To be presented at the World Automation Congress Conference (WAC 2000), Maui, HI, June 11-16, 2000.

INTELLIGENT CONTROL FOR UNMANNED VEHICLES

KARL MURPHY, MARILYN ABRAMS, STEPHEN BALAKIRSKY, DAVID COOMBS, TSAI HONG, and STEVEN LEGOWIK
National Institute of Standards and Technology (NIST)

TOMMY CHANG and ALBERTO LACAZE
University of Maryland & NIST

ABSTRACT

The NIST robot vehicle, a HMMWV with sensors and computer controlled actuators, detects and avoids obstacles while it drives off road at speeds up to 35 km/h. During tests the vehicle drives through the back fields of NIST detecting large obstacles up to 50 m away. Obstacles are sensed using a 30¡x60¡ field of view laser range scanner. The planner computes smooth, obstacle free paths that follow an operator s commanded path.

KEYWORDS: robot vehicle, LADAR, obstacle detection, path planning, RCS, sensor processing, world modeling, GPS.

INTRODUCTION

Many challenges must be overcome before robot vehicles can drive off road effectively. Most pressing is sensing and modeling the terrain and planning safe paths through it while under the real-time constraints of a fast moving vehicle.

Although challenging, the benefits are substantial, especially for the military where robot vehicles would remove humans from high risk missions. The current autonomous vehicle research effort of the U.S. DOD s Joint Robotics Program is the Demo III program. NIST is supporting the Demo III program in the areas of control architecture, terrain sensing, modeling and path planning.

Mobility algorithms and control modules are developed and tested on the NIST outdoor mobility test bed, a robotic HMMWV. See Fig. 1. The NIST vehicle has electric actuators on the steering, brake, throttle, and transmission. A Kalman filter estimates vehicle position using inertial, dead reckoning, and GPS sensors. To detect obstacles the vehicle uses a LADAR, a laser range scanner, that produces a 128 by 64 pixel range image once per second. The control software is then ported to a different vehicle that is used on the

Demo III program. This vehicle, termed the XUV, is a custom, state of the art platform and has a broader array of sensors than the NIST vehicle.

This work describes the control structure used on the NIST vehicle, how it senses obstacles, creates local obstacle maps, and plans paths through them.

Figure 1. The NIST Robotic HMMWV

CONTROL STRUCTURE

The NIST robot vehicle uses a hierarchical controller that follows the NIST Real-time Control System, RCS, methodology [1]. Sensor processing modules sense the state of the vehicle and the surrounding environment. Sensory data flows to world modeling modules which update the estimated state of the vehicle and its surroundings. The world model is then used by behavior generation modules to plan actions and to execute the resulting plan. Planned paths and actions are stored in the world model and can be used by the sensor processing modules to direct sensor attention or processing cycles to locations in the environment that are more critical.

The current RCS implementation on the vehicle is shown in Fig 2. Sensor processing and world modeling modules are on the left and behavior generation modules are on the right. Four levels of the hierarchy are implemented: Servo, Prim, Autonomous Mobility (AM), and Vehicle levels. A LADAR (a laser scanner) produces a range image from which obstacles are detected. At the subsystem level, the obstacles are placed and tracked in a scrolling map that extends 50 m from the vehicle. Using the current obstacle map, the planner computes the shortest obstacle free path that drives the vehicle to the path commanded by the vehicle level. The AM level cycles at 4 Hz. Lower levels then compute steering and speed commands and servo the electric actuators.

At the vehicle level, sensed obstacles are combined with a priori maps in a 500 m map. The vehicle level planner then selects the lowest cost path that achieves the mission goals that were specified by a human operator. The vehicle level replans at 1 Hz.

The obstacle detection and planning modules are addressed in more detail below.

Obstacle Detection

To detect obstacles, the vehicle uses a rugged commercial LADAR that produces a 128 by 64 pixel range image once per second. The sensor has a maximum range of 50 m and a measurement resolution of 6 cm. This is being upgraded to the LADAR used on the Demo III vehicle, a sensor with similar image size but a significantly faster update rate, up to 60 Hz. Each pixel in the range image is classified as Ground if it is traversable terrain, as Obstacle if it is non-traversable, and Cover if it is an object higher than the

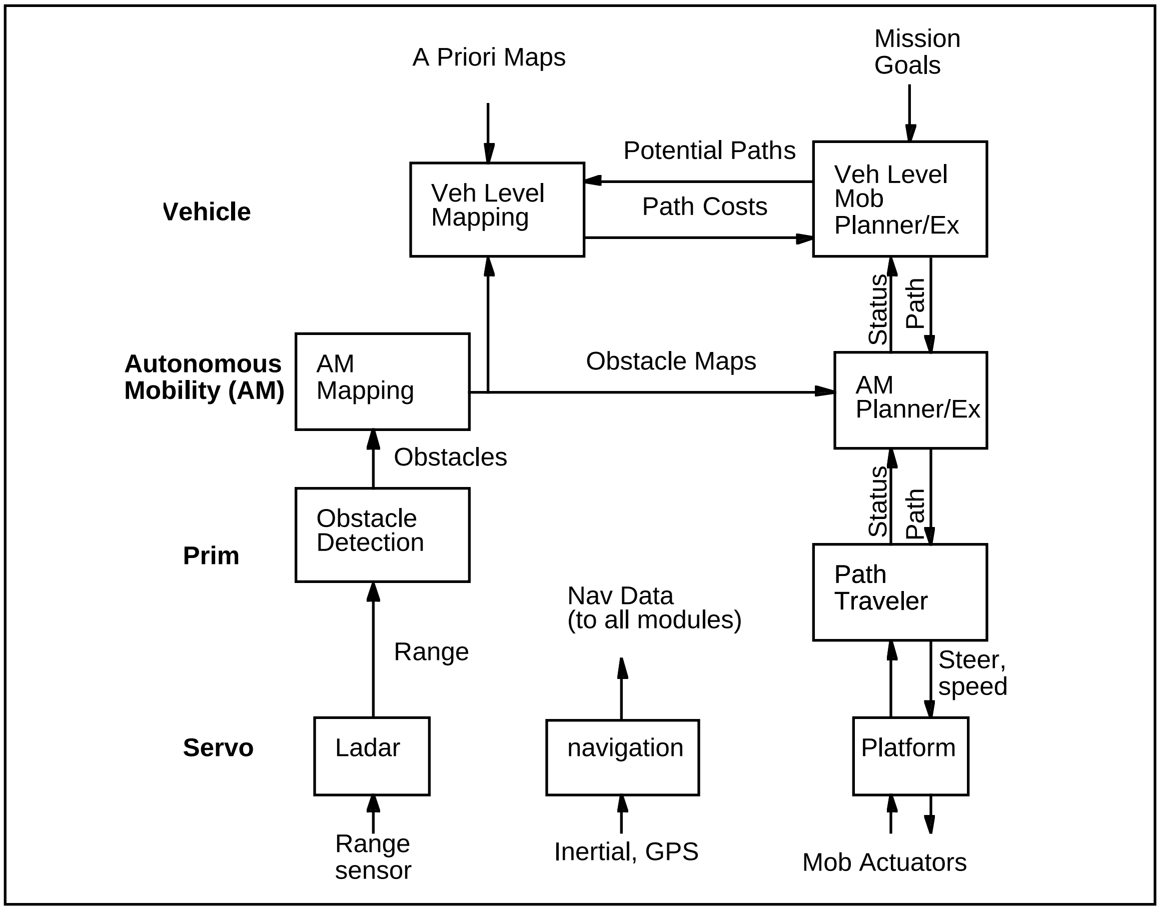

Figure 2. Mobility Software Architecture. A priori data and sensed obstacles feed maps for path planning.

top of the vehicle [2].

Classification is determined by processing each column in the range image independently. The x, y, z coordinates are calculated for each pixel with a valid range. Progressing from the bottom pixel, the height and slope from the closest ground pixel is calculated. If the slope and height is small, the new pixel is classified as ground. If the height and slope are greater than threshold values of 30¡ and 0.3 m respectively, then the new pixel is classified as an obstacle. If the height above ground is greater than the vehicle height, the pixel is classified as cover. See Fig 3. Future work includes identification of other terrain features such as tall grass.

AM Mapping

The elevation and feature classification are placed in a map with 0.4 m square grid cells that extends 50 m from the vehicle [2]. The map is north oriented and scrolls as the vehicle moves. The various features are integrated over time, computing confidence and filtering out spurious false detections. On the Demo III vehicle, data from the various terrain sensors are fused in this mapping process. For each planning cycle, a copy of the obstacle map is rotated from a north-oriented into a vehicle-oriented map and is sent to the planner.

Fig. 4. shows a display used for debugging and analysis. It displays attributes of the world model such as obstacles and cover and displays the current paths.

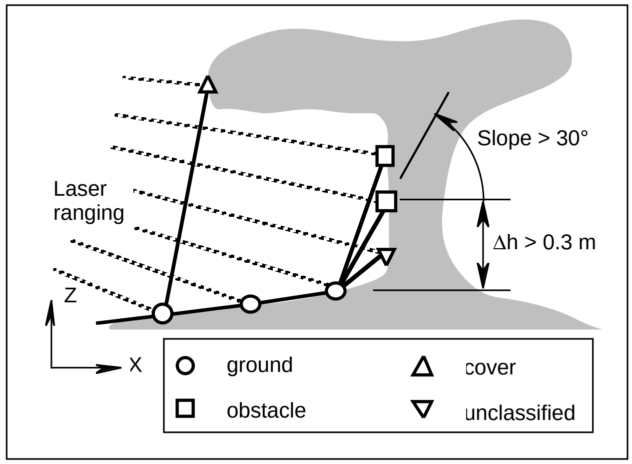

Figure 3. Each column in the laser range image is processed and each range pixel is classified as ground, obstacles, cover, or unknown.

AM Planning

The AM planner selects the shortest obstacle free path that takes the vehicle along its commanded path [3]. To do this, the planner begins with a web of potential path segments that extend out to 50 m. If there are no obstacles, the vehicle can drive on any combination of these path segments.

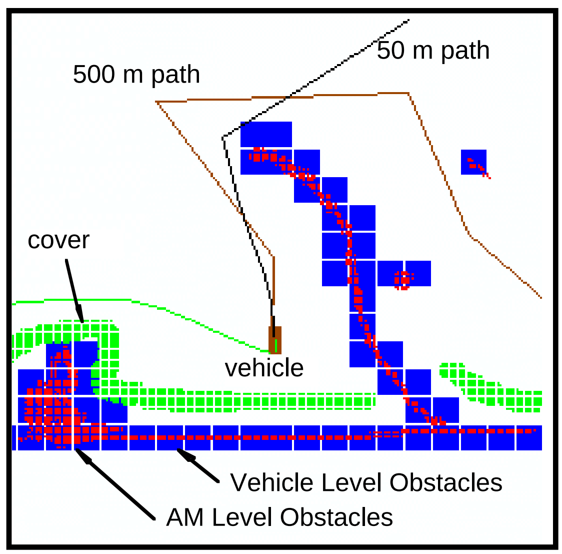

Figure 4. Real-time diagnostic display shows world model attributes and planned paths.

There are two types of path segments, straight and curved. Curved segments extend 20 m from the vehicle. Each is a series of clothoid segments which are kinematically feasible based on the turn rate of the steering wheel. These paths are simulated offline for different initial speeds and steering wheel positions. Initial steering position is a major factor influencing the paths the vehicle can travel. Fig. 5 shows allowable paths with initial steering wheel position to the right and at two different velocities.

Straight path segments are used from 20 m to 50 m. Although not kinematically feasible, they are computationally simpler. At 4 Hz the path will be replanned before the vehicle would reach the straight line path segments. Although smaller obstacles are not seen at these ranges, the straight line path segments will steer the vehicle away from larger obstacles.

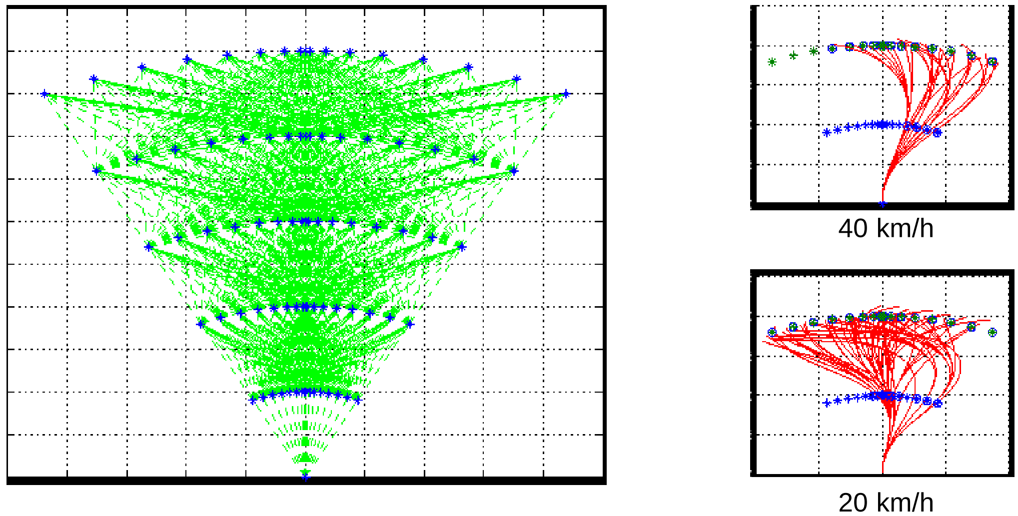

Figure 5. Vehicle oriented paths extend out 50 m. Path segments from 0 to 20 m are smooth paths that depend on velocity and initial steering wheel position.

The planner then selects the best combination of segments leading to the goal. It does this by first pruning all the segments blocked by obstacles. Then it searches through the remaining segments to find the least cost path. Path pruning is optimized by precomputing some of the data. First a curve-to-cell table is computed. For each path segment this table lists the map cells that the vehicle would pass through if it traversed that segment. The width of each swath is at least as wide as the vehicle and is made wider for path segments that will be either driven faster or are further away from the vehicle. This table is inverted to obtain the cell-to-curve table which is later used online to quickly determine which path segments are blocked by obstacles.

Vehicle Level

The vehicle level models the terrain and plans paths out to 500 m [4]. The world model is a multi-layered map with 4 m grid cells. Data comes from both sensed obstacles in the AM level maps and from a priori digital terrain maps. The planner creates a web of potential path segments. The world model then computes the cost to traverse each segment accounting for obstacles, slope, visibility to potential enemy locations, etc. The segments are fixed to the ground and their traversal costs are only updated if they intersect newly sensed obstacles. New segments are added at the leading edge of the map and are removed from the trailing edge as the vehicle moves.

Changes to cost functions used in the world model cause the vehicle to exhibit different behaviors. For example, making roads and smooth terrain low cost would cause high speed routes to be chosen, while making tree lines and low visibility areas low cost would cause stealthy routes to be chosen.

CONCLUSIONS

These techniques enable the vehicle to travel over rolling meadows at speeds up to 35 km/hr (10 m/s) while avoiding obstacles that are well within the vehicle'scurrent sensing capabilities, such as large trees and shrubs. Smaller obstacles are harder to detect and require slower speeds. The higher frame rate LADARs will improve latency, allowing faster speeds.

In general, it is probably desirable for the vehicle to select smoother terrain when possible and to slow down as the path becomes rougher. This will require a more sophisticated terrain analysis and planning model than the binary model used presently. However, the more complex model should enable the system to address other challenges. For many applications, it is necessary to maneuver the vehicle in tight quarters, in some cases even to the point of driving through overhanging branches and pressing through brush. It is hoped that a general solution could address the entire spectrum from high speed driving over smooth terrain to maneuvering in tight quarters on rough terrain.

ACKNOWLEDGMENTS

Development of an intelligent perception and vehicle navigation control system is a jointly sponsored effort which leverages technology from the NIST Intelligent Machines Initiative program and receives financial support from the Army Research Lab Demo III UGV program and the AUTONAV research project agreement between the German Ministry of Defense and the U.S. D.o.D. This work has benefited from innumerable discussions with Maris Juberts, Jim Albus, John Evans, and other Demo III colleagues.

REFERENCES

1. Albus, J.S. "A Theory of Intelligent Systems." *Control and Dynamic Systems*, Vol. 45, 1991. 197-248.
2. Chang, T., Hong, T., Legowik, S., Abrams, M.N. "Concealment and Obstacle Detection for Autonomous Driving," *Proceedings IASTED International Conference on Robotics and Applications*, Santa Barbara, CA. Oct. 28-30, 1999
3. Lacaze, A., Moscovitz, Y., DeClaris, N., Murphy, K. "Path Planning for Autonomous Driving Over Rough Terrain." *Proceedings ISIC/CIRA/ISAS 1998 Conference*, Gaithersburg, MD, Sept. 14-17, 1998
4. Balakirsky, S. and Lacaze, A. World Modeling and Behavior Generation for Autonomous Ground Vehicles." *Proceedings IEEE International Conference on Robotics and Automation - 2000*, San Francisco, April 24-28, 2000