software which was interoperable since and modular.

Figure 8: Client/Server communication paths

In keeping with a modular software system, two main components make the software of Calculon. The two components are the server and client. The server program acts to maintain vehicle speed and direction while continuously retrieving positional sensor information and wireless joystick input. This server is a low level system upon which intelligent systems can be built without needing to worry about low level details by providing speed and heading control and position information for the client. The client and the server communicate over a TCP/IP connection. The client program acts as the intelligence of Calculon. This client performs obstacle detection, path planning, and obstacle avoidance. It is able to drive the vehicle by sending speed and heading commands to the server.

7.1 Server

The server program performs two main functions. The first is to create modular system design providing a platform for the development of higher level programs. The second is for manual driving of
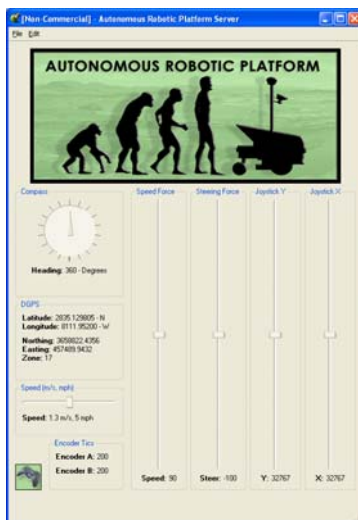

Figure 9: Server program user interface

the vehicle by wireless joystick and to initiate autonomous drive mode. The server program interfaces to the DGPS, compass, encoders MCU and vehicle control MCU. It continuously receives the output from the sensors making it available to the client over TCP/IP. This acts as a level of abstraction so that sensors can be replaced without affecting the client program. Sensor information is also displayed on a graphical user interface for quick debugging. The user interface displays the current heading, position, and speed of the vehicle so that the user can quickly determine if a sensor functioning. Settings for these sensors can be modified through built in menus. This program also provides an interface for the client to set the speed and heading of the vehicle. Speed and heading control loops within the program communicate with the vehicle hardware to maintain the desired speed and heading requested by the client when in autonomous operations. All of this functionality frees the client from having to interface to additional sensors and maintain vehicle speed and heading so that it can perform more "higher level" tasks.

8

7.2 Client

The client designed to connect to the server is the intelligent behavior engine of Calculon. Within this program machine vision, path planning, and obstacle avoidance is carried out. While connected to the server, positional and heading information is retrieved, and the desired speed and heading are requested. Two different path planning algorithms are used by the client, one for the Autonomous Challenge, the other for the Navigation Challenge. Specific path planning algorithms are chosen based on requirements of the specific challenge. The differences in the path

Figure 10: Client program user interface

planning are further described in the Autonomous Challenge and Navigation Challenge sections of this paper. Both versions use the same machine vision engine, driver program, lookout, Laser Range Finder interface, local map, and server interface. These systems are further described later in this paper. A graphical user interface displays the local map, video from the camera, classified image data, and output from the path planner. This interface is extremely useful to quickly determine what the vehicle detects and where it plans to go. Buttons and settings tabs on the interface are used to switch between Autonomous and Navigation Challenge modes and configurations.

7.3 Vision Engine

The vision engine is built to be completely modular and to interact with our laser rangefinder to make a single cohesive sensor system which the higher level intelligence, namely the path planning and driver systems, use to complete the competitions. The system makes use of a Gaussian-based color classifier for finding obstacles and a cellular decomposition based Hough transform which gives us a curve-like approximation for line finding. These and other filters are accessed through script files created in the Discover Vision Engine program. All identified obstacles found by the vision engine are placed into a local map.

Through work with previous entries in the IGVC as well as research over the past year, it was found that most of the sensor information necessary to complete both the autonomous challenge as well as the navigation challenge can come from a combination of the Laser Range Finder and visual color information. Many items in the courses can be classified purely by color, such as grass, painted lines, orange cones, etc. While this in no way constituted the entire effort towards classifying course items, it was an important part; because of this fact a large slice of development time towards color classification algorithms was dedicated.

7.3.1 Color Classification Algorithm

One of the filters used is an adaptation of a Gaussian-based skin color classification system. It is trained on images which found to be typical of the course. Training consists of using images from the camera system along with hand made custom classification masks. The

Figure 11: Color classification training image and mask

training algorithm then places a small Gaussian sphere of either positive or negative values depending on the mask into a large RGB cube. Once the training has been completed, color recognition is done by comparing each pixel in the input image's color to its corresponding location in the RGB cube. If the value at that location is over a preset threshold, then the pixel is assigned the trained color.

7.3.2 Line Detection

Line detection is performed by applying a Hough Transform across a segmented filtered image. Several filters are used to preprocess the image before applying the Hough transform. These filters serve to remove noise and other obstacles which do not represent lines. Filters used include the color classification algorithm, brightness and contrast adjustment, and segmentation filter. The resulting image is decomposed into a matrix and the Hough transform is applied to each cell. Once line

Figure 12: Example line detection image

segments are identified, an additional step is taken to join adjacent lines together. This is done to fill in gaps between lines which are typical of the Autonomous Challenge course.
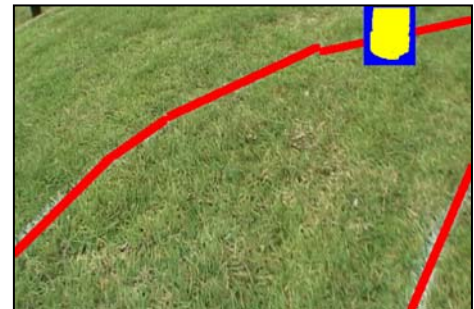
7.4 Driver and Lookout

Good sensor data is useless unless the vehicle is intelligent enough to make use of it to provide the proper physical movement necessary to complete the competitions. The teams design called for breaking the job of moving the vehicle through the competition courses into three major parts; path planning, driver, and lookout. Breaking the software up in this way allowed us to maximize parallel development and modularity within the software.

7.5 Path Planning

7.5.1 Autonomous Challenge

For the Autonomous Challenge, the path planning system relies entirely on a local map system. The robot is able to make decisions, plan the path and traverse the course entirely with only local information and a small amount of history. The actual algorithms used in path planning include Gaussian
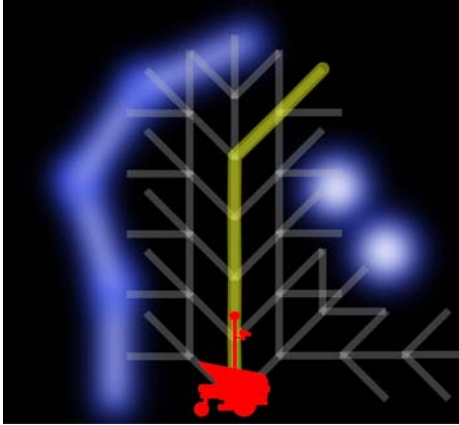
Figure 13: Autonomous Challenge path planning diagram

potential fields to create an entirely Gaussian search space where higher values indicate a lower probability of being on the path. Obstacles and line information from the Laser Range Finder and digital camera in the local map are converted to Gaussian space. From this new map, a maximum of 300 possible solutions through the local map are generated. These paths are then searched using a depth first branch and bound method which takes into consideration which paths are safest (in terms of lower values in the Gaussian space), follow a straight path when possible, and go the farthest distance. Specific safety checks are also performed on the paths to verify that they are valid. These safety checks include verifying that the path does not go over obstacles, lines, or off the course and that the vehicle has enough space to successfully turn around in the case that it has entered a trap. The path that is finally chosen will have a distance of up to eight meters and consists of a series of points which can be given to the driver software. A path history is also kept to guarantee that the vehicle is not retracing an old path except in the case of escaping from a dead end or trap.

7.5.2 Navigation Challenge

While the Autonomous Challenge needed only information on its local surroundings in order to properly steer through the course successfully, the Navigation Challenge requires global knowledge in order to most efficiently reach each GPS waypoints without getting caught in a trap by obstacles, navigating off course or mistakenly returning to the start point. The path planning algorithm for the Navigation Challenge starts the vehicle within a boundary known as the start box. Internally, Calculon keeps a list of the navigation GPS coordinates for the waypoints, which it drives to in order. A Global map keeps track of the location of the vehicle as well
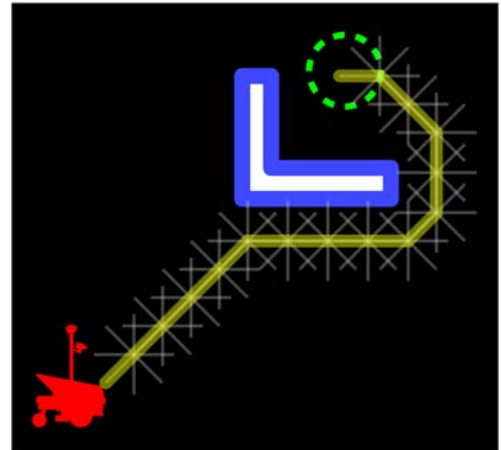

Figure 14: Navigation Challenge path planning diagram

as the location of each of those GPS way points. As soon as the vehicle has left the area designated as a start box, the algorithm places a large, artificial obstacle there, which will prevent the vehicle from inadvertently passing over the start area before all waypoints are reached. Once the vehicle has reached all waypoints in its internal list, other than the return to the start position, the artificial obstacle is removed from the global map, which allows Calculon to return to its start position and complete the course. The global map kept for this path planning method is constantly being updated from a local map which

contains sensor data from both the Laser Range Finder as well as the vision engine. This local map data is then transformed according to the vehicle's global position and added to the global map. The paths between way points are created using a Rapidly Expanding Random Trees method. This method starts at the vehicle's current position and branches from that position, iterating over a 360° sweep. Each new branch position is evaluated, taking into consideration whether or not it is still within the boundaries, that it does not go through an obstacle and finally safety checks which guarantee that the vehicle will be able to turn around safely should this path prove to be a trap or dead end. Given that all of these safety checks are satisfied, the algorithm will choose the branch which moves the vehicle closest to the target waypoint. The process is then repeated, plotting a safe and efficient path from the vehicle's current position to its next waypoint. This method will plot a course which will reach a waypoint within one meter.

7.5 Driver

Since the software systems are built in a hierarchical way in order to best make use of parallel development, the 'driver' program makes use of the path planning algorithms. Once the path planning algorithms generate a set of points to represent the proposed path, it is the responsibility of the driver software to decide the speed and direction of the vehicle necessary to reach and follow those points. Based on the current position of the vehicle, the driver determines what heading and speed the vehicle should go to reach each point on the path while checking to see if the point has been reached. Since the server program does speed and heading control, the driver only determines the best settings for path completion. Through this method, the driver is able to maintain path with accuracy within 1 foot. This makes Calculon capable of driving within the minimum six foot distance between obstacles and course lines. A lookout program makes sure that the driver does not accidentally hit any obstacles in case the path planner chose a poor course.

7.6 Lookout

The combination of the path planning algorithm along with the driver software allows the vehicle to successfully complete both competitions in an efficient and timely matter, in theory. However, good design as well as experience has proven that theory and reality do not always coincide, especially in competition. Because of this, a 'lookout' program whose responsibility is to keep track of nearby objects both from the vision engine as well as from the laser rangefinder data is used. In the unlikely event that the path planner routes the vehicle too close to an obstacle or other situation that would be hazardous to the vehicle, the lookout program takes over control from driver program to stop the vehicle. The lookout continuously analyzes the vehicles surroundings and calculates if the vehicle will hit an obstacle based on the current speed, heading, and time it takes to stop. If it determines a hit is imminent, the driver is stopped. When the driver is stopped, the current course to be followed is erased, and the path planner is required to find a new solution.

8. Vehicle Performance

The values and analysis below represent information and value found through the various testing phases of development.

8.1 Speed

The COTS wheel chair used as a mechanical platform was built to be able to transport an average sized person at low speeds. Because of this, speeds are limited to approximately 3.0 mph. The software system is able to maintain speed within 0.5 mph, and never exceeds 2.5 mph.

8.2 Ramp Climbing

Calculon's wheelchair based mechanical platform was designed to carry humans weighing up to 300lbs. With this fact in mind, Calculon was tested at a variety of different ramp gradients passed 10°. Because we did not wish to risk damaging the vehicle and because IGVC competitions only contain gradients of up to approximately 8.5°, we decided not to determine the maximum gradient that the vehicle was capable of handling. However, Calculon has successfully driven up a ramp of 10° with a payload of over 200lbs. Thus, we do not expect any ramp encountered during the competitions to pose any problem for Calculon.

8.3 Reaction Times

Calculations based upon software loop time and sensor refresh rates have found that Calculon, in addition to planning its path around early detected obstacles, is able to detect new obstacles and react in approximately 0.16 seconds. After substantial testing we have determined that Calculon is able to react and stop to new obstacles and unforeseen objects in its path within less than 12 inches. In specific test cases, a moving obstacle was placed in Calculon's path, and the vehicle stopped before hitting it; thus verifying a quick reaction time.

8.4 Battery Life

As explained more fully above in the power distribution section, Calculon's power is supplied by 2 55Ah 12-volt batteries. Testing has shown that these batteries are able to run Calculon for up to four hours on a single charge. For the competition we will be bringing two sets of batteries, which can be easily interchanged allowing for a total run time of 8 hours without recharging.

8.5 Obstacle Detection Distance

Calculon's sensors are able to detect solid objects in front of it from as far as 80m through the use of its laser rangefinder sensor. Depending on the angle of the camera, Calculon's vision engine is able to detect and identify object as close as directly in front of it and as far as 4 meters. Through testing it has been found that the vehicle can consistently and reliably detect obstacles up to 9 meters away through a combination of sensor data from both the laser rangefinder and its digital video camera. This distance has been shown to be more then adequate for completion of both competitions through testing.

8.6 Dead Ends, Traps, and Potholes

Calculon is designed to be able to deal with dead ends, traps, and potholes throughout both the autonomous challenge as well as the navigation challenge. Calculon's path planning algorithms keep a complete record of the paths previously taken by the vehicle. This means that in the event that Calculon inadvertently plans a path into a dead end, it can turn in place with a turning radius of approximately 23 in. and return to the last safe point before re-plotting a new course. Both path planning algorithms make use of this record to prevent plotting a course to the same dead end, trap, or pothole.

8.7 Waypoint Accuracy

Through the driver program's high level of control and the use of the Trimble DGPS, Calculon is able to reach DGPS waypoints within 1 meter. This meets and exceeds the requirements of the competition to reach waypoints within a 2 meter radius.

9. Analysis and Conclusions

9.1 Budget

Item	Company	Cost	Qty.	Total Cost	Our Cost
Ranger II Power Wheel Chair	Inva-Care	$3,200.00	1	$3,200.00	$0.00
SICK LMS Laser Range Finder	SICK-usa	$4,449.00	1	$4,449.00	$4,449.00
Sony DCRHC1000 3 CCD digital camera	Sony	$1,183.00	1	$1,183.00	$1,183.00
Desktop Replacment Computer	Alienware	$3,065.00	1	$3,065.00	$3,065.00
Trimble DSM132 DGPS receiver and antenna	Trimble	$4,050.00	1	$4,050.00	$1,000.00
Encoder Rings	Baumer Electric	$15.00	2	$30.00	$30.00
Encoder Readers	Baumer Electric	$89.00	2	$178.00	$356.00
Coral AHRS compass	Coral	$1,245.00	1	$1,245.00	$1,245.00
Power Inverter (24V to 110 AC)	VoltageConverters.com	$105.00	1	$105.00	$105.00
Power Converter (24V to 24V)	Vicor	$112.00	1	$112.00	$112.00
Power Converter (24V to 12V)	Vicor	$178.00	1	$178.00	$178.00
Wheels	Inva-Care	$124	2	$248	$496.00
Fiberglass Material	US Composites	$294.00	1	$294.00	$294.00
Machine Work	Geo Systems	$1,800.00	1	$1,800.00	$1,800.00
Circuit Board	4pcb.com	$48.00	1	$48.00	$48.00
Miscellaneous	various	$1,000.00	various	$1,000.00	$1,000.00
			Total	$21,185.00	$15,361.00

9.2 Safety Analysis

A necessary step of any entry into IGVC is an analysis of vehicle safety. IGVC vehicles, while sometimes small, can be over 100lbs in weight and are often driven by powerful, high-torque motors. No

design is complete without a proper analysis of the safety measures taken to guarantee that no person can or will be hurt during the operation of the vehicle. For Calculon, safety has been a primary consideration beginning in the pre-planning stages of design.

During construction, safety was considered for both the team and any persons that would encounter the vehicle. Electrical connections are kept safely within the electronics drawer. Batteries are charged only via the charging port on the outside of the vehicle and all connections for sensors and vehicle control are done using standard port interfaces such as DB9, as opposed to direct solders. This allows for easy and safe disassembly of the vehicle without worrying about electrical problems. In addition to this, all hazardous sharp metal edges were ground down to prevent cuts and scrapes.

In accordance with IGVC rules, Calculon includes multiple forms of emergency stopping. At the highest level of software, Calculon's 'lookout' program is capable of providing a measure of safety since its main function is to prevent the vehicle from hitting unforeseen objects. In testing, moving objects have been placed directly in Calculon's path and the 'lookout' program has successfully stopped the vehicle within 12 inches, thus avoiding damage. In normal operation, the software system is in constant contact with the wireless joystick used for remote driving. This wireless remote also has the ability to take control of the robot at any time or to emergency stop the vehicle. The vehicle also contains backup systems in the event that the software system is incapable of stopping the vehicle. On the outer casing of the vehicle is a large red emergency stop button. This button is wired directly to the power control system of the vehicle. When triggered, power is cut to motor controllers of the vehicle, which causes the motors to lock in place. This prevents the vehicle from rolling to a complete stop by acting as a mechanical brake. In addition to the software wireless stop, an electronic wireless switch is also connected to the power system identically to the emergency stop button. The range of the wireless emergency stop is up to 100 feet. This acts as another means to prevent any dangers to observers and surroundings.

9.3 Final Conclusions

Calculon is the combined effort of the hardworking members of the Robotics Laboratory at the University of Central Florida. We are confident that with its solid construction, reliable sensors and modular software systems, Calculon will be very successful in the upcoming 2005 IGVC competitive events. Our incremental design paradigm fit perfectly with our team. Members independently developed their corresponding portions of the project and by careful planning all parts came together successfully. It is our hope that the innovative features of Calculon, such as the 3CCD digital video camera and our unique Discover Vision Engine will make us stand out against the dozens of other competitors at this year's competition.