The DARPA Grand Challenge – Development of an Autonomous Vehicle

Reinhold Behringer, Sundar Sundareswaran,
Brian Gregory, Richard Elsley, Bob Addison,
Wayne Guthmiller

Rockwell Scientific (RSC)
1049 Camino Dos Rios
Thousand Oaks, CA 91360
{rbehringer, vsundar, bgregory, relsley, raddison,
wguthmiller}@rwsc.com

Robert Daily, David Bevly

Department of Mechanical Engineering
Auburn University, AL 36849-5341
{dailyrl, dmbevly}@eng.auburn.edu

Abstract

The DARPA Grand Challenge (DGC) was an opportunity to test autonomous vehicles in a competitive situation. In addition to intelligent behaviour, the participating vehicles must also exhibit ruggedness and endurance in order to survive the fast ride over rough terrain ("win with the software – lose with the hardware"). The SciAutonics teams decided to use compact and agile vehicles that employ proven mechanical designs very suitable for the desert environment. 4-wheel drive ensures robust controllability even in slippery ground, and a roll cage protects the vehicle components from damage in case of a collision. The control system relies primarily on a differential GPS (Starfire) and a set of inertial sensors for navigating between the given set of waypoints. A sensor suite using infrared laser (LIDAR) and ultrasound sensing provides the capability of obstacle avoidance and path following. This paper shows the components of the vehicle and results from driving at the DGC.

1. Introduction

When the DARPA Grand Challenge (DGC) was officially announced by DARPA in summer 2002, it was clear that such an event was exceptionally ambitious: a competitive comparison of autonomous road vehicles, through challenging terrain with a tight time limitation. The route was said to lead from Los Angeles to Las Vegas, and the vehicles were required to drive autonomously without any user control or intervention. The route was supposed to be 210 miles (380 km) and was required to be completed in 10 hours. In February 2003 more details were revealed regarding organization and goals. This event initiated the forming of participating teams. Employees of Rockwell Scientific Company (RSC), a privately owned research facility in Thousand Oaks (in Southern California), decided to participate in the DGC and form a team. Previous work of these employees had involved research and development in the domains of path planning, decision-making, intelligent autonomous behaviour, and autonomous road vehicle development [1]. Further, our team had expertise in software architecture, computer vision, sensor physics, and material science. Since several employees had significant expertise in active participation in (manually driven) desert / off-road races, we were quite confident and decided to go ahead and form a team for participation in the DGC. For specific expertise that was complementary to the expertise of the RSC employees on our team, we partnered with two universities: Auburn University provided valuable expertise in designing the vehicle control system, and California Lutheran University (in Thousand Oaks) provided image acquisition technology. In addition, specific vehicle expertise was provided by external off-road race specialists and the vehicle manufacturer ATV Corporation. DARPA rules explicitly forbid charging labour and costs to US government contracts. In addition, RSC did not choose to provide coverage of the direct labour costs. Therefore, in order to have a legal entity that would allow us to preserve the technology and intellectual property that we would develop during the work on this DGC and to facilitate volunteer participation from the broader community, we formed a Limited Liability Company: SciAutonics, LLC, registered in California. RSC provided SciAutonics direct financial support and use of its facilities during non-business hours with the stipulation that no time could be spent on the DGC during regular working hours.

2. System Architecture

One of the key issues in participating in the DGC is the robustness against failure of individual components. Therefore, our design and architecture placed significant focus on a highly modular approach, in which individual components could fail and the overall system performance would slowly degrade instead of abruptly terminate. This required that the modules were highly independent and could run asynchronously. The overall architecture that resulted from these considerations is shown in Figure 1.

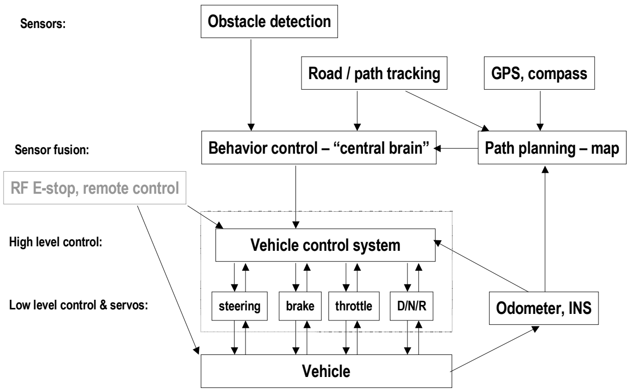


Figure 1. Vehicle system architecture.

2.1 Vehicle Control

At the core of the architecture is the vehicle control. This module is the one that "is not allowed to fail" since it serves as the fall-back mode for the autonomous driving capability. It uses differential GPS and a set of inertial sensors. The vehicle control module consists of three parts, the navigation estimator, throttle control, and steering control. Each part operated at a 50 Hz update rate.

2.1.1 Navigation Estimator

An Extended Kalman Filter (EKF) was designed to fuse the inertial measurement unit (IMU) output with the GPS measurements to provide high update rate measurements to the vehicle controller. The EKF provided a 50 Hz update rate of position, velocity, yaw, roll, and pitch as well as estimates of the IMU sensor biases. The EKF was based on the algorithms developed in [3] and [4]. The EKF integrated the IMU to provide dead-reckoning navigation between the 5 Hz GPS update rate measurements and during periods when no GPS measurements were available. Figure 2 is a plot from an experimental run on a practice field. There was a tree located at the North-East corner of the field which created a GPS outage during testing. After a GPS outage the Starfire measurements downgrade to "Mode 6" which has an accuracy of 1-3 meters. Once the Starfire GPS receiver reacquires the differential correction signal, the receiver returns to "Mode 11" which has an accuracy of approximately 10 cm. The ability of the EKF to provide accurate position estimation after a GPS outage can be seen in Figure 2.

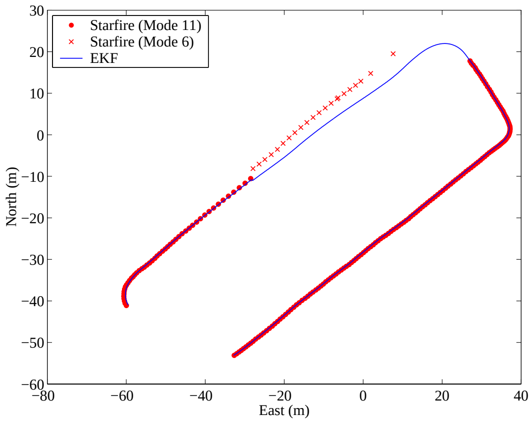


Figure 2. Experimental test of GPS/INS integration, including a GPS outage.

2.1.2 Throttle Control

The job of the throttle control is to cause the vehicle to drive at a certain speed. To develop this controller the speed dynamics of the vehicle were modelled as a first order lag. The control was accomplished using proportional and integral terms. Figure 3 shows the response of vehicle during the actual race. One drawback of this control method we noticed was a tendency to overshoot the desired speed when the vehicle was initially started. The overshoot was due to the large error between the desired and actual speed when the vehicle was not moving. The large initial jump in speed caused undesirable behaviour in both the steering control and the obstacle detection software. To minimize the effects of this overshoot the desired speed was set to 1 m/s for the first 100 waypoints. After that the desired speed jumped to 3 m/s. This is the step shown in the figure. Another unexpected aspect was the oscillation in the speed once the system reached steady state. This oscillation implies un-modelled dynamics somewhere in the system such as the continuously variable transmission.

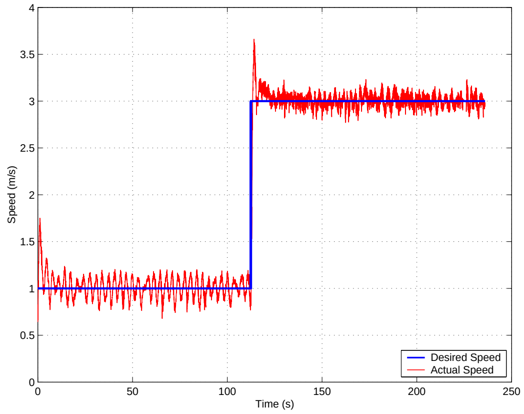


Figure 3. Speed response of the vehicle during the race.

2.1.3 Steering Control

The steering control drives the vehicle to the different waypoints. Through testing, the vehicle's lateral dynamics (steer angle to yaw rate) were determined to be second order with a natural frequency of approximately 2 Hz and change with the speed of the vehicle. However, due to a lack of time and inability to measure all of the vehicle states (such as sideslip) the lateral control was developed around a first order system and the controlled natural frequency was kept low enough not to excite the higher order modes of the system.

$$\frac{r}{\delta}=\frac{V_x/L}{\tau s+1}$$

This equation contains the following terms: yaw rate r, steer angle δ, speed V_x, wheel base L, and yaw rate time constant τ. The vehicle's heading, ψ, adds an integrator to the total system leading to the following state space representation

$$\begin{bmatrix}\dot{\psi}\\ \dot{r}\end{bmatrix}=\begin{bmatrix}0 & 1\\ 0 & -\frac{1}{\tau}\end{bmatrix}\begin{bmatrix}\psi\\ r\end{bmatrix}+\begin{bmatrix}0\\ \frac{V_x}{L\tau}\end{bmatrix}\delta$$

The steering control was implemented using state feedback methods.

$$\delta=-k_1\left(\psi-\psi_{des}\right)-k_2 r$$

Gains for the yaw rate and heading error were calculated to place the closed-loop poles at a desired natural frequency and damping (ω_{ndes}, ς_{des}). These gains were modified to keep the closed-loop dynamics constant regardless of the vehicle speed. If the yaw rate gain (k_2) became negative it was set to zero.

$$k_1=\frac{L\tau\omega_{ndes}^2}{V_x}$$

$$k_2=\frac{\frac{2\zeta_{des}\omega_{ndes}}{L\tau}-L}{V_x}$$

The next step of the waypoint control was to determine which waypoint the vehicle should drive towards. To smooth out the vehicle's response and reduce the overshoot when finding the path of waypoints, a radius, R, was drawn around the vehicle. This radius was based on the vehicle speed; as the vehicle moved faster the radius increased. The vehicle drove to the first waypoint ahead of it and outside the radius as shown in Figure 4.

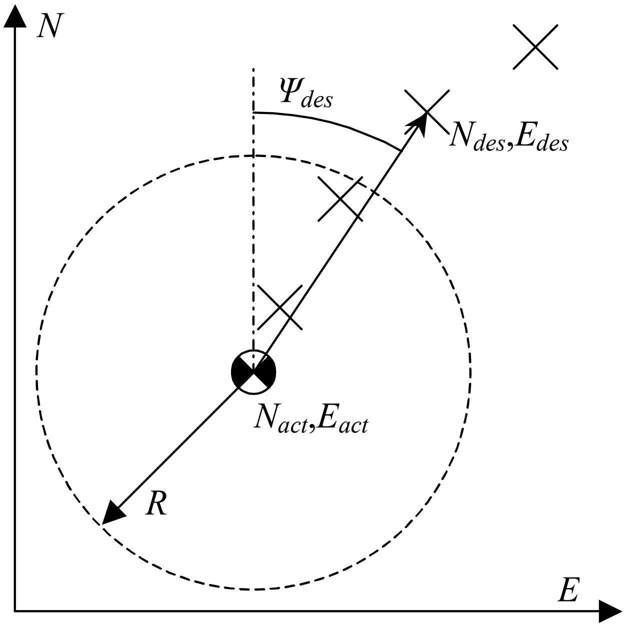


Figure 4. Determining the waypoint to drive towards.

The desired heading to the next waypoint (ψ_{des}) could then be computed.

$$\psi_{des}=\tan^{-1}\left(\frac{E_{des}-E_{act}}{N_{des}-N_{act}}\right)$$

Figure 5 shows the waypoint tracking during a portion of the race. The jaggedness of the desired path is due to the precision of the data file and is not present in the actual control computations. The times where the desired path goes to an earlier waypoint is due to the waypoint search algorithm finding a slightly different point. The closeness of the varying points and the time the vehicle drives to a point before seeing a different point combine to make this waypoint hunting a very minor issue that caused no problems to the vehicle control.

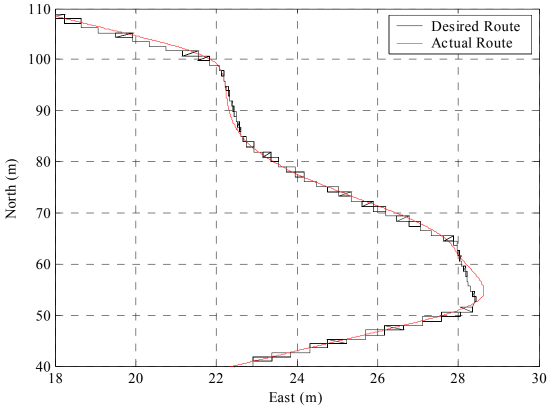


Figure 5. Waypoint tracking during the race.

2.2 Path Planning

Detected obstacles and road boundaries are placed into a 2D birds-view map. It is now up to the path planning module to follow the given waypoint course in order to avoid collision with obstacles, stay within the path, and stay within the DARPA-issued corridor boundaries. For this we developed a simple approach than implements a shifting of the waypoint course to the side around obstacles. In a given constant spatial interval, waypoints are sent to the vehicle control (steering control), making the vehicle follow this path of waypoints.

3. Hardware

3.1 The Vehicle(s)

Following the paradigm “winning with the software, loosing with the hardware” we placed a significant emphasis on finding a vehicle that would be able to withstand errors or mistakes induced by the autonomous navigation software. In our search for the ideal vehicle that would master the challenges of desert off-road and dirt road driving, we concluded that a typical desert race vehicle would be the optimal solution. We teamed with desert race driver Paul Gunthner who offered to use his race vehicle, which provided a very rugged structure and high maximum speed. It has a 500 hp Chevrolet 350 cci V8 engine that can bring the vehicle to a speed of 128 mph (200 km/h) on typical off-road ground.

For development and testing, however, the vehicle seemed too powerful and too large. Therefore, when SciAutonics got the offer from ATV Corporation to use and to modify one of their Prowler vehicles for the DGC, we decided to begin implementing the system for autonomous driving on this vehicle (Figure 6). This vehicle is basically a modified Yamaha All Terrain Vehicle (ATV). The engine is a one-stroke 660 ccm engine, driving a 4-wheel-drive train. It can accelerate the vehicle to a speed around 110 km/h. In manual driving on rough desert surfaces, we reached a speed of 90 km/h. The original Yamaha design was modified by ATV Corporation to encapsulate the vehicle with a solid roll cage. We named it RASCAL – Robust Autonomous Sensor-Controlled All-terrain Land-vehicle.

Figure 6. The RASCAL vehicle of team SciAutonics-I, in the Grand Challenge race shortly after the start.

The fact that SciAutonics had two suitable vehicles, led to the consequence that we submitted two participant applications to DARPA. Both got accepted, and SciAutonics ended up to be the only sponsor among the 25 contestants to have two entries in the DGC. The RASCAL vehicle was fielded by the team SciAutonics-I. As the race came closer, we realized that funding for SciAutonics-II was too low for equipping the vehicle with an autonomous control system. We accepted the offer by Elbit, a company that already had a vehicle equipped for autonomous waypoint following. This vehicle – AVIDOR-2004 (see Figure 7) – was then the entry for the team SciAutonics-II, or “Elbit-SciAutonics”. The manufacturer of this vehicle is Tomcar. It is a 2-wheel drive off-road vehicle and in some ways similar to RASCAL. After we received the RASCAL vehicle in August 2003, we began implementing actuators for steering, brakes, throttle, and gear shifting. Although the vehicle transmission is an automatic system, we need the external control to shift into Neutral, Drive, Reverse, or engage 4-wheel drive or the differential block. A DARPA provided electronic stop device was installed which would allow to execute a remote control shut-down of the vehicle in case of an emergency (impending collision or other danger). During the challenge drive, only DARPA was allowed to execute that E-stop.

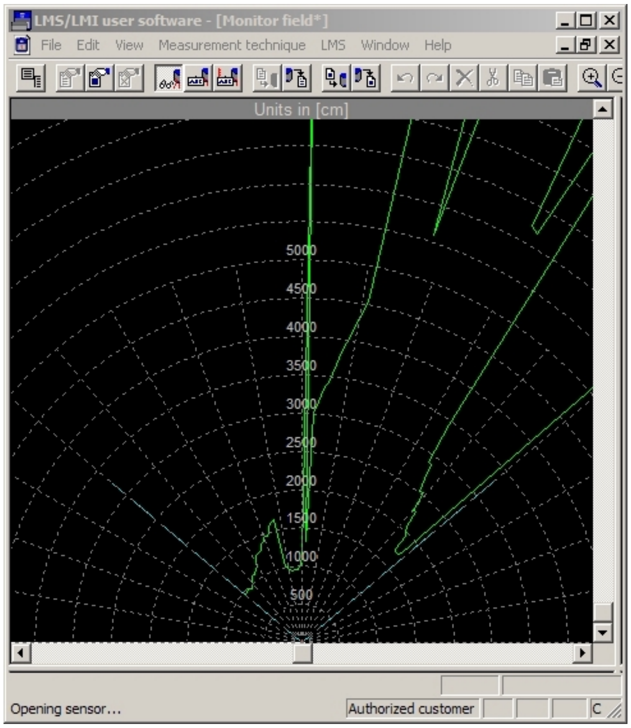
Figure 7. AVIDOR-2004 - the actual entry of the team "Elbit-SciAutonics" (SciAutonics-II), driving on the California Speedway QID track in Fontana .

3.2 Sensing Systems

3.2.1 Obstacle Detection

Sensors for obstacle detection are essential for an autonomous vehicle. In our design we choose a combination of several long-range and short-range sensors for obtaining a coverage of the areas that are within our path. The farthest look-ahead distance that this suite can cover is 80 m. This is the reach of the infrared LADAR scanning unit manufactured by SICK. This scanner provides a line scan result every 13 ms and returns an array of distance points along a 180 degree scan. A situation as shown in Figure 8 results in a scan plot shown in Figure 9. The vehicle that is within the field of view, is clearly shown by the LIDAR scan. We set up a configuration of several Lidar units, scanning both horizontal and vertically. A vertical scan provides information about the road surface and about cliffs. In order to have redundant obstacle detection for short range situations, such as crossing under an overpass, we implemented an array of ultrasound measurement units. These devices provide additional "beams" with distance measurements within 4 m for slow navigation.

Figure 8. A van serves as an "obstacle" during testing on the RSC parking lot.



Opening sensor...

Figure 9. Screenshot of LIDAR scan from the situation shown in Figure 8. The green plot indicates the detected obstacle environment.

In addition, optical sensors provide further short-range object detection capability. All these devices are mounted at fixed locations on the vehicle, aiming in fixed directions and monitoring specific areas. Their overlap ensures an increased confidence in the detection of obstacles, provides redundancy, and allows to reduce the possibility of “ghost obstacles” due to sensor failures.

3.2.2 Path Following

According to DARPA, most of the course would go over dirt roads. For detecting the borders of such unpaved paths, we developed software based on computer vision approaches for path detection. Simple, fast, and robust image processing operators detect the edge of the path, even if this “edge” is not as pronounced as that of a paved road (see Figure 10). These edge locations are fed into a spatio-temporal update of vehicle offset and yaw angle, similar to the well-published 4D-approach [2].

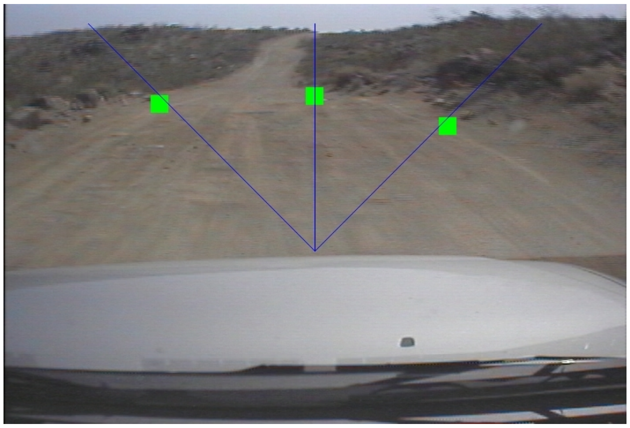


Figure 10. Detection of path boundaries of typical dirt road scene.

The imaging system that we selected is a pinhole camera. This provides low mass and robust mechanics for mounting. The complete image processing is implemented on a ruggedized laptop running Windows 2000, using a DirectShow video capture graph for an analog-to-firewire capture device. It allows rapid image capture at 30 fps and at a size of 320x240 pixels. For reasons of rapid processing, the image analysis only considers grey scale values of the pixels.

4. Performance during the DGC

As the DGC came closer, we realized that not all of our ambitious goals could be achieved in time. Since most of the work on this project was done in spare time (evenings and weekends), we were quite behind our original schedule. That forced us to make certain compromises in the actual implementation of the autonomous vehicle guidance system. The visual road recognition was not activated, and the obstacle detection was activated with using only one (horizontal) scanner in order to avoid potential problems that had not been solved earlier. In the actual event on March 13, RASCAL achieved autonomous driving for 0.75 miles before quitting operation due to a hard drive failure – this made it 7th in the ranking. Data plots from log files of RASCAL will be shown at the Intelligent Vehicle conference. The other vehicle sent by the Elbit-SciAutonics team, AVIDOR-2004, achieved driving for more than 6 miles before getting stuck at the side of the road in a slope. The farthest came the RedTeam vehicle “SandStorm” (sponsored by CMU): 6.7 miles distance.

5. Summary

This event had been highly publicized; therefore, the outcome seemed a little disappointing to participants. The course was not very difficult, with the hardest part was not too far from the starting line (mountainous). There need to be significant technical improvements for the next DGC in 2005, especially on the sensor interpretation side and the mechanical hardening (vibration proofing).

Acknowledgments

We would like to thank Rockwell Scientific Company, Rockwell Collins, Auburn University, and all the individuals that supported the SciAutonics effort in the DGC.

References

[1] R. Behringer and N. Müller, “Autonomous Road Vehicle Guidance from Autobahnen to Narrow Curves,” *IEEE Transactions on Robotics and Automation,* Vol. 14, no. 5, pp. 810-815, 1998.

[2] E.D. Dickmanns, “An Integrated Approach to Feature Based Dynamic Vision,” In: *Int. Conf. On Vision and Pattern Recognition (CVPR)*. Ann Arbor, pp. 820-825, 1988.

[3] Bevly, D. M., “GPS: A Low Cost Velocity Sensor for Correcting Inertial Sensor Errors on Ground Vehicles,” *Journal of Dynamic Systems, Measurement, and Control*, June 2004.

[4] Bevly, D. M., Rekow, A., Parkinson, B., “Evaluation of a Blended Dead-Reckoning and Carrier Phase Differential GPS System for control of an Off-Road Vehicle,” *Proceedings of the 1999 ION-GPS Meeting*, Nashville, TN, September 1999.