

Integrating Terrain Maps into a Reactive Navigation Strategy

Ayanna Howard*, Barry Werger, Homayoun Seraji

NASA-Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109, USA

*Ayanna.Howard@jpl.nasa.gov

Abstract. This paper presents a new method for integrating terrain maps into a reactive navigation strategy of field mobile robots operating on rough terrain. The method incorporates the Regional Traversability Map, a fuzzy map representation of traversal difficulty of the regional terrain, into the navigation logic. A map-based regional navigation behavior provides speed and direction recommendations based on the current status of the robot. In addition, recommendations from two sensor-based reactive behaviors, local avoid-obstacle and regional traverse-terrain, are fused with the map-based regional behavior to construct a comprehensive navigation system. The algorithms are tested both in graphical simulations and in the field using a commercial Pioneer 2AT robot to demonstrate traversal over rough natural terrain.

I. INTRODUCTION

With a sensor-based reactive navigation strategy, an autonomous mobile robot operates in an unknown environment using only locally-sensed information. This allows actions to be performed without the need to build an exact world model or to use complex planning processes. Although such reactive strategies allow the robot to operate in real-time while requiring minimal memory storage and computational resources, the robot is not guaranteed to select the best path to its goal, or even attain its goal. Situations can arise in which the robot becomes trapped in dead-ends during exploration, unable to reach its desired goal location. This realistic possibility is detrimental to the mission, resulting in lost time and resources, which can ultimately lead to mission failure.

To enhance the performance of sensor-based reactive navigation strategies, prior global terrain information can be used when available. By utilizing a global path planner to specify an optimal path between the start and a desired goal location, the possibility of goal attainment can be evaluated before actual traversal and thus the probability of mission success will be improved significantly.

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

The process of planning global paths for robots operating on harsh, natural terrain is a difficult task. Field robots must traverse harsh *natural* terrain that is uneven, rough, and rugged. The physical properties of the terrain add a new dimension to the complexity of the path planning problem. To ensure mission success, the robot must have the ability to assess the terrain and to determine the risk associated with traversal of each terrain segment. In addition, once a global path is constructed, dynamic changes to the environment can invalidate the original path, and limit the real-time response capability that is necessary for autonomous navigation on unknown terrain. A process thus needs to be implemented that can robustly integrate the benefits of the global path planning methodology with the sensor-based local and regional navigation strategies.

In order to allow successful completion of robotic exploratory missions in high-risk access terrains, we have developed a new method to combine terrain information with a local/regional behavior-based navigation strategy. The method focuses on creating a traversability map, a construct used to represent the ease-of-traversal for a field robot navigating on the terrain [1-2]. A global path planner generates a sequence of waypoints along an optimal path passing through safe regions of the terrain. These waypoints are then integrated with a map-based regional planner that computes desired speed and direction recommendations for the mobile robot. This map-based behavior is integrated with sensor-based regional terrain assessment navigation behavior, as well as with local hazard avoidance behavior, to develop a complete mobile robot navigation scheme capable of traversing long distances on hazardous terrain. Figure 1 provides the architecture of this navigation strategy.

II. BACKGROUND

In this section, we provide a brief overview of existing research focused on the integration of path planning methods with reactive navigation techniques.

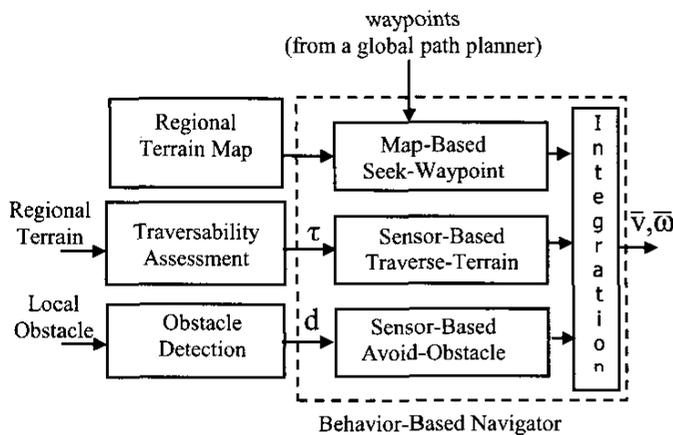


Figure 1. Navigation system architecture

Arkin [3] develops the Autonomous Robot Architecture for integrating path planning and reactive control. In this work, a planning subsystem uses *a priori* maps to construct a global path consisting of linear segments. A reactive navigator is then executed to successfully drive the robot along each segment.

Ma et al [4] construct a strategy for combining reactive behavior control and global path planning. The low-level reactive control process is based on a fuzzy logic control system that receives inputs from a planner, which generates subgoal positions using prior knowledge modeled in a Voronoi diagram. The reactive controller is designed to navigate to a specific subgoal position, avoid obstacles, and follow edges.

In the work of Murphy et al [5], an optimal path planner, the Trulla algorithm, incorporates terrain preferences directly into the navigation logic. The Trulla algorithm represents the world model using a grid construct that contains information regarding direction of the path to the goal, distance to goal, and terrain weighting preference. Based on *a priori* maps, a planner computes all possible paths that lead to the designated goal location. During path execution, a reactive controller selects the optimal path to direct the robot toward the goal, while avoiding obstacles.

Finally, Singh et al [6] describe the D* algorithm in which path planning occurs incrementally and optimally in real-time. In this algorithm, an initial path to the goal position is planned, and the path is updated continuously during the robot traversal by computing and executing a “local patch” to the existing path based on sensory information. For this application, the Morphin algorithm is used for regional navigation.

Although these research efforts focus on integrating reactive behavior with global path planning, only the Trulla and D* algorithms effectively deal with incorporating terrain characteristics, a necessary feature when traversing autonomously on rough terrain. In addition, except for the Trulla and D* algorithms, the global path planner still remains a separate entity from the navigation behaviors. Thus, map-based information, though utilized, is not an integral part of the navigation logic. The main limitation of the Trulla algorithm, however, is that the manner by which the terrain traversability factors are determined is not addressed, and although terrain preference is incorporated in determining the ideal direction to traverse, sensor-based information is not utilized during run-time to validate the traversability factors determined during the path planning process.

III. MAP-BASED SEEK-WAYPOINT BEHAVIOR

In our approach, a global path planner first generates an optimal path passing through safe regions of the terrain. The focus of the global path planning algorithm is to find a path that minimizes our traversal cost function. Given the robot’s initial position and destination, and the traversability grid, an A* search strategy [1] is implemented to determine the sequence of waypoints necessary for goal achievement.

During run-time, the sequence of waypoints generated *a priori* from the optimal path is forwarded to the on-board map-based navigation behavior to direct the robot from a given start to a waypoint location. The map-based regional navigation behavior uses a gradient approach to compute desired speed and direction recommendations for the mobile robot using a perceptual representation of the regional terrain, quality called the Fuzzy Traversability Index [7]. The traversability index is calculated from fuzzy descriptors of terrain characteristics extracted from prior visual imagery data and is represented by the linguistic fuzzy sets {*POOR*, *LOW*, *MODERATE*, *HIGH*}, associated with unsafe, moderately-unsafe, moderately-safe, and safe terrain, respectively.

Once generated, the sequence of waypoints is defined as $\{w_0, w_1, \dots, w_n\}$, where w_0 is the actual robot starting location, and w_n is the designated global goal position. For two consecutive waypoints w_i and w_{i+1} , we define the regional distance D_r as twice the distance between two adjacent waypoints, that is

$$D_r = 2 * \|w_{i+1} - w_i\|$$

The regional distance is used to segment a square region of size $D_r \times D_r$ from the global terrain map such that w_{i+1}

is located at the center of the region. The region is then divided up into equal-sized square cells of dimensions $d_r \times d_r$, where d_r is defined as the maximum physical dimension of the robot. This process produces the *Regional Traversability Map* and provides D_r^2/d_r^2 cells in which to compute speed and direction recommendations² (Figure 2).

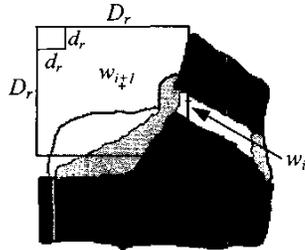


Figure 2. Segmenting the Traversability Map

Traversability	Distance to Waypoint	Speed
LOW	FAR	STOP
MEDIUM	FAR	SLOW
HIGH	FAR	FAST
	NEAR	SLOW
	VERY-NEAR	STOP

Table I: Fuzzy rule-base for speed recommendation

We utilize a flood-fill algorithm [8] to compute speed and direction commands for each cell. First, a traversability cost is calculated for each cell in the Regional Traversability Map. Initially, the cost of traversing each grid cell is set equal to a weighted factor of the reciprocal of the *minimum* traversability index associated with the enclosed terrain area. This weighting factor is utilized to denote the acceptable traversal risk. A large weighting factor represents a conservative approach, with a preference for traversal through safe regions only. A small weighting factor denotes an aggressive approach, with tolerance for traversing through more risky areas.

Starting at the waypoint location w_i , the cost to navigate each cell is updated based on adding in the lowest cost associated with an adjacent neighbor's traversal cost plus the distance traveled to reach that neighbor, that is:

$$\text{cost}_i = \text{cost}_i + \min_j (\text{cost}_j + d_{i,j})$$

where j represents one of the 8 adjacent cells and $d_{i,j}$ is the traversal distance between cell i and cell j .

² The white, light gray, dark gray, and black regions in the image correspond to HIGH, MODERATE, LOW, and POOR traversability indices respectively.

By formulating the cost in this manner, minimal cost paths can be propagated throughout the region, since a cell's cost consists of combining its own traversal cost with the minimal cost associated with an adjacent cell.

Once cell costs are computed, a direction recommendation is calculated for each cell such that the robot is directed toward the adjacent cell having the lowest cost. The speed recommendation is computed using the linguistic rule-base described in Table I, and is represented by the linguistic fuzzy sets {*STOP*, *SLOW*, *FAST*}. Inputs into the speed recommendation rule-base are the cell traversability index and the distance to the waypoint, which is represented by the linguistic fuzzy sets {*VERY-NEAR*, *NEAR*, *FAR*}. When at a distance from the waypoint position, the rule-base computes speed based on the traversability index of the currently occupied cell. As the robot approaches the designated waypoint position, the speed is adjusted accordingly. Note that empty fields in the Table mean that the variable has no effect on the outcome.

Figure 3 shows an example of direction recommendations computed given two consecutive waypoints provided by the global planner. In the Figure, arrows point in the direction of the next recommended cell for traversal.

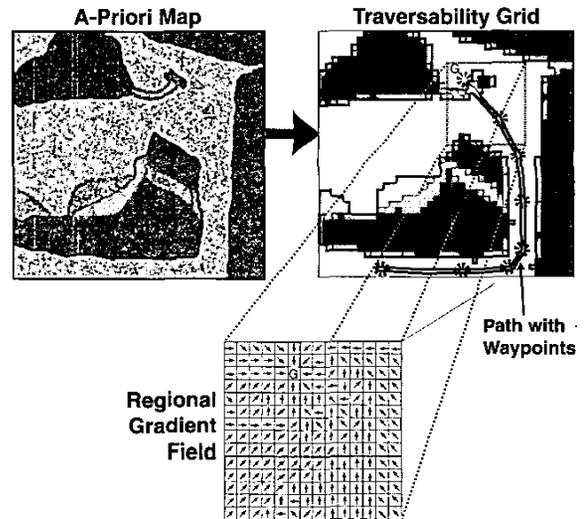


Figure 3. Map region selected and seek-waypoint recommendations computed.

IV. INTEGRATING MAP-BASED AND SENSOR-BASED NAVIGATION BEHAVIORS

The map-based seek-waypoint behavior directs the robot safely from a start location to a designated waypoint position. The sensor-based traverse-terrain behavior

determines the safest regional segment to traverse based on on-board cameras covering the regional terrain up to distance r meters from the robot (where r defines the radius of the regional terrain sensing envelope and is typically 5 meters). The sensor-based avoid-obstacle behavior navigates the robot around local obstacles that impede the robot motion.

Each behavior operates at a different physical range. The traverse-terrain behavior looks at regional terrain information (at approximately 5 meters) to determine the preferred direction and speed necessary to navigate the robot around regional hazards. The avoid-obstacle behavior reacts to immediate hazards and thus operates at a local range of approximately 5-50cm. The seek-waypoint behavior is based on global information, but reacts at a range equal to the robot dimension of d_r . The range factor determines when the recommendation from each behavior is updated, but at any instance in time, all recommendations are integrated in order to construct a unified control strategy for the robot.

The weighting factors g^w , t^w , and a^w represent the strengths by which the seek-waypoint, traverse-terrain, and avoid-obstacle recommendations are taken into account to compute the final motion commands \bar{v} and $\bar{\omega}$, where \bar{v} and $\bar{\omega}$ represent the speed and direction recommendations, respectively. These weights are represented by three linguistic fuzzy sets {LOW, NOMINAL, HIGH} and are computed using the following weight rules:

The traverse-terrain weight rules are as follows:

- IF τ_f is LOW, THEN t^w is HIGH.
- IF τ_f is MEDIUM, THEN t^w is NOMINAL.
- IF τ_f is HIGH, THEN t^w is LOW.

where τ_f denotes the traversability index of the terrain sector facing the current robot heading.

The avoid-obstacle weight rules are as follows:

- IF d_f is VERY-CLOSE, THEN a^w is HIGH.
- IF d_f is CLOSE, THEN a^w is NOMINAL.
- IF d_f is DISTANT, THEN a^w is LOW.

where d_f denotes the distance to the closest front obstacle. Finally, the seek-waypoint weight g^w is set to NOMINAL at all times.

The above weight rules essentially adjust the traverse-terrain and avoid-obstacle weights relative to the seek-waypoint weight in response to the prevailing conditions. Specifically, in critical conditions when the robot is

facing an unsafe terrain segment or a nearby obstacle, the traverse-terrain or avoid-obstacle weighting factors are increased significantly to avoid the impending hazard, at the expense of deviating from the nominal path to the waypoint. Conversely, when the robot faces a safe obstacle-free terrain, the seek-waypoint behavior dominates and drives the robot toward the waypoint.

Once the weighting factors are computed, recommendations from the independent behaviors are blended together to construct the combined, coordinated control actions for the robot using the method described in [9]. The blending method permits more than one behavior to influence the final motion commands. Mathematically, the final motion commands are computed using the Center-of-Gravity defuzzification method as:

$$\bar{v} = \frac{t^w \sum v_p^t A_p^t + a^w \sum v_p^a A_p^a + g^w \sum v_p^g A_p^g}{t^w \sum A_p^t + a^w \sum A_p^a + g^w \sum A_p^g}$$

$$\bar{\omega} = \frac{t^w \sum \omega_p^t B_p^t + a^w \sum \omega_p^a B_p^a + g^w \sum \omega_p^g B_p^g}{t^w \sum B_p^t + a^w \sum B_p^a + g^w \sum B_p^g}$$

where v_p is the translational speed peak value, A_p is the truncated area under the velocity membership function, ω_p is the rotational speed peak value, and B_p is the truncated area under the rotational speed membership function. The final motion commands are utilized to command the robot to safely traverse the natural terrain.

The proposed framework for combining map-based with sensor-based behaviors allows for the construction of an autonomous navigation strategy for long-range robot traversal on hazardous terrain. The navigation approach utilized is highly robust and tolerant to imprecision inherently present in sensing and perception of the natural environment.

V. GRAPHICAL SIMULATION RESULTS

The Robot Simulator is a software tool developed at JPL that provides graphical visualization and testing of the capabilities offered by the navigation methodology described in this paper. A snapshot of the graphical user interface is shown in Figure 4.

The main components of the graphical simulator are:

- A graphical user interface that includes selectable options used by the planner.

- o A terrain image viewer/editor that displays terrain maps and optimum paths.
- o An interface to display the sensor-based navigation path and recommended speed and direction of the robot traverse.

The algorithm is tested in different scenarios and terrain environments. An example simulation run is presented in Figure 5. In this example, the terrain covers an area of 29m x 38m and contains regions of rocky hills, steep slopes, and flat sandy zones. Based on this *a priori* terrain information and a given set of start and goal locations, a global planner computes a set of 8 waypoints $\{w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7\}$, located at 5 meter intervals (Figure 5a), where w_0 is the actual robot starting location and w_7 is the designated global goal position. Figure 5b shows the seek-waypoint recommendation computed for two of the computed waypoints. Notice that in each case, the recommendations direct the robot toward the waypoint until the final goal position is reached for waypoint w_7 .

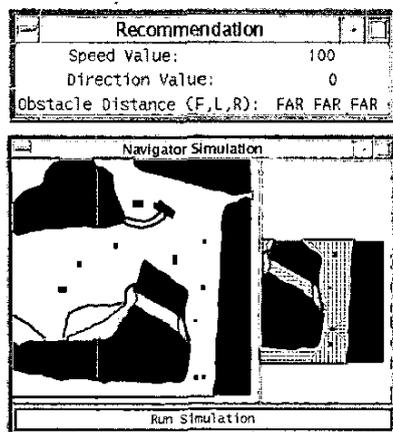


Figure 4. Robot simulator

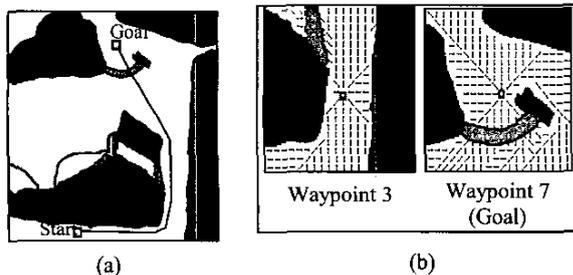


Figure 5. Example simulation run of regional planner. (a) path computed by global planner, (b) map-based seek-waypoint behavior for selected waypoints

VI. FIELD TESTS

Field tests are conducted at the JPL Mars Yard, a 20m x 20m area of simulated Martian terrain composed of sandy soil and rocks of various sizes. For these tests, we used a commercial ActivMedia Pioneer 2AT robot (Figure 6), a four wheel drive skid-steering platform which provides wheel odometry and eight sonar sensors. We supplemented the basic Pioneer with a vision turret containing four USB cameras, an electronic compass, and a notebook computer running the Linux operating system. The notebook computer sends commands to the Pioneer's low-level microcontroller via a serial link.

The four cameras mounted on the Pioneer combine to provide a 180° field-of-view, which is split into the three sectors used for regional terrain assessment as described in [10]. Areas of overlap between camera images is used to provide a "beam" of slope estimation for each sector through stereo matching. Both the vision system and the Pioneer's built-in sonar sensors are used for local obstacle detection. The software system run on the notebook computer is written using TINKU, a C-based port-arbitrated-component-based language [11].

As a skid-steering platform on rough, rocky terrain, the Pioneer is unable to localize effectively using wheel odometry alone. To improve localization, we utilize a hybrid scheme using both wheel odometry and the electronic compass. Odometry is reasonably accurate for forward motion, and worst for rotation. Therefore, we add a level of motor control which translates all motion commands into straight-line segments. The compass is used to orient the robot to the proper direction at the beginning of each segment, and to detect variance from this heading at the end of each segment. Length of each segment is determined through wheel odometry, and the segment information is continuously accumulated to provide position information. In Mars Yard testing, this technique is found to provide an average localization error of approximately 2m over 50m traverses.



Figure 6. The field test rover, a Pioneer 2AT outfitted with high-mounted USB video cameras and compass.

Ten separate trial runs are tested in the Mars Yard (Figure 7), on two sets of start and goal locations that lead to paths of differing complexity. A-priori map information is derived from a three-dimensional laser scan of the Mars Yard (seen in Figure 8, with example paths). The scans do not reflect exactly the current state of the Mars Yard, and thus during traversals the rover must navigate unexpected obstacles and rough terrain areas. In our trials³, many clear examples are seen of regional and local behaviors overriding the global plan in such unforeseen situations. In 90% of these trials with typical 35-50m traverses (including detours around obstacles), the rover arrived within a 2m radius of the goal position.

These trials demonstrate the effectiveness of the navigation system in traversals with a variety of start and goal positions.



Figure 7. The augmented Pioneer 2AT navigates through the JPL Mars Yard.

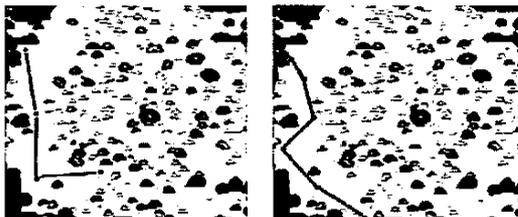


Figure 8. Two paths of differing complexity are superimposed on the a-priori terrain map (laser scan data) of the Mars Yard.

VII. CONCLUSIONS

This paper presents a new approach for incorporating *a priori* terrain map information into the reactive navigation logic for a mobile robot. The terrain map information is used in a map-based seek-waypoint behavior to find a regional path that minimizes a traversal cost. By integrating the sensor-based navigation strategy with the

global terrain map, the mobile robot can travel safely in an uncertain environment without risk from detected obstacles or sensed poor traversability regions. Thus, it is not necessary for the map-based behavior to construct an *exact* path. The mobile robot only requires a sequence of reasonably spaced waypoints in order to reach its designated global goal position. The approach proposed in this paper is validated through computer graphical simulations and is demonstrated in field trials on rough natural terrain.

REFERENCES

1. A. Howard, H. Seraji, B. Werger, "Fuzzy Terrain-Based Path Planning for Planetary Rovers," Proc. IEEE Intern. Conference on Fuzzy Systems, Honolulu, HI, May 2002.
2. H. Seraji, B. Bon, "Multi-Range Traversability Indices for Terrain-Based Navigation," Proc. IEEE Intern. Conference on Robotics and Automation, vol. 3, pp. 2674-2681, Washington, D.C., May 2002.
3. R. Arkin, "Integrating Behavioral, Perceptual, and World Knowledge in Reactive Navigation," Robotics and Autonomous Systems, 6 (1990), pp. 105-122.
4. C. Ma, W. Li, L. Liu, "Mobile Robot Motion by Integration of Low-Level Behavior Control and High-Level Global Planning," Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics, pp. 310-315, 1996
5. R. Murphy, K. Hughes, E. Noll, "An Explicit Path Planner to Facilitate Reactive Control and Terrain Preferences," Proc. IEEE Int. Conf. on Robotics and Automation, vol. 3, pp. 2067-2072, 1996.
6. Singh et al, "Recent Progress in Local and Global Traversability for Planetary Rovers," Proc. IEEE Intern. Conf. on Robotics and Automation, vol. 2, pp. 1194-1200, San Francisco, CA, April 2000.
7. H. Seraji, "Fuzzy Traversability Index: A New Concept for Terrain-Based Navigation," Journal of Robotic Systems, vol. 17, no. 2, pp. 75-91, 2000.
8. J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, Computer Graphics: Principles and Practice, Addison-Wesley, Reading, 1990.
9. H. Seraji and A. Howard: "Behavior-Based Robot Navigation on Challenging Terrain: A Fuzzy Logic Approach", IEEE Transactions on Robotics & Automation, vol. 18, no. 3, pp 308-321, 2002.
10. A. Howard and H. Seraji, "Vision-based terrain characterization and traversability assessment," Journal of Robotic Systems, vol. 18, no. 10, pp. 577-587, 2001.
11. B. B. Werger, "Ayllu: Distributed Port-Arbitrated Behavior-Based Control," in Distributed Autonomous Robotic Systems 4, L. E. Parker, G. Bekey, and J. Barhen (eds.), Springer, 2000, pp. 25-34.

³ Videos are publicly available at <http://telerobotics.jpl.nasa.gov/people/werger/snvttrials>