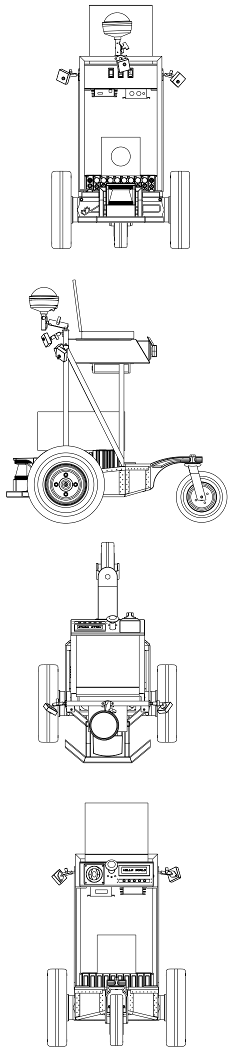
2006 URSA Minor Design Report

14th Intelligent Ground Vehicle Competition



ETLC
ETLC ENGINEERING TEACHING AND LEARNING COMPLEX
MAIER[illegible]RNING CENT[illegible]
Trimble

ARVP
autonomous robotic vehicle project

Autonomous Robotic Vehicle Project
4-9 Mechanical Engineering Building
University of Alberta
Edmonton, CANADA T6G 2G8
ph: 780-492-9440
fax: 780-492-2200
web: http://arvp.org

1. Introduction

The Autonomous Robotic Vehicle Project (ARVP) from the University of Alberta draws on the experience and success of nearly ten years of robot design and fabrication with the advent of its latest vehicle URSA Minor. This report outlines the design considerations and the design-build process undertaken to realize this new platform. Descriptions of the mechanical, electrical, and software systems are presented with an emphasis on safety, reliability, and durability. The combination of the URSA Minor platform, more sophisticated sensor use, and improved software intelligence make the ARVP the team to beat at the 2006 IGVC.

2006 ARVP highlights
proven chassis and reliable drivetrain
full system status monitoring with datalogging
new guidance system with environment mapping

2. Back-To-Basics Design

In 2004-2005, the ARVP set the ambitious task of designing and building a wholly original vehicle with a novel four-wheel drive and passively articulated steering system. Through parallel design and construction processes that involved nine separate machine shops, the core vehicle systems were complete within nine months. This vehicle was registered for the 2005 IGVC under the URSA nameplate but as the competition drew near it was apparent that its lack of reliability did not justify the expense of traveling to Michigan. Given the previous eight consecutive appearances by the ARVP at the IGVC, the decision to not compete in 2005 was difficult for the team.

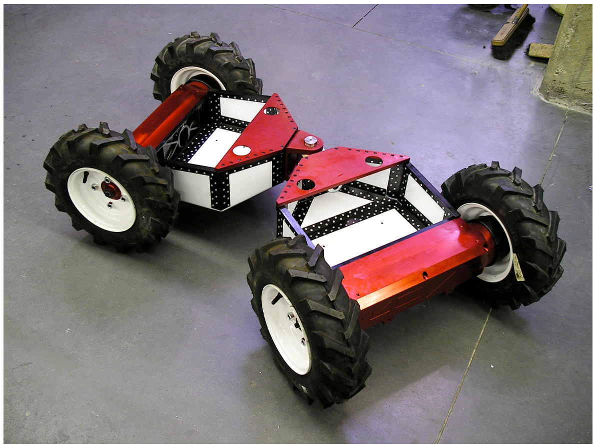


Figure 1: Completed URSA mechanical platform

Part of URSA's shortcomings were attributed to insufficient communication between the ARVP's Mechanical, Electrical, and Software sub-teams. As a result, in addition to regular team-wide General Meetings, weekly Build Sessions were created to establish a set time when members of all sub-teams would be available for discussion and support purposes. To accommodate the Build Sessions, the team's common work environment was improved with the addition of tools, workspace, and computer design workstations. Finally, to further facilitate access to information and project documentation, a user-editable website (wiki) was created to complement the existing ARVP-administered e-mail and file server.

2.1. Goal and constraint definition

Following the experience with URSA, the ARVP moved to reestablish its goals and constraints for a successful IGVC platform. As always, vehicle development is regulated by two customers: the IGVC (external) and the ARVP (internal). The IGVC imposes dimensional, payload, and climbing requirements and a maximum speed.

The ARVP adds constraints for practical purposes. Weight is limited to facilitate lifting, indoor-outdoor operation is necessitated by the harsh Alberta climate (with a width restriction for doorways), and the ease of vehicle disassembly is considered for airline transport. In addition, the design should be able to support a human payload that has been found to be an inevitable part of a student vehicle project. Beyond these requirements, the ARVP set two important goals as a direct result of URSA's shortcomings: improve overall drivetrain reliability and enhance the vehicle's user interface to facilitate testing for software development.

External	Internal
length: at least 3', at most 9'	weight: less than 250 lb
width: less than 5'	indoor-outdoor capable
height: less than 6'	width: less than 30"
maximum speed: 5 mph	breakdown for airline checked baggage
payload capacity: 20 lb	support human payload
climbing ability: 15% grade	endurance: 150 minutes continuous
	improve overall drivetrain reliability
	enhance user interface for testing facilitation

Table 1: External and internal customer demands

2.2. Concept feasibility

While URSA's chassis was found to be sound, it exhibited drivetrain problems characterized by overheating motors. Despite the experience of the ARVP, the combination of unique steering dynamics, unfamiliar motors, and new custom motor drivers made diagnosing the root cause of overheating difficult. It was thus decided to simplify the problem by eliminating some unknowns. Steering would revert to a differential steer variety and a commercial off-the-shelf (COTS) motor driver unit would be acquired while other hardware would remain constant. Efforts would also be made to enable system status monitoring of quantities such as motor temperature and current flow. With these simplifications and this additional information, the team hoped to create a reliable platform that would allow for a smooth eventual transition to more advanced steering methods.

2.3. Design and build process

This transition vehicle was dubbed URSA Minor and makes use of proven surplus parts manufactured in 2004-2005 for URSA while leaving that platform intact. A caster-follower differential steer design was selected for its simplicity and previous IGVC success.

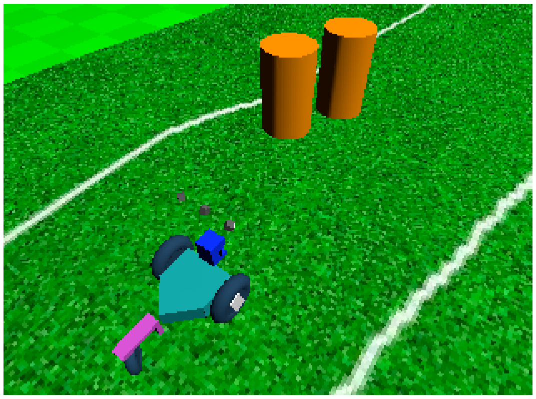


Figure 2: Player/Stage/Gazebo environment

The Player/Stage/Gazebo simulation environment (http://playerstage.sourceforge.net) shown in Figure 2 was exploited to produce a simple 3D model of the vehicle with dynamics. This simulation complements engineering calculations by each of the Mechanical, Electrical, and Software teams in the establishment of geometries and the selection of components to ensure desired performance. By linking this simulator to the actual onboard software, control system and other software development can be done using realistic sensor models also provided by Gazebo.

As with URSA, this simulation proof of concept was followed by extensive use of PTC's Pro/Engineer for mechanical part and assembly modeling and engineering drawing production while Pro/Mechanica enabled the optimization and verification of components and assemblies via finite element methods. Altium's Protel was also used for printed circuit board design of necessary electronics modifications. Before manufacturing began, a complete set of system drawings was assembled for the critical review of peers and faculty. While this process lengthens design time, it promotes a successful final assembly and facilitates the parallel involvement of multiple fabricators in the manufacturing phase.

Given this approach, URSA Minor was able to demonstrate IGVC qualification requirements and was ready for on-vehicle software development and testing by March 2006.

3. MECHANICAL SYSTEM

URSA Minor's mechanical systems comprise two major roles: a rolling platform and support systems for sensor mounting and component enclosures. Customer demands are of great concern at this level as the mechanical systems provide the basis for the remainder of the vehicle.

URSA Minor Specifications	
outside dimensions (l x w x h)	45.5" x 30.0" x 40.3" (1.15 m x 0.76 m x 1.02 m)
weight	240 lb
weight distribution (front/rear)	75%/25%
clearance	5.5" (0.14 m)
turning radius	zero
top speed	4.92 mph (7.92 kph)
maximum grade	23°
payload capacity	20 lb
battery life (continuous)	180 minutes

3.1. Platform

3.1.1. Chassis

URSA Minor uses the basic chassis building blocks developed for URSA. Where URSA consists of two identical tubs connected by a two-DOF joint, URSA Minor uses a single tub and a custom caster pivot. The tub provides a rigid yet lightweight vehicle base and battery housing due to its riveted aluminum honeycomb composite panel construction. A CAD model rendering of URSA Minor is shown in Figure 3.

3.1.2. Drivetrain and steering

A motor housing is attached to the tub and contains two 24V permanent magnet motors coupled to planetary gearboxes with 25:1 reduction. An assembly with a 16.5" diameter tire, wheel, and hub is threaded into each gearbox shaft. This configuration features a low part count and quick access for service and provides adequate torque for climbing a 23° incline while carrying a 20 lb payload. At the same time, speed is not sacrificed as the vehicle is capable of a straight and level top speed of 4.92 mph (2.2 m/s).

Vehicle steering is accomplished by regulating the speed of each wheel via proportional, integral, derivative (PID) control based on wheel encoder feedback. Despite a tripod stance, stability is assured by a low centre of mass (12.5” above the ground) and a 75/25 front-to-rear weight distribution. A large 10” pneumatic caster is also chosen to prevent hangups on course obstacles such as ramp lips.

The motor overheating issue is addressed in URSA Minor with the addition of two fans to the motor housing that are each capable of displacing 119 CFM. Sensors have also been added to the end bells of each motor to monitor internal motor temperature.

3.2. Support Systems

In addition to the base platform, mounting hardware for sensors and electronics must be provided. URSA Minor features an integrated front bumper and sensor deck for mounting and protecting a variety of sensors. The bumper also acts as a handle for lifting and anchors a U-shaped mast for antennae and adjustable camera mounts. A centralized enclosure houses electronics and interfacing hardware, provides an ergonomic support for the vehicle’s computer, and features tactile switches and a system status display.

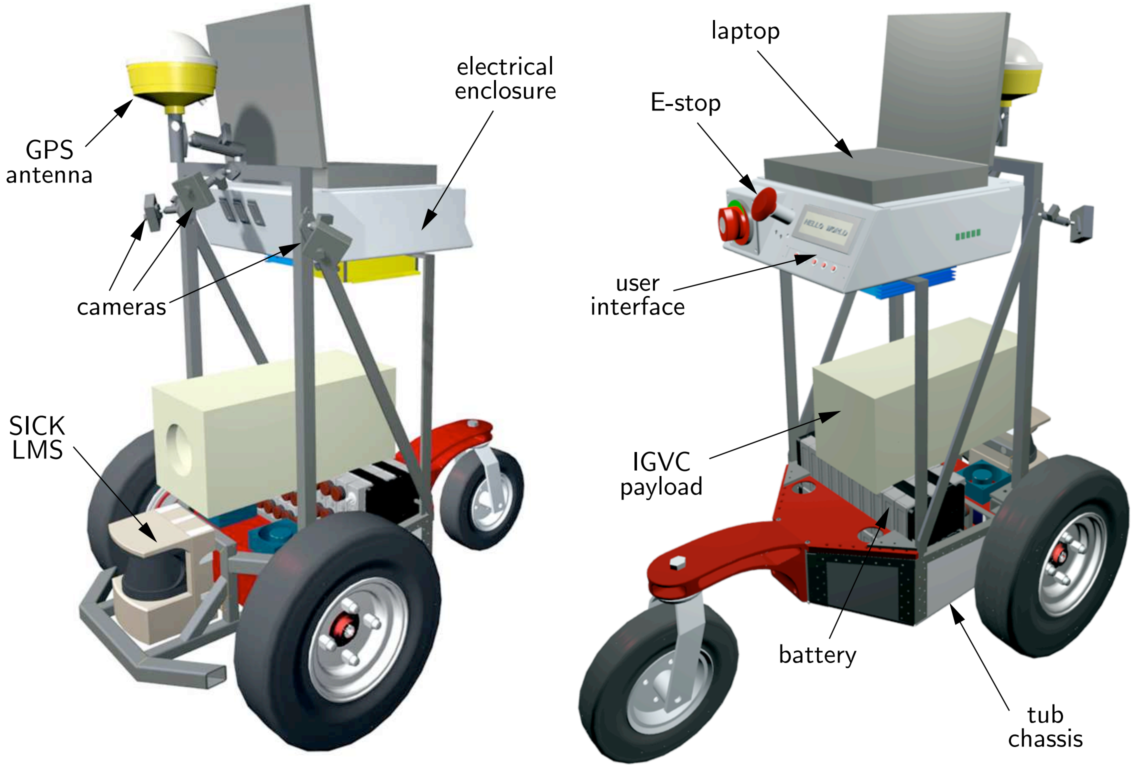


Figure 3: URSA Minor CAD model rendering

4. Electrical System

As shown in Figure 4, the electrical system has two main nodes: the laptop computer and the microcontroller/ daughterboard combination. The laptop hosts the high-level software (discussed in section 5) and interfaces directly with high-level sensors (digital compass, DGPS, and LMS via USB and cameras on the IEEE-1394 bus).

Low-level control, safety, and system status monitoring duties are handled by the microcontroller/daughterboard. This system hierarchy was established to exploit the robustness of the Motorola MC68332 and establish a consistent interface across all ARVP hardware platforms independent of the high-level computer.

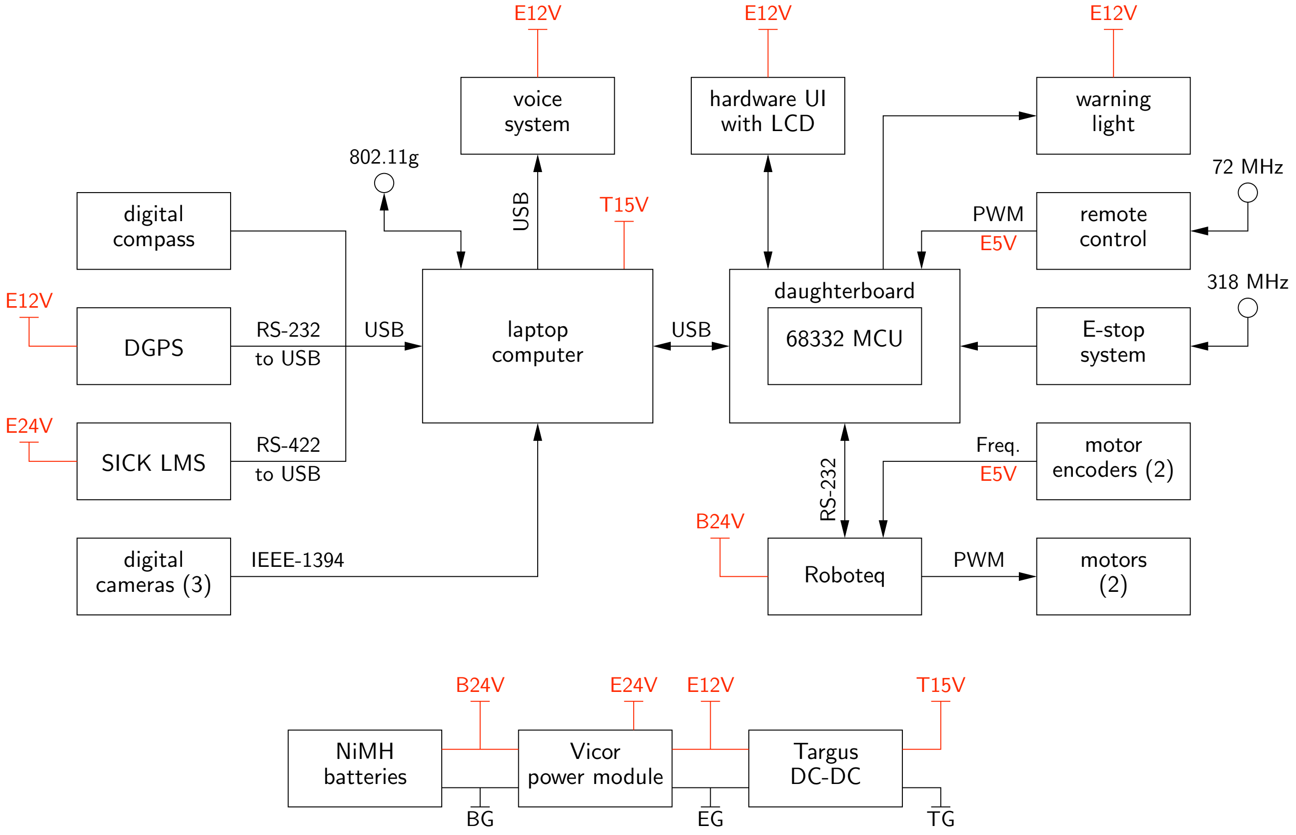


Figure 4: Electrical system interconnection diagram

4.1 Power

Batteries are the only viable power source for indoor operation and are thus used on URSA Minor. Two Panasonic EV-95 12V nickel-metal-hydride (NiMH) packs in series provide 24V motor power and isolated 12V, 15V, and 24V power for onboard electronics via Vicor ComPAC and Targus DC-DC converters. The NiMH packs are selected for their excellent energy density of 70Wh/kg (as compared to 30Wh/kg for sealed lead-acid gel cells used previously) and manageable package size and weight (42 lb each). To improve safety, a polarized quick connect receptacle is fitted to each battery and all battery posts are insulated. Elsewhere in the vehicle, Anderson Power Pole and Power Pak products are used to create unique connector mating patterns to render misconnections impossible.

4.2 Low-Level Control

4.2.1. Motor control

A Roboteq AX2850 dual-channel DC motor controller replaces a set of custom H-bridge-based motor driver boards used on URSA. The Roboteq product was chosen for its adequate current output (60 A continuous) and numerous desirable features. The unit reads motor shaft encoder output directly and offers a tunable closed-loop

PID wheel speed control. It also features current limiting, 2-channel 8-bit analog-to-digital (A/D) conversion for reading motor temperature sensors, and an RS-232 interface.

4.2.2. Microcontroller and Daughterboard

Motor commands are issued to the Roboteq by the MC68332 microcontroller. A custom daughterboard links the MC68332 with both the Roboteq controller and the laptop computer through RS-232 and USB interfaces respectively.

The microntroller/daughterboard combo also drives an external LCD and LED warning light and provides the necessary circuitry to interpret emergency stop and remote control signals. As a result, the vehicle can be driven manually without an onboard laptop. Remote control is accomplished with a COTS FM transmitter-receiver pair and has been shown to function at up to a range of 60-90' (about 20-30 m).

4.3 System Status Monitoring

The Roboteq controller can return system status information including battery voltage, FET amplifier temperature, and temperature, current draw, and shaft speed for each motor. This information is displayed on a 240x64 pixel LCD on the rear of the vehicle and is also sent to the laptop. In turn, status from high-level software such as GPS location can also be reported by the LCD. A series of push buttons below the LCD select a data display mode.

Voice feedback is provided by an RC Systems V-Stamp text-to-speech board, an audio amplifier, and a speaker. While audible feedback can readily be tailored for nearly any message, the system is currently used for high operating temperature and low battery voltage alerts. This sort of feedback is beneficial on a vehicle such as URSA Minor where an operator is not always immediately present.

An LED array mounted on the robot is activated when the vehicle is in autonomous mode. The state of the indicator is determined by the microcontroller so it can be activated during the JAUS competition at the IGVC.

4.4 Safety Systems

4.4.1. Emergency stop

URSA Minor features two levels of emergency stop (E-stop). A soft E-stop can be triggered in software to quickly ramp down the vehicle speed to produce a smooth stop. A hard E-stop uses relays to physically short the leads on each motor thus using the inductive properties of the motors to prevent rolling.

Each of the E-stops can be triggered by switches on the vehicle. A large push-button at the centre rear of URSA Minor invokes the hard E-stop and ensures that the vehicle stops within 6 feet on a 15% grade. Each of the stops can also be triggered using a COTS UHF remote that functions at up to 131' (40 m) from the vehicle.

4.4.2. High temperature shutdown

When the onboard MCU detects a motor temperature greater than 120ºC, it automatically shuts the robot down to prevent damage. This feature can potentially be expanded to include a current-stepping approach to maintain motor temperature below the threshold at the cost of overall performance.

4.5 Sensors

Other than the addition of temperature monitoring and a digital compass upgrade, the ARVP's reliable sensor pack remains unchanged. In summary:

- **cameras**: three Videre Design DCAM digital video cameras are used to identify colored regions in a 180º field of view forward of the vehicle.
- **laser scanner**: Sick LMS-291 laser range scanner detects physical obstacles with a 180º field of view forward of the vehicle.
- **differential GPS (DGPS)**: Trimble AgGPS 132 receives position and heading information corrected by Omnistar differential data.
- **digital compass**: Honeywell HMR3300 digital compass provides 3-D heading, pitch, and roll information and replaces a 2-D compass found to be unreliable on non-level terrain.
- **motor shaft encoders**: US Digital E3 optical shaft encoders indicate the rotation rate of each wheel.
- **temperature sensors**: National Semiconductor LM45AH analog high-temperature sensors are used to measure internal motor temperature up to 150ºC.

5. Software System

URSA Minor's software systems build on the Hazard-Oriented Obstacle Detector (HOOD) created from scratch in 2004 for the Kodiak platform. The HOOD is a completely modular and flexible system for intelligent robot navigation that allows for rapid design and integration of new components. A Toshiba Satellite M40 laptop with a Pentium-M 1.6GHz processor runs the main HOOD software.

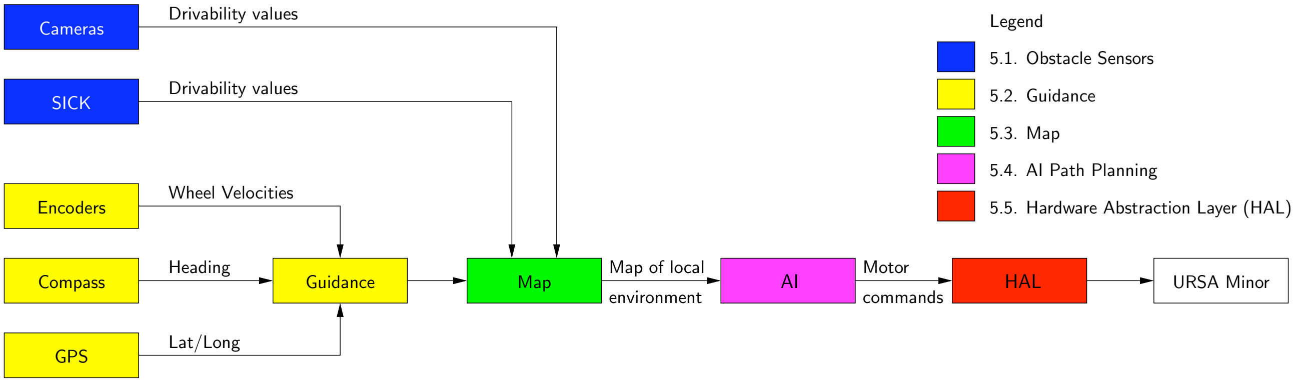


Figure 5: Schematic view of software system

The software system is depicted schematically in Figure 5. Camera images and SICK laser range finder data are processed to find obstacles in the robot's environment. A map containing these obstacles is created and the robot is localized therein by fusing wheel encoder, digital compass, and differential GPS (in the Navigation Challenge) data to estimate position and orientation. The map is used to find an acceptable path to meet the robot's goals and corresponding commands are issued to the drive system. As time progresses and the robot moves, this process repeats and the vehicle is guided around the course. Each aspect of this process is visualized by a flexible graphical user interface (GUI). The GUI can also be used to modify settings that control the operation of HOOD algorithms in realtime.

One important feature that was added in 2006 to meet the design goal of testing facilitation is the HOOD Log of Unified Messages (HOODLUM). HOODLUM is a data logging system that records relevant robot status information during vehicle operation. Post-processing of this data provides debugging and performance evaluation information.

5.1. Obstacle Sensors

Lines, potholes, and physical objects are all considered to be "obstacles". Obstacles detected by the vision system and laser scanner are assigned a negative drivability value. The complement to this scheme is that drivable parts of the environment such as clear lanes are also identified and assigned a positive drivability value. The configuration and range of the cameras and SICK LMS are shown in Figure 6. Sample sensor data is shown in Figure 8.

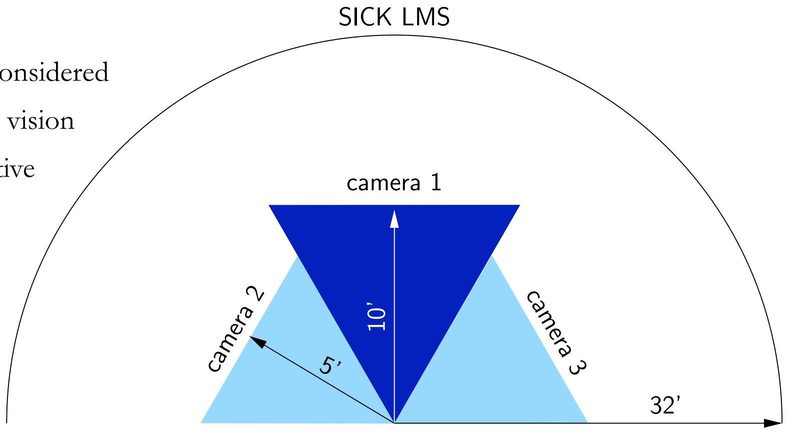


Figure 6: Overhead view of camera and LMS configuration, field of view, and range (up is looking ahead of the vehicle)

5.1.1. Vision

Three digital video cameras provide images with a 180º field of view in front of the robot at a rate of 7.5 Hz. The multi-camera system provides redundancy since the robot can still function with fewer than three cameras. Each camera is calibrated to find a transformation between any point in an image and the corresponding point on the ground in the robot-centric reference frame.

Pixels of interest (white and yellow for the IGVC) are identified using hue-saturation-luminance (HSL) thresholding to find obstacle course lines and potholes. HSL is chosen over other schemes such as red-green-blue (RGB) for its more intuitive color description. Pixels are assigned a value based on how closely they match the colors of interest and the image is divided into rectangles. The average of the values in a rectangle determines its drivability and effectively smoothes image noise. Each rectangle is then transformed into the robot-centric frame and added to the map. In this way, the vision system provides positive and negative feedback to the map by defining areas that are safe and unsafe to drive, respectively.

5.1.2. SICK

The SICK LMS scans a 180º field of view in front of the robot in 0.5º increments at a rate of 37.5 Hz. While the LMS is still accurate to better than 0.4" at a range of 100', data beyond 32' is ignored since it is not relevant to local navigation. Large discontinuities in the range data identify obstacles and are assigned a negative drivability value. Positive drivability values are created in the space between the robot and the obstacle to indicate that the area is traversable. Since ramps at the IGVC are seen by the laser but are traversable, they are specially detected as straight lines of a particular width and are ignored. This ramp detection is optional and is disabled to allow the robot to avoid walls during indoor navigation.

5.2. Guidance System

The vehicle requires an accurate estimate of its pose (position and orientation) in order to build a map of the local environment. The HOOD Guidance system fuses data from multiple sources to provide a pose estimate relative to either an initial position or an absolute reference.

A technique known as odometry is used in relative mode. Wheel velocity from wheel encoders is fed into a simple robot kinematic model at 20 Hz to provide a pose estimate. Odometry is accurate over many meters so a local map around the robot can be maintained. All odometry systems are subject to drift and errors caused by factors such as wheel slippage and varying terrain so an accurate map of the entire course cannot be built.

In absolute mode, a differential GPS (DGPS) signal is used to update the robot's position at 10 Hz. This position is absolute in the sense that it has a one-to-one mapping with a standard earth-fixed latitude/longitude coordinate system. Since GPS cannot provide a reliable heading when stationary, a digital compass is used for heading estimation. Magnetic north to true north corrections are automatically done using magnetic declination information provided by the GPS unit.

The HOOD Guidance system automatically chooses between absolute and relative modes depending on the permission to use and the availability of the DGPS signal. In absolute mode, odometry provides position estimates between GPS updates and acts as a backup when no DGPS signal is available. This backup makes the Guidance system tolerant to GPS outages or sensor malfunction.

5.3. Map

With the robot pose estimated and the position of environment obstacles measured, a map can be generated around the robot. This map provides a top-down view of the robot and its surroundings to the path-planning artificial intelligence (AI) module (see Figure 8). Since only the local environment is of concern, the map moves with the vehicle and maintains a constant size. It was found that a square map extending about 33' (10 m) from each side of the vehicle made up of small square tiles roughly 4" (10 cm) to a side produced desirable results.

Each obstacle sensor translates its obstacle information into shapes with drivability values. The map takes these shapes and generates a corresponding set of map tiles. After the next sensor update, the map takes the most pessimistic drivability value for each tile and combines it in a weighted average with the current map. This scheme ensures that the obstacles are always included even if sensors provide conflicting information.

The map system is generic in the sense that obstacles are not classified as lines, potholes, or barrels, but rather simply as areas of negative drivability. It also presents a unified view to path planning modules that is independent of the sensors used to generate the map. From a software architecture viewpoint, this is advantageous since it separates sensors from path planning so adding new obstacle or pose sensors does not require changes to path planning modules.

5.4. Artificial Intelligence (AI)

All artificial intelligence components that drive URSA Minor are based on a common obstacle avoidance algorithm. Although the Autonomous and Navigation Challenges have different goals, the modular nature of the HOOD allows the same basic principle to be used for both.

5.4.1 Obstacle avoidance algorithm

The goal of the obstacle avoidance algorithm is to choose a path for the robot in the direction that deviates the least from the destination direction.

Obstacle avoidance actions can be divided into four distinct cases illustrated in Figure 7. Each case depends on the location of obstacles relative to a forward threshold (4 m or about 12' ahead of the robot) and a trap threshold. These thresholds are determined experimentally to achieve smooth performance at IGVC speeds and are GUI-adjustable.

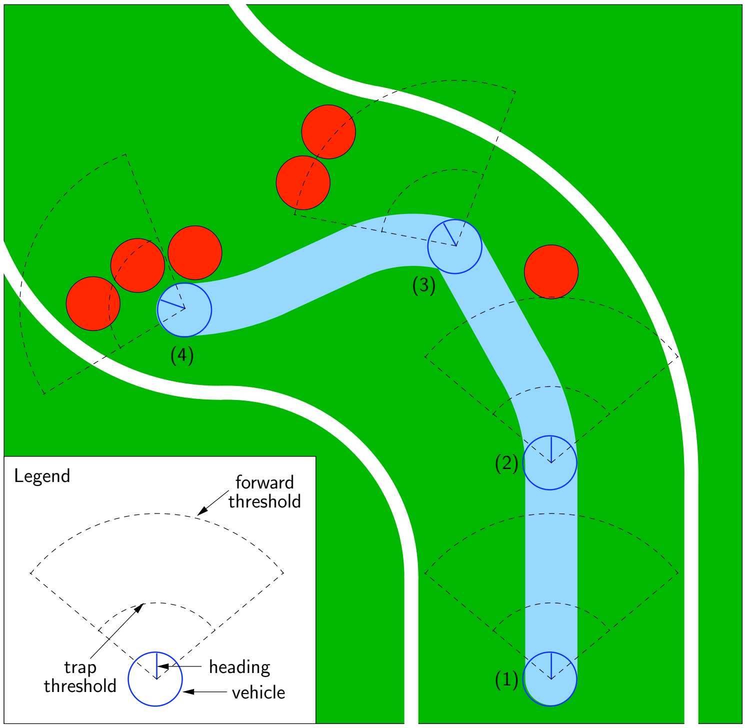


Figure 7: Overhead view of obstacle course illustrating four cases considered by path planning AI. Refer to text for explanation.

Case (1): No obstacles on the current heading within the forward threshold. The vehicle is commanded to continue on the same path.

Case (2): An obstacle is found on the current heading at the forward threshold. The distance the robot can travel along an arc ±5° from the current heading is calculated. The magnitude of this arc angle is incremented by 2.5° until a path is found along which the robot can travel at least as far as the forward threshold.

Case (3): If the arc angle is incremented to ±50º without finding a path that satisfies the forward threshold, the arc resulting in the greatest forward progress is taken so long as this distance is greater than the defined trap threshold.
Case (4): If all possible paths within the ±50º arcs do not produce forward progress beyond the trap threshold, a trap or dead end is assumed. The forward region is labeled undrivable and the robot backs up until a clear path is found. Reversing without rear-facing sensors is possible due to the existence of the environment map.

Once a path is chosen, the speed of the vehicle is based on its proximity to obstacles. The robot travels more slowly near obstacles as to minimize any possible impact in case of system failure. Analogously, the robot moves more quickly in open areas. A real-world example of the path planning AI is shown in Figure 8.

5.4.2. Autonomous Challenge

During the Autonomous Challenge, there is no prior information available about the course so no optimal destination direction can be selected. As a result, the obstacle avoidance algorithm in described in section 5.4.1. is commanded to drive the robot straight forward. Lines, potholes, and other obstacles cause the robot to drive in smooth arcs around the course.

5.4.3. GPS Navigation Challenge

The obstacle avoidance algorithm in section 5.4.1. can also be implemented in the GPS Navigation Challenge with an overall goal change. Where the Autonomous Challenge required continued forward progress, this event has a set of well-defined position goals. After an offline calculation of the shortest path between a set of given waypoints, the GPS AI chooses a destination direction for the obstacle avoidance algorithm that drives the robot to the first waypoint. Once the robot has reached within a certain threshold of the

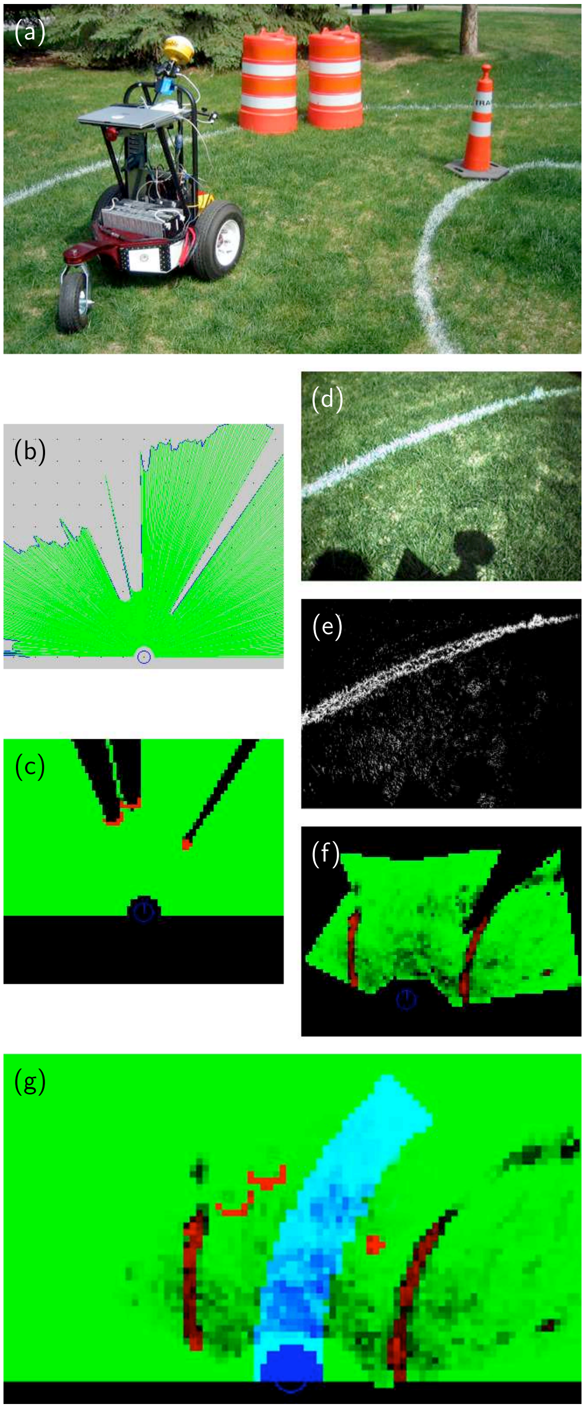


Figure 8: Stages of data processing in the HOOD from GUI screenshots. (a) external view of scene; (b) SICK laser range data; (c) SICK data converted to obstacles and assigned drivability values in map; (d) raw captured image from left camera; (e) image thresholded for white colored pixels, showing detected line and noise; (f) drivability values from cameras after transforming to map. Red squares represent undrivable areas, green squares are drivable, and black is unknown. (g) Path chosen by the AI shown in blue.

waypoint, the AI switches its destination to the next waypoint. After traveling to all the assigned waypoints, the GPS AI navigates the robot to the event's starting point and uses the Guidance system to orient the robot to true north.

5.5. Hardware Abstraction Layer (HAL)

A Hardware Abstraction Layer (HAL) gives the HOOD a generic interface to the robot's hardware. General commands such as "move forward at 1.0 m/s" or "turn at 0.5 rad/s" produced by the AI module are translated by the HAL into hardware-specific commands and sent to the robot's hardware. HAL interfaces exist for URSA Minor as well as previous ARVP platforms such as URSA, Kodiak, and Bearcub. There is also a HAL interface to the Gazebo simulator that allows algorithms to be developed and tested in a virtual environment and then moved to physical platforms without additional programming.

5.6. Joint Architecture for Unmanned Systems (JAUS)

The Joint Architecture for Unmanned Systems (JAUS) is designed to standardize communications with and between unmanned vehicle systems. To learn bout JAUS, ARVP members investigated the JAUS Working Group website and studied the protocol's specifications in the Reference Architecture documentation available online.

Due to the flexibility and modularity of the HOOD, the integration of JAUS message handling was trivial. A JAUS module was created that receives messages via the UDP protocol on the appropriate port (3794). This implementation can be used with any network interface that supports UDP/IP and is currently configured to use the 802.11g wireless specification as per the IGVC rules. As required, the laptop running the HOOD software has a switch to power off the 802.11g radio when the system is not performing in the JAUS competition.

Valid JAUS messages are received by the HOOD's JAUS message handling module. The system can currently interpret the JAUS Resume, Standby, and Set Discrete Devices messages. Resume and Standby instruct the HAL system to begin and cease sending commands to the physical robot hardware, respectively. The Set Discrete Devices message uses the Horn On/Off bit to instruct the HAL to activate URSA Minor's warning light. A JAUS Operator Control Unit (OCU) has also been created as a software program that can run on any computer with a network interface. The OCU can communicate with the the robot's onboard laptop to issue JAUS commands.

No significant challenges were encountered when implementing basic JAUS capabilities. The modularity of the HOOD system made JAUS easy to integrate and further functionality can be added by translating other JAUS messages into appropriate commands to various HOOD subsystems. Expansion is planned to support more commands in the Core, Platform, and Environment Sensor JAUS subgroups.

6. Performance Summary

6.1. Speed

URSA Minor is regulated to never exceed a speed of 4.92 mph (2.2 m/s). Speed is governed by the Roboteq motor controller hardware based on feedback from the motor shaft encoders. PID control ensures that wheel speeds never exceed this maximum speed threshold to ensure compliance with the 5 mph IGVC speed limit.

6.2. Ramp climbing ability

URSA Minor’s motors are able to supply sufficient torque to climb a 23° slope (42% grade) . This specification exceeds the requirements of the IGVC where natural and artificial inclines with gradients do not exceed 15%.

6.3. Reaction times

The breakdown of HOOD system tasks is shown in the table at right with a corresponding average execution time. Image processing and generating obstacles in the map are the most time consuming. The time to complete an iteration of these software tasks is ~162 ms so the the robot has an equivalent reaction time.

Task	Average time required
Vision capture (3 cameras)	1 ms
Vision process (3 cameras)	93 ms
SICK capture	0.002 ms
SICK process	1 ms
Guidance update	0.002 ms
Map position	0.004 ms
Map obstacles	65 ms
AI path plan	2 ms
HAL commands	0.035 ms
Total	**~162 ms**

6.4. Battery life

With all systems running at peak values, URSA Minor has a theoretical endurance of 180 minutes with a set of charged batteries. In practice, such continuous use is rare so battery life is typically observed to be many times this number.

6.5. Distance at which obstacles are detected

Cameras detect obstacles based on color at a distance of up to 10’ ahead of the vehicle and 7’ on the sides as shown in Figure 6. Physical obstacle data within 32’ of the front of the robot is captured from the SICK laser range scanner.

6.6. How the vehicle deals with dead ends, traps, and potholes

Dead ends and traps are considered in Case (4) of the obstacle avoidance algorithm in section 5.4.1. If the robot finds itself unable to progress forward in such a situation, it backs up until a clear path is found. Potholes are identified by the cameras and assigned a negative drivability value in the map.

6.7. Accuracy of arrival at navigation waypoints

While the vehicle must arrive within 2 meters of the Navigation Challenge waypoints, the onboard DGPS system with Omnistar correction data has up to a meter of position error. Therefore, the GPS AI drives the robot to within 1 meter of the course waypoints.

7. Conclusion

URSA Minor represents a step forward for the ARVP in terms of safety, reliability, and durability. Proven chassis components are leveraged by COTS hardware, custom electronics, and intuitive hardware and software interfaces to create a capable vehicle for autonomous applications. A highly modular software system promotes rapid innovation including the recent implementation of portions of the JAUS protocol. These factors combined with a talented team ensure that URSA Minor will be a strong contender at the 2006 IGVC.

8. Team Members

The total estimated person hours committed by ARVP members to the development of URSA Minor's mechanical, electrical, and software systems is **1750 hours**. This number is in addition to the estimated 3750 hours spent developing URSA during 2004-2005. The team member breakdown is shown below.

Name	Team	Program	Year
Barkwell, William	Mechanical	B.Sc. Mechanical Engineering Co-op	3
Blinzer, Michael	Mechanical	B.Sc. Mechanical Engineering Co-op	4
Bothe, Juval	Electrical	B.Sc. Electrical Engineering	3
Brook, Eli	Mechanical	B.Sc. Engineering	1
Bulley, Eric	Electrical	B.Sc. Engineering	1
Kastelan, David	Team Leader	M.Sc. Electrical Engineering	2
Kilgour, Marion	Mechanical	B.Sc. Mechanical Engineering Co-op	3
Kalogirou, Nicholas	Electrical	B.Sc. Electrical Engineering Co-op	2
Klaus, Jason	Software	M.Sc. Electrical Engineering	2
Klippenstein, Jon	Software	M.Sc. Computer Science	2
Kluthe, Nancy	Outreach	B.A. Industrial Design	4
Knowles, Robert	Mechanical	B.Sc. Computer Engineering	5
LaFleche, Mathieu	Electrical	B.Sc. Engineering Physics	2
McIvor, Jake	Mechanical	B.Sc. Mechanical Engineering Co-op	4
Melenchuk, Steven	Software	B.Sc. Engineering	1
Murugan, Sada	Electrical	B.Sc. Electrical Engineering	2
Ng, Jason	Electrical	M.Sc. Electrical Engineering	2
Parseyan, Hassan	Electrical	B.Sc. Electrical Engineering	4
Simpson, Doug	Mechanical	B.Sc. Mechanical Engineering	2
Toogood, Roger	Faculty Advisor	Mechanical Engineering	
Wirsz, Ryan	Mechanical	B.Sc. Mechanical Engineering	3

9. Component Cost Summary

Component	Model	Quantity	Unit Price (USD)
Mechanical Components			
Materials	6061 T6 Aluminum, tubing, sheet	1	$1,750
Motors	Magmotor S-28-100	2	$235
Gearboxes	Apex Dynamics AB090-025-S2-P2	2	$650
Tires	Kenda AG Tire w/rim	2	$65
Electrical/Computer Components			
Laser Range Scanner	SICK LMS-291	1	$3,600
GPS	Trimble AgGPS 132	1	$3,700
Video Cameras	Videre Design DCAM	3	$210
Digital Compass	Honeywell HMR3300	1	$250
Motor Shaft Encoders	US Digital E3	2	$55
Motor Controller	Roboteq AX2850	1	$620
Daughterboard	Custom	1	$85
LED lights	Custom	2	$45
Wiring/Connectors	Anderson Powerpole, miscellaneous	1	$300
Laptop Computer	Toshiba Satellite M40	1	$1,400
Batteries	Panasonic EV-95	2	$250
Power Module	Vicor Custom	1	$450
Remote Control	72 MHz Analog FM	1	$140
		TOTAL $15,525	