

-EL TORO-

2014-2015



Adrian Inghelterra
Benjamin Davis
Bryan Juarez
Erik Wagner
Gavin Fallen
Jangjujhar Singh
Jeff Sutherland
Karen Guerrero
Luis Garcia
Manpreet Sidhu
Melody Mojib
Michael Gaunt
Misael Toledo
Naimat Shifa
Philip Fadriquela
Samuel Gaxiola
Sidney Lim
Steven Moro

Faculty Advisor Statement

I certify that the engineering design of the vehicle described in this report, El Toro, has been significant and equivalent to each team member earning four semester hours of senior design credit for their work on this project.

Dr. C. T. Lin
Department of Mechanical Engineering
California State University, Northridge

1 INTRODUCTION

The Intelligent Ground Vehicle team of the College of Engineering and Computer Science at California State University, Northridge (CSUN) is proud to introduce El Toro. In the 2014-2015 academic year, a series of unique innovations have been implemented to provide a new level of intelligence to the robot. El Toro's main design intent was to be a low-profile vehicle, allowing for high traction and maneuverability. The design features a wide compartment for components with gullwing doors, allowing for maximum accessibility to crucial components. For increased reliability in any scenario, hardware redundancy was utilized in a design that features three laser range finders and two cameras. Algorithms designed to effectively utilize the added hardware have been implemented, creating a vehicle with remarkable performance in autonomous challenges. Innovations to this platform include dynamic fail-safes, an overhauled simulation environment, and a new system of course mapping. The mechanical design of El Toro began in the fall, and continued until it was completed in the spring, accommodating the rapidly changing requirements for each team.

2 ORGANIZATION

The intelligent ground vehicle (IGV) project at CSUN is a senior design experience that progresses over the course of two semesters in the Mechanical Engineering Department. Overall group leadership is decided democratically and includes a project manager, a technical manager and team leads, respectively. Team leads are responsible for organizing team meetings and events, as well as keeping open communication between the six sub-system groups. The six sub-system groups are: Mechanical, Power, Cognition, Vision, Navigation, and JAUS. Within each of the subsystems is a group lead, indicated in red in Figure 1. Each of the group leads is responsible for delegating tasks within their own group, as well as enforcing open communication with other groups.

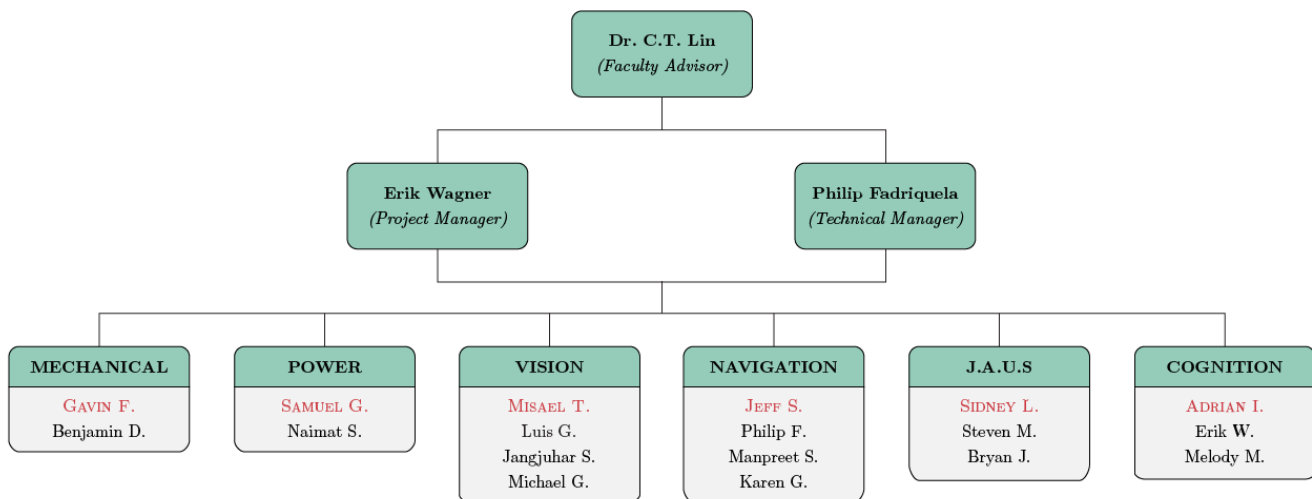


Figure 1: Intelligent Ground Vehicle (IGV) team organization chart for 2014/2015

3 MECHANICAL

3.1 Overview

The mechanical design of El Toro was focused on creating a low center of gravity on a completely modular robot, operating on a differential drive motor system. El Toro was designed with 80/20 T-slotted framing. This framing allowed for easy design alterations to be performed as the needs of the teams changed throughout the year. An octagonal chassis was utilized above the motors to create additional space for components as well as making them easy to access.

3.2 Frame Design ‡

Much as its predecessor did, El Toro utilizes 80/20 aluminum extrusions in the frame to allow for an interchangeable design. These extrusions contain several structurally sound mounting faces, which help the overall frame design. The frame contains very strong connections to the suspension, while also providing a strong base platform for the main housing of components. The octagonal design is built horizontally outward over the tires in order to maintain a low center of gravity as shown in Figure 2. The most significant benefit to using the 80/20 framing is the fact that no welding is required in order to assemble the frame itself and anything that attaches to it. The 80/20 aluminum extrusion members have a specific cross-section which not only improves its strength over standard square aluminum tubing, but ensures that attached parts remain secure. This specialized-cross section also allows for the ability to attach parts to any face on the extruded aluminum using the t-slot.

In order to construct the frame and attach brackets to it, a series of specialized nuts and bolts were used. Attaching components to the aluminum extrusions required the use of three major parts: a $1/4 - 20 \times 3/8$ " bolt, a bracket (or the component itself), and a t-nut. An example of how these parts work together to secure a part to the aluminum is shown in Figure 3.

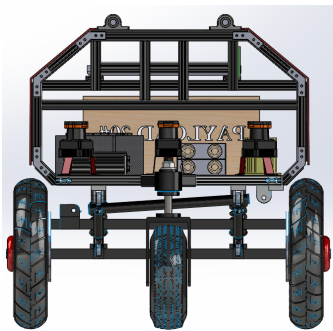


Figure 2: Illustration of radial polar plot path planning

Parts that were attached to the frame itself include the floor boards, laptop tray frame, fish plate, solid strut brackets, spring brackets, and the caster housing. All of the aforementioned parts were attached using the same methods described above.

3.3 Suspension

The suspension as a whole has been carried over from El Toro's predecessor, VADER, with changes made to allow it to be utilized on the new vehicle. The previous suspension design proved to be adequate, whereas complete mechanical overhauls were needed on other parts of the vehicle. The suspension is broken into two main components; the lower suspension and the caster wheel suspension. The lower suspension consists of two front ridged struts, and two rear spring and damper shocks. The combination of these prevents unwanted sway in all directions, along with dampening inputs from the drive wheels. These members are the connection points to the upper body frame, as shown in Figure 3.

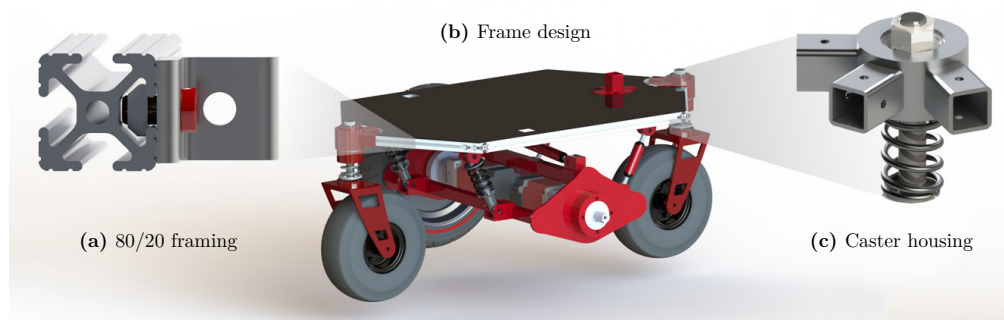


Figure 3: El Toro's platform built with 80/20 slotted framing utilizing a caster suspension system

‡ Indicates section(s) containing new innovations.

The caster wheel suspension is an innovative design used to incorporate suspension into the caster wheel housing blossomed into multiple benefits throughout the rest of the overall design concept. This feature gave a lower mounting surface for the front laser range finder (LRF), doubled the surface area to mount components on the body frame, and kept true to the original low-profile design. The caster aluminum housings are the connection points for the triangular front and rear 80/20 aluminum framing, as shown in Figure 3c. and allows for rotational movement of the caster wheel. The new spring-incorporated design also allows for z-axis (i.e., vertical) translation.

3.4 Mast ‡

The primary focus for the mast design was to provide a sturdy support system to allow for a non-oscillatory vision feed. To do this, a length HT Series, hollow aluminum extrusion was used at the primary mast extrusion. This hollow extrusion allowed for a myriad of connections to supporting members. To reduce vibrations in the lateral direction, an aluminum plate was machined that bolts the mast to the frame of the vehicle. Similarly, to prevent unnecessary low-frequency vibrations in the forward and backward direction, a t-slotted 80/20 member was attached at a pinned hinge at the rear of the vehicle. This rigid extrusion connects to the top of the mast and permits very little low-frequency vibration which is the main source of concern for image processing. Because of the flexibility that 80/20 allows for, the mast can still very easily be removed whenever necessary, such as for transport, as shown in Figure 4.

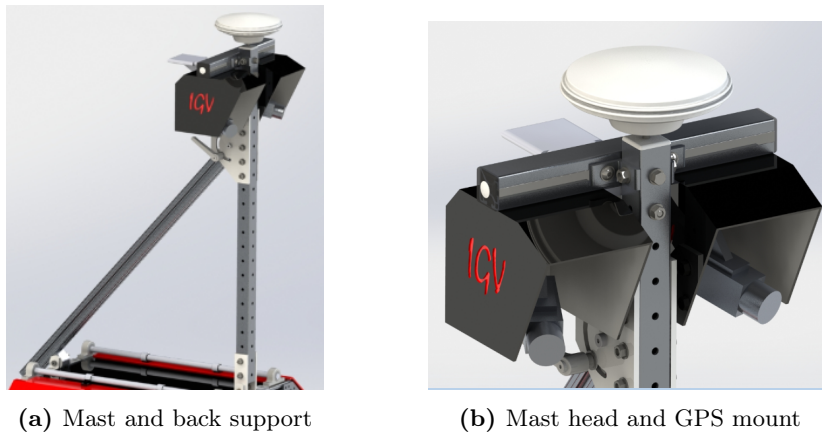


Figure 4: Mechanical components used for the mast.

At the top of the mast lies a single horizontal 80/20 two slotted extrusion, designed to hold the two camera and camera mounts in place. The HT Series mast was predominantly selected due to its pre-machined hole pattern along the extrusion. This member allows for single bolt mounting capabilities that can be adjusted incrementally at various heights. Similarly, the cameras can be slid along the extrusion to change the distance between the cameras to find an optimal field of view. Furthermore, inside the top of the mast lies a threaded aluminum block. This threaded block is bolted to the inside of the mast extrusion and holds the GPS antenna safely to the highest vantage point of the robot where there is the least interference for the signal.

3.5 Gearboxes & Motors

The gearbox and motors are in line with each other, mounted underneath the aforementioned frame. This layout was selected to provide a differential drive to El Toro, allowing the motors to react independently from each other. This arrangement makes for a very maneuverable platform.

A torque analysis was performed in order to determine the amount of torque that is required to allow the vehicle to move without straining the motors. Calculations were performed using the assumptions provided in Table 1.

‡ Indicates section(s) containing new innovations.

Table 1: Parameters used for torque analysis of the gearboxes and motors.

Parameter	Value
Weight of vehicle	210 lb
Coefficient of friction	0.35
Number of motors	2
Wheel diameter	15 in

Newton’s second law of motion along with a free body diagram was needed in order to calculate the forces. When calculating the torque required for El Toro, a worst-case calculation was conducted; that is, the analysis of the robot on an inclined plane.

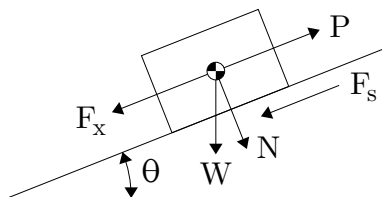


Figure 5: Free body diagram used for torque analysis of the gearboxes and motors

The IGVC rules state that there could be a maximum angle of incline of 15°. Taking this into consideration, a free body diagram was used to derive the corresponding force equations, as shown in Figure 5. Using the known angle of incline, θ , weight of the robot, and the coefficient of friction for a damp grassy surface, the forces and torque in question were calculated. The calculated values ensured that the selected motors would be sufficient for El Toro to successfully navigate through all field conditions.

4 POWER

4.1 Overview

The goal of El Toro’s power system is to consistently provide an adequate amount of power to every component used by the vehicle. The entirety of the robot is powered through the use of a single power source, as show in Figure 6.

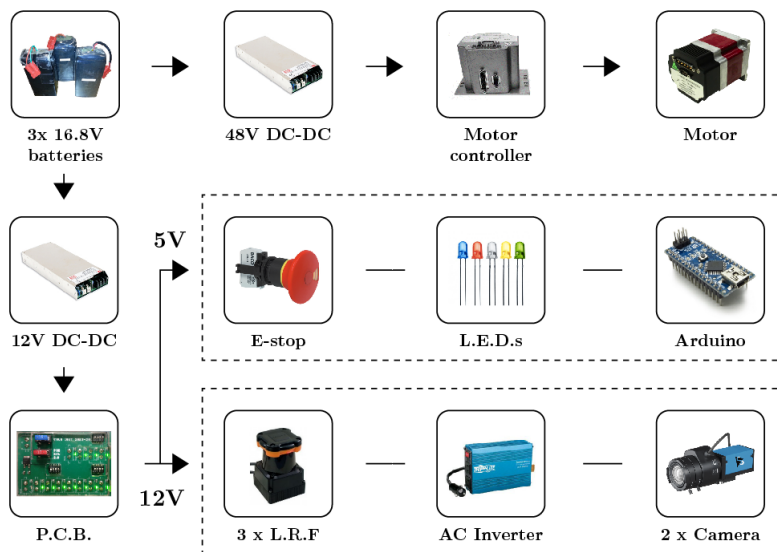


Figure 6: Power distribution map with indicated voltage rails.

4.2 Batteries

El Toro utilizes three 4-cell packs of Lithium Po batteries that provide power to the system. These were selected by determining the power usage in normal situations and extreme situations. In the following table, normal conditions were tested at approximately 1 mph, whereas the extreme load condition was tested at 6.8 mph was used for extreme conditions.

Table 2: Power consumption for normal and extreme conditions in Watts

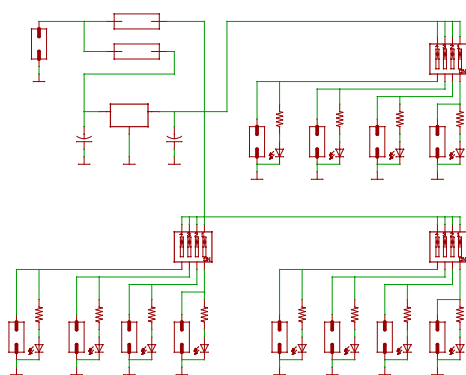
Type of Load	Normal Load	Extreme Load
Total base load	177	177
Transient motor load	154	910
Total load	331	1087
Total at 88% DC/DC efficiency	376	1235

It was determined that El Toro needs to be provided with at least 1235 W, as seen in Table 2. There were multiple options that could have been used to provide this power, but ultimately LiPo batteries were decided upon because of their power-to-weight ratio compared than their Li-ion counterparts. Each of the battery packs that has been assembled has a nominal voltage of 14.8V and a 30A output. The rated power output of each battery is 444.4W, which with three, can provide well over the 1235W that are necessary. The battery voltage is regulated by two separate DC-to-DC converters; a 48V DC output for the motors, and a 12V DC output for most of the other electrical components.

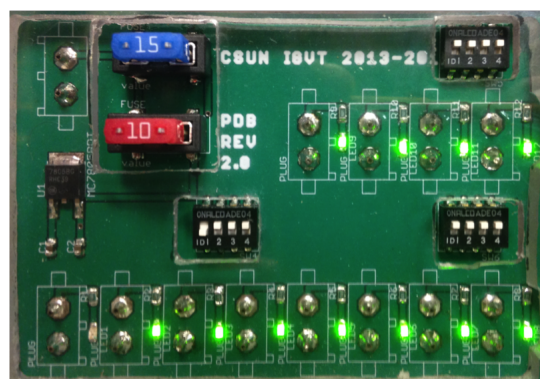
4.3 Power Distribution Board

The power system for El Toro is accomplished through the use of a printed circuit board (PCB). To try to keep the PCB as small as possible, the 48V rail is kept separate from the board. A relatively minimalistic design was implemented using EAGLE CAD to produce the required Gerber files, as shown in Figure 7a.

To keep the board small, fuses were added to entire rails, rather than individual lines. As long as the fuse is changed based on the components in the rail, having only a fuse for each rail is adequate. A switch was dedicated for each component, which was to remain on unless in the event of a short circuit. If this occurred, all switches would be opened, the fuses replaced, and each switch turned on one-by-one until the source of the short circuit was determined. Reducing the number of fuses present on the board, along with the removal of the 48V traces and a section for the voltmeter, allowed for an overall reduction in area of 70%, as shown in Figure 7b. The switches to send power to the board were designed on a separate acrylic board, along with the switch connecting the 48V to the DC motors.



(a) EAGLE CAD circuit schematic



(b) Actual power distribution board on the robot

Figure 7: New power distribution board with 70% reduction in area

4.4 Emergency Stop

The emergency stop system on El Toro functions by making use of the motor controllers' built-in stop conditions to bring the robot to a safe and controlled stop. When the emergency stop is not activated, a logic LOW (i.e., ground) is sent to the motor controllers. When the emergency stop is activated, a logic HIGH (+5V) is sent to the motor controllers' input/output pins. A series configuration is used for the push-button stop and the wireless remote stop to eliminate any confliction between the two.

4.5 Hot-Swap

The hot-swap system was implemented in order to reduce down-time of the vehicle while testing and at competition. Hot-swapping allows the robot to have the ability to plug in fully-charged batteries to the system before removing the depleted ones. This method of switching batteries makes it possible to keep all of the components on the robot powered on, thus eliminating the need to re-initialize all of the hardware and software. Each initialization can take up to 15 minutes, which cannot be spared at competition.

5 VISION

5.1 Overview

Vision is important to El Toro because of its ability to gather information about the surroundings, including identifying obstacles such as white lines and potholes. El Toro achieves this through the use of two industrial cameras that undergo various filtering processes, such as color plane extractions and thresholding. The filtered image is then run through particle filtering, where noise and other unwanted particles are removed, before sending the refined image to System Integration.

5.2 Hardware Redundancy ‡

El Toro employs two Imaging Source industrial cameras each with a resolution of 640x480 to acquire images. Each camera is equipped with a wide angle lens, from Theia, which provides a horizontal field of view of 129°. The two cameras are then mounted so that an effective horizontal field of view is 180° is achieved. This allows El Toro to see at least five feet from its center at both 0° and 180°.

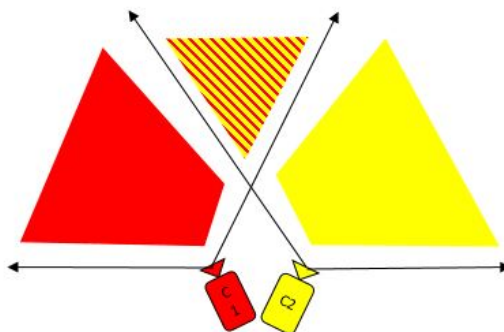


Figure 8: The overlapping view of two cameras

There are several advantages of using two cameras versus a single camera with a fisheye lens to obtain a 180° field of view. One benefit from this configuration is the introduction of sensor redundancy. The two cameras overlap in the middle hatched region, as shown in Figure 8). This redundancy of the data in the overlapping region minimizes the possibility of passing erroneous obstacles to systems integration. In addition, the cameras are mounted in different locations on the mast head and thus the perspective of the competition field is different for each one. In the case of a camera being unable to detect an obstacle due to its position, the other camera (with a different perspective) is meant to act as a fail-safe and detect the obstacle.

‡ Indicates section(s) containing new innovations.

5.3 Niblack Thresholding ‡

The improved detection of objects of interest is accomplished through the use of adaptive thresholding, specifically Niblack thresholding. An adaptive threshold differs from a global threshold by one key aspect: a global threshold applies a single, fixed range of values across the entire image for which to threshold by, whereas an adaptive threshold scans the image in small segments and determines an optimal range of values for the given area. The Niblack thresholding is utilized in various ways, which include vertical Niblack, horizontal Niblack, and square Niblack, as shown in Figure 9.

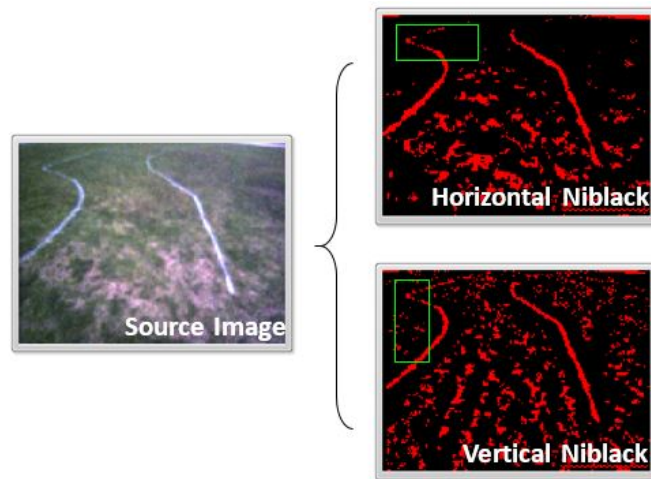
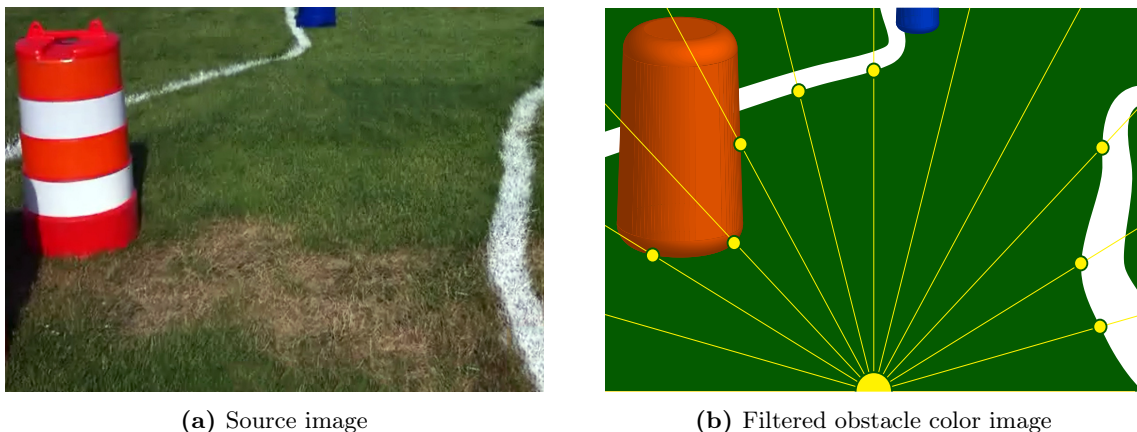


Figure 9: Example of Niblack thresholding

Each of these methods designate a different method in which the image is discretized into smaller pieces before being analyzed. Depending on the given light conditions, each has the ability to detect different pieces of the image effectively. Using these methods in unison allows for repeatable accuracy, regardless of the given lighting conditions.

5.4 Closest-Edge Algorithm

Because of the increased amount of image processing required with the addition of a second camera, efficiency of the process plays a large role. To help facilitate this, the closest-edge algorithm was developed. Rather than looking for data in every pixel of the image, closest-edge detection looks for the nearest edge of an obstacle, and then stops processing at this point. This is accomplished through the use of LabVIEW's IMAQ Edge Tool, which starts at the robot's position and extends lines every 1° from $0-180^\circ$ in front of the robot.



(a) Source image

(b) Filtered obstacle color image

Figure 10: Example of closest edge algorithm

‡ Indicates section(s) containing new innovations.

Once the line detects an edge of an obstacle, or the edge of the image, it stores the coordinate's location, and begins checking the next angle. While the rest of the image may provide useful data for the future, the vehicle is unable to store information for future iterations, so only current obstacle information is useful.

6 SYSTEM INTEGRATION

6.1 Overview

System integration, which is handled by the Cognition team, is responsible for interpreting all of the data gathered by the robot's physical sensors in order to detect and achieve a safe path through the course. This means that information gathered by the Vision, Navigation, and Cognition teams are received in the form of a polar plots, covering the 180° in front of the robot. The polar plots have a resolution of one degree, where they will pass the distance of the nearest obstacle to the robot. This information allows the robot the opportunity to effectively avoid all obstacles, while traveling to the required GPS waypoints. The system integration algorithms are based on the same concepts as used by its predecessors, with major overhaul in the actual algorithms used to achieve the concepts.

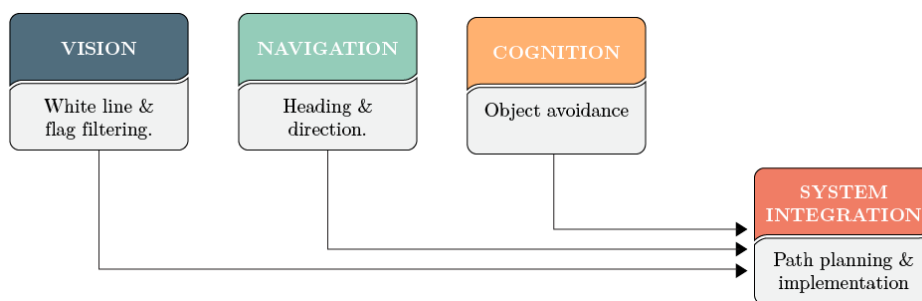


Figure 11: Integrating Systems

6.2 Sensor Redundancy‡

Cognition is responsible for supplying information about discrete obstacles in the vehicles path. Through the use of three LRFs, obstacles in a 240° horizontal field of view can be detected and passed to System Integration, as shown in Figure 12.

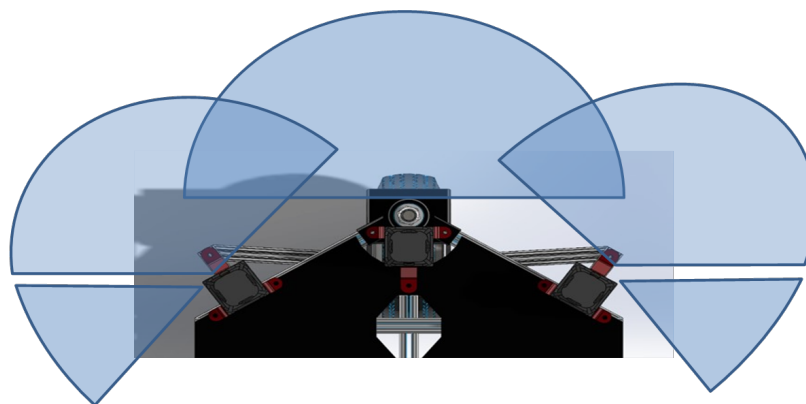


Figure 12: The angles covered by the LRFs

The addition of sensor redundancy has a multitude of benefits, including protection of potential hardware failure as well as increasing the field of view. When objects are outside of the 180° directly in front of the robot, they can be detected on the right or left side of the robot. Objects detected next to the robot are simply used to alert the path planning of a possible obstacle near the side of the vehicle. These are preventative measures to ensure that the

‡ Indicates section(s) containing new innovations.

vehicle does not clip barrels as it decides to turn sharply in either direction

6.3 Radial Polar Plot

The path planning of El Toro is achieved through a proprietary method known as the radial polar plot (RPP). The goal of RPP is create safe paths for El Toro by filtering through all the data received, keeping only the most necessary pieces. Initially, polar plot with data corresponding to the locations of obstacles are received from vision and the LRFs. Once the data is received, RPP first interprets the data and identifies groups of objects, which is any set of objects that are close enough to each other that the robot will not be able to pass in between them. Once the objects are grouped, they are then inspected for possible openings, which is simply any gap in obstacles that is large enough for the robot to travel between.

6.4 Non-Sequential Openings ‡

Openings are defined by the edges of the last point of an obstacle in one group, followed by the first point of an obstacle in a separate group. To improve the intelligence of the vehicle, a new method, called non-sequential openings was developed, which not only utilizes the openings between two obstacle groups next to each other, but also utilizes the openings created by obstacle groups further away from each other. This method is designed to give the vehicle more choices, so that all possible methods of travel are exhausted before the robot is forced to do a zero-radius turn. An opening is selected through various parameters, with the ultimate goal to let the vehicle achieve the most direct path to the GPS waypoint. To have the robot move through this opening, a radial path is taken to a desired location on the opening. Because of the complexity of certain situations, openings are divided into vertical openings and lateral openings, which designate the path that will be taken to enter them. Vertical openings may be much tough to enter with a single radius, so a method known as double-radius was developed.

6.5 Double Radius ‡

The double-radius method gives the robot the ability to choose a path that resembles a second order spline, which means the first radius is the positioning radius, while the second radius is the achieving radius. The double-radius method is utilized when the achieving radius is unable to safely enter an opening with one path. A radius is determined to put the vehicle in a position to make its next path easier. The positioning radius generally takes the robot in a direction away from the opening, which will position the robot in such a way that the achieving radius can effectively utilize the information. An example of a scenario where the double-radius method could be utilized is seen in Figure 13.

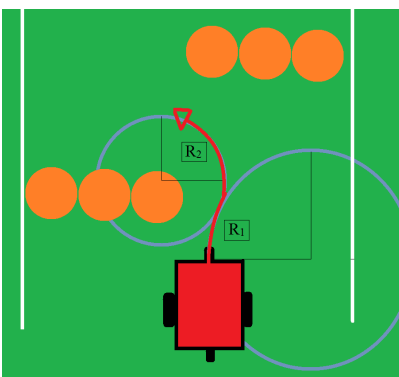


Figure 13: Example situation for a double radius

6.6 Obstacle Path Interference ‡

Once a radial path is chosen, the possible path is sent to OPI, or obstacle path interference. The goal of OPI is to eliminate the possibility of taking any path that would lead the vehicle into an obstacle. If a chosen radial path

‡ Indicates section(s) containing new innovations.

were to send the vehicle into an obstacle, the next possible radial path is chosen. This is accomplished by analyzing a possible radius, and accounting for the width of the robot, which creates a polygonal area that is the path that the robot would take to achieve the radius. This polygonal area is then overlaid on the polar plot containing information about obstacles, and if there are no obstacles that intersect the polygonal area, the path is deemed safe. In Figure 14, the red polygonal area represent the possible path that El Toro would take, while the white semi-circles represent obstacles.

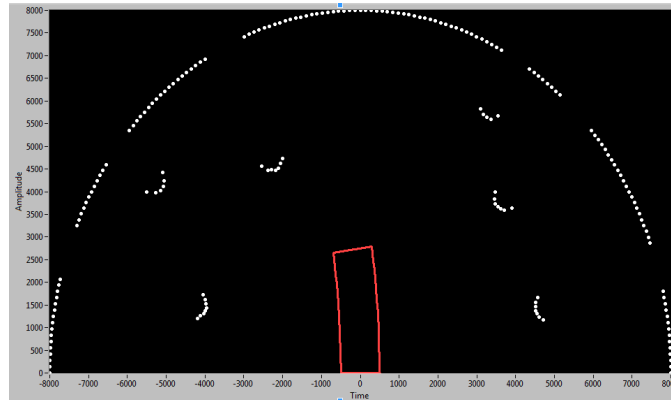


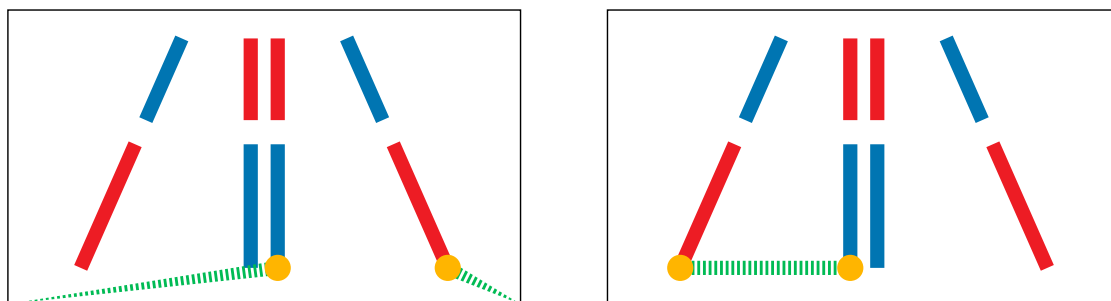
Figure 14: Chosen path as shown in OPI

6.7 Dead End ‡

In the case where a radial path cannot be chosen, a dead end scenario is triggered. This forces the robot to complete a zero radius turn in the direction of the last opening that it was attempting to enter. It will keep turning until it sees an opening, at which point, the robot switches back into normal path planning routines. To avoid the possibility of the robot consistently getting stuck in the same dead end location, GPS coordinates are recorded at each instance of dead end being triggered. These GPS coordinates are then used to tell the vehicle to turn the opposite direction that it turned previously, in hopes of letting the robot find a new way out of the situation.

6.8 Ghost Lines

To identify the red and blue flags on the course and maintain the proper path when traveling through them, the ghost lines algorithm was developed. The term "ghost lines" refers to a process wherein simulated blockages are intelligently inserted into the robot's sensory data to force it into the desired course through the flag sections. This algorithm works by identifying the nearest red and blue flags and comparing the locations of these flags. If the nearest flag on the left of the robot is red, a "ghost line" is drawn across the area between these flags, as shown in Figure 15a). This line will force the robot into the other section of flags, presumably, the correct one. The algorithm will then identify the correct configuration of flags, and will allow the robot to pass through the opening.



(a) Case 1: select edges in correct orientation

(b) Case 2: select edges in incorrect orientation

Figure 15: Illustration of ghost lines effect.

‡ Indicates section(s) containing new innovations.

7 NAVIGATION

7.1 Overview

The navigation system allows the robot to obtain information about the current vehicle position in its local and global environment. This is accomplished through the use of a NovAtel Pinwheel GPS antenna, a NovAtel Propak V3 Receiver, and a Sparten AHRS-8 Module. The use of these devices gives the robot the ability to locate waypoints to which the robot needs to travel, which are then used to calculate a desired heading. This desired heading will then give the vehicle a goal orientation, which will be the direction of the shortest distance to that waypoint. From there, the information pulled from the GPS and compass are sent to system integration to ensure the robot has the ability to achieve the desired point.



Figure 16: Navigation flowchart [from left to right: GPS antenna, GPS receiver, computer, compass]

7.2 Intermediate Waypoints

Intermediate waypoints is a method in which the vehicle is able to record a trail of locations that the robot had traveled to on the course during a specific run. A frequency is designated before the run that determines how often the GPS location should be recorded. This breadcrumb trail can then be used in conjunction with a new mapping algorithm to provide feedback about the robot's performance.

7.2.1 Mapping ‡

The mapping program records the polar plots of obstacle data at set frequencies, which are later converted to GPS waypoints. This data provides a plot containing points at the edges of all of the obstacles that were seen while on the course. When the breadcrumb trail is overlaid onto this plot, a map with the data corresponding to the robot's location while navigating through obstacles is clearly seen. This allows for information on failures of the last run to be interpreted and inserted into the interactive waypoints program.

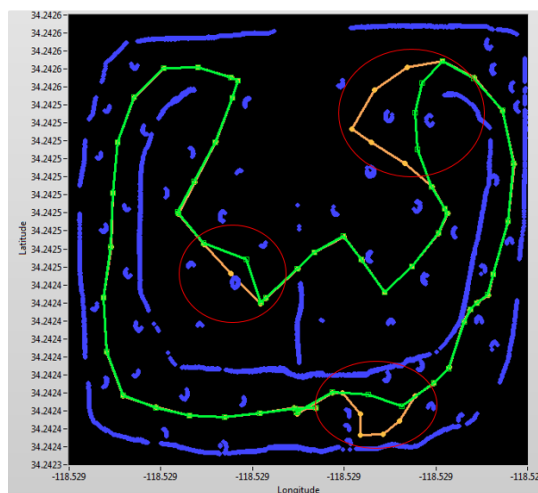


Figure 17: Example of a course map with paths

‡ Indicates section(s) containing new innovations.

Figure 17 shows the path of the robot and the data from the mapping program during one of the robot's runs. The white line and barrel data is extracted for use in the plots, which is indicated by the dark blue lines. The original path of the robot is represented by the orange line. There are a few instances where the robot contacted a barrel or took an inefficient path, which is indicated by the red circles. As seen in Figure 17, the areas where the robot previously had encountered issues have been fixed by adjusting the robot's path, as seen by the green line.

8 SIMULATION ‡

8.1 Overview

A simulation program was developed using LabVIEW in order to test navigation and system integration algorithms without the physical platform. The simulation was developed as a method to allow testing of new codes without endangering the vehicle with a previously untested code, which may have bugs that create unsafe conditions for El Toro. The simulation also provides a testing environment that can be used any time, regardless of whether the El Toro platform is being worked on mechanically, or is being used by a different part of the team. Virtual LRF data is created, while inducing specified levels of Gaussian white noise to more realistically represent the stream of data that would come from the sensors. This allows the vehicle to choose different paths each time it navigates through the simulation. The simulated data gathered by the LRF and compass is passed to the navigation and system integration code, allowing the vehicle to run autonomously.

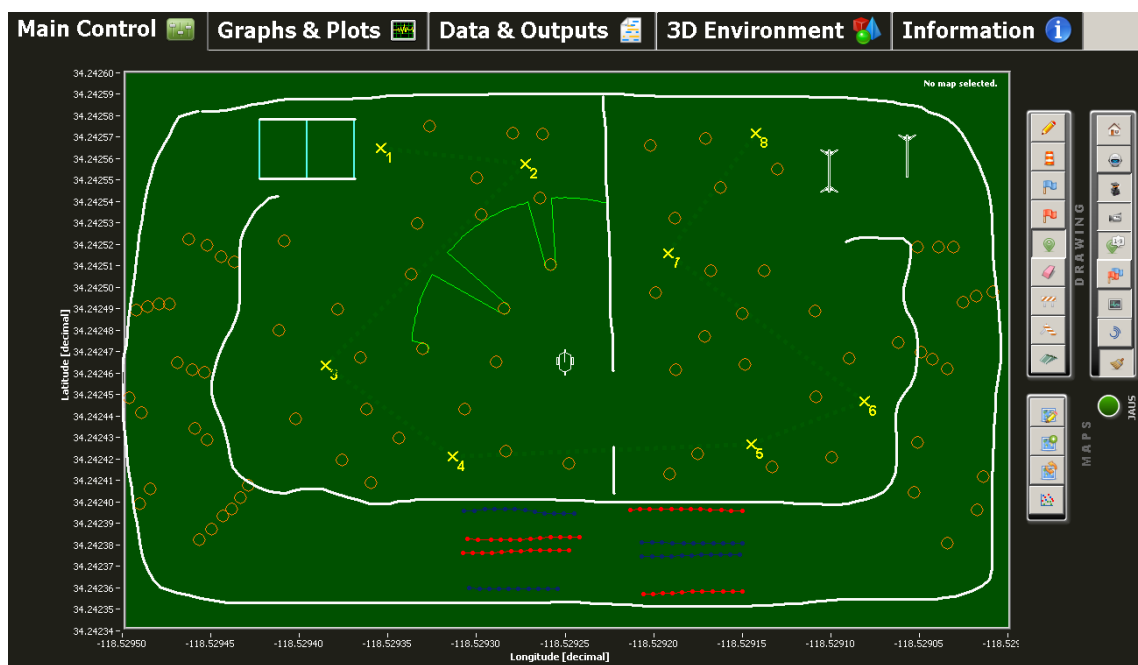


Figure 18: Computer-generated IGVC map used for simulation

8.2 Simulation Course

To allow quick access to a simulated situation, different obstacle types can easily be added with the click of a mouse. The different obstacle types that can be simulated are barrels, white lines, fallen sawhorses, red and blue flags, and ramps. In the simulation, a course can be drawn, full of white lines, barrels, and other obstacles, along with waypoints that need to be achieved, as seen in Figure 18. Along with autonomous mode, there is also a manual driving mode. If a particular scenario is seen as an issue, the vehicle can be easily positioned in the same location before allowing it to autonomously navigate through the problem area again.

‡ Indicates section(s) containing new innovations.

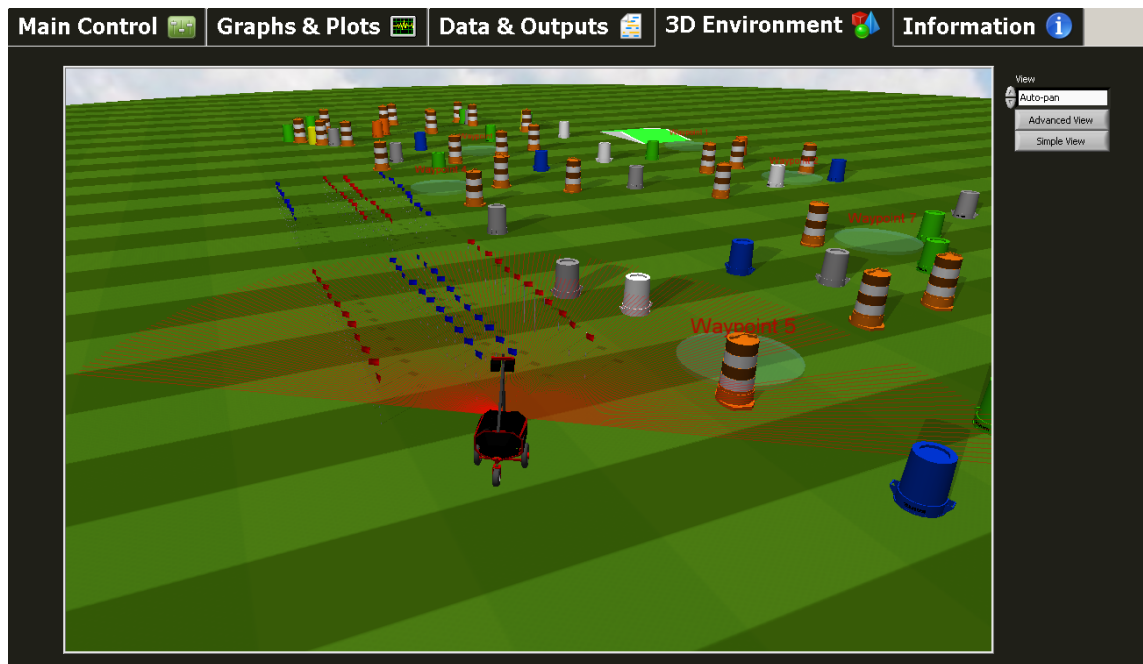


Figure 19: Computer-generated IGVC map used for simulation in 3D

To aid in a more realistic view of the course, a 3D environment was also created that accurately depicts many of the obstacle types at competition, along with a true model of the vehicle. This allows users to have an easy understanding of what the vehicle is seeing when it encounters a situation, which allows for easier debugging.

9 JAUS‡

9.1 Overview

A new strategy in the JAUS team was to adopt a new architecture to run their systems. The predecessor used the JAUS toolset (JTS) to create skeleton code that was used to complete the interoperability tasks. However, the performance tasks were never implemented. This results from interfacing difficulties faced when trying to communicate between LabVIEW and C++. El Toro is primarily written in LabVIEW whereas the main JAUS code is in C++ (JTS code). Multiple options were explored such as shared memory allocation; however, the use of user datagram protocol (UDP) was adopted during this iteration.

In order to overcome this issue, a C++ header file was created that allows for communication between the C++ and LabVIEW code through the use of UDP sockets. Once this problem was overcome, the new architecture was introduced that parses data received from the JTS code and determines what the robot needs to perform. The header file allows the C++ code to write to one port and read from another port. Then, the LabVIEW code reads from the port where the C++ code has been written to, and vice versa.

The LabVIEW code utilizes a state machine architecture that consists of three states: parse, send, and receive. The program is defaulted to the receive state upon startup until it acquires data from the JTS code. After execution, it will enter the parse state where it will determine the message being sent and parse through the message. It then sends or receives corresponding data from the GPS, compass, or motors. Next, the state machine switches to the send state where it will deliver a message back to the JTS. Finally, it returns to the receive state until another message is queued for reading. A diagram of the state machine, as well as the path of communication, is shown in Figure 20.

‡ Indicates section(s) containing new innovations.

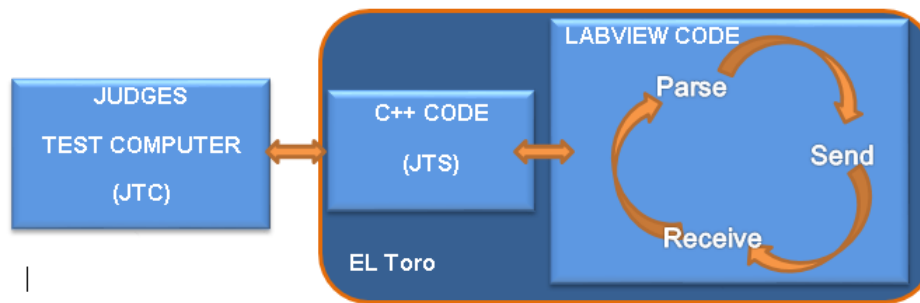


Figure 20: JAUS flowchart

10 OVERALL SYSTEM PERFORMANCE

Some key performance attributes of El Toro can be seen in Table 3. A low-profile, design coupled with the use of three LiPo batteries in series allows El Toro to maintain a top speed of 6.8 mph for approximately 3 hours while running all components. It takes El Toro an average of 68 ms to decide on a path from the time that it first recognizes obstacles. Under normal operating conditions, battery life is roughly 4.5 hours. All three LRFs detect obstacles up to 8 meters away. Vision utilizes the cameras to detect obstacles and lines up to 5 meters away. Lastly, through the use of a differential GPS system (DGPS), the waypoint resolution is approximately 30 cm.

Table 3: Overall system performance parameters

Parameter	Value
Top speed	6.8 mph
Reaction time	68 ms
Battery life	3 - 5 hrs.
Obstacle detection distance	8 m
Waypoint resolution	30 cm

11 TOTAL COST ESTIMATE

The total retail cost of all the materials and components utilized on El Toro is approximately \$33,000. The most expensive components contributing to the cost were the GPS antenna, the GPS receiver, as well as the LRFs. This year, very few items needed to be purchased, so construction of El Toro cost approximately \$2,400, as provided in Table 4. In addition to the cost of parts, each of the students spent approximately 15 hours per week on the IGV project. Thus, during the 2014-2015 academic year, each of the students spent approximately 600 total hours working on El Toro.

Table 4: Total cost estimate for 2014/2015

Group	Item	Reused Cost	New Cost
Mechanical	Motors	\$2,250	-
	Motor controllers	\$1,030	-
	Gear boxes	\$2,000	-
	80/20 aluminum	\$60	\$250
	Lower chasis	\$250	-
	De-couplers	\$60	-
	Casters	\$100	\$100
	Wheels	\$100	\$100
	Springs	\$150	\$150
Cognition	Left LRF	\$6,000	-
	Middle LRF	-	\$6000
	Right LRF	\$6,000	-
Navigation	GPS antenna	\$2,000	-
	GPS receiver	\$3,500	-
Vision	Imaging Source camera	-	\$1,000
	Theia lens	-	\$500
	Arduino Nano	\$30	-
Power	DC-DC converters	\$510	-
	PCB	\$100	-
	Miscellaneous parts	\$50	-
	LiPo batteries	\$600	\$300
TOTAL		\$24,790	\$8,400