Experimental Results from a Terrain Adaptive Navigation System for Planetary Rovers

Daniel Helmick, Anelia Angelova, Larry Matthies
Jet Propulsion Laboratory
firstname.lastname@jpl.nasa.gov

Chris Brooks, Ibrahim Halatci, Steve Dubowsky, Karl Iagnemma
Massachusetts Institute of Technology
kdi@mit.edu

Abstract

Results from the experimental testing of a navigation system for planetary rovers called Terrain Adaptive Navigation (TANav) are shown here. This system was designed to enable greater access to and more robust operations in terrains with widely varying slippage. The system achieves this goal by using onboard stereo cameras to remotely classify terrain, predict the slippage of that terrain, and use this information in the planning of a path to the goal. An end-to-end onboard demonstration of the system in a Mars analog environment is shown with promising results.

1. Introduction

This paper describes the results from an end-to-end onboard demonstration of a navigation system for planetary rovers called Terrain Adaptive Navigation (TANav). Unlike the current state of the art in planetary rover navigation [4,11] this system is capable of using more than just terrain geometry to navigate; it is also able to navigate while taking into account potential rover slippage due to varying terrain types. This system has the capacity to learn the rover’s dynamic response to the terrain mechanics and to use this knowledge in the navigation architecture.

Many scientifically interesting sites are located in very steep, very rough terrain with the potential for high slippage and the danger of the rover being immobilized by terrain hazards, possibly a mission ending scenario. The “follow the water” strategy taken by NASA’s Mars Exploration Program inherently requires access to demanding terrains such as dry river channel systems, putative shorelines, and gullies emanating from canyon walls [3]. The TANav system is designed to greatly increase access to and enable more robust operations in this type of terrain.

In this paper we discuss a summary of the high-level architecture of this system. A much more thorough discussion of the system can be found in [6]. The main contribution of this paper is to detail the experimental results from a research rover in a Mars analog terrain with an onboard implementation of the TANav system and to demonstrate a proof-of-concept of this system. In summary, this system uses stereo camera pairs to remotely classify the surrounding terrain, predict the slippage that the rover would experience while traversing that terrain (using learned models), plan a path through this terrain (considering the effects of slippage, terrain geometry, terrain class, and rover dynamics), and finally, precisely follow this path while actively compensating for slippage and reaching the desired goal.

Section 2 briefly describes the architecture and algorithms of the TANav system that were implemented and tested. Section 3 describes the experimental design, including discussion of rover hardware (e.g. sensing, computation, control, mechanisms, etc.) (see Figure 1), software implementation details, and the Mars analog environment that was used during these experiments. Section 4 describes the results from each of the steps of the system during an end-to-end experiment and as a whole represents a proof-of-concept demonstration of the TANav system. This section also compares the results of the TANav experiment to a control
An experiment that does not use this system to demonstrate the advantages of such a system. Section 5 discusses conclusions drawn from the experimental results and Section 6 discusses future work in this area of research.

### 2. System Architecture

As described above the TANav system is designed to enable navigation over steep terrain that potentially has very high slippage. A block diagram of this system is shown in Figure 2. The various subsystems of this diagram are described below. A more detailed description of this architecture can be found in [6].

The terrain classification algorithm relies on visual features drawn from color, texture, and range to classify terrain regions at a distance into a set of distinct terrain types with potentially significantly varying terrain mechanical properties. A supervised classification architecture based on a support vector machine has been implemented for each feature type. The results of these “low-level” classifiers are merged in a Bayesian fusion approach to yield an accurate final class assignment. Further details about the terrain classification algorithm can be found in [5].

The slip prediction algorithm learns a slip model (rover slip in longitudinal, lateral, and yaw directions as a function of each of the components of terrain slope -- pitch and roll) for each of the terrain classes determined by the terrain classification algorithm. It does this by measuring slippage onboard the rover as it traverses over classified terrain and fitting a model to the vehicle response. Slippage is measured by comparing wheel odometry to visual odometry. After these models have been learned, the rover is able to predict slip by inputting the slope (remotely measured using stereo imagery) of a classified terrain into the model, which then outputs a slippage estimate for that location of the terrain. A much more detailed description of this algorithm can be found in [1,2].

A slip-augmented goodness map is then generated by using a weighted combination of geometric goodness (determined from slopes, step heights, surface roughness, etc.) and a slip derived goodness estimate for each cell of the map. A path planning algorithm (as described in [13]) is then used to plan an initial path through the slip-augmented cost map (cost map = 1 - goodness map). After the initial path is planned, a “terrain triage” algorithm is performed on the slip-augmented goodness map to determine if any part of the path goes through terrain that is not known to be easily traversable nor known to be extremely difficult to traverse; in other words, the traversability is “uncertain.” The main purpose of terrain triage is to allow this system to scale (in computational cost) with the difficulty of the terrain. If any part of the path goes through “uncertain” terrain (terrain that is neither “definitely traversable” nor “definitely not traversable”) then the High-Fidelity Traversability Analysis (HFTA) algorithm is invoked to refine the path through these regions. The HFTA algorithm uses a dynamic rover simulator called ROAMS (Rover Analysis, Modeling and Simulation) [8] to do a full kinematic/dynamic forward simulation of the rover performing slip-compensated path following over terrain generated from both stereo data and outputs from the terrain classifier (used to define mechanical terrain properties such as cohesion, density, and internal friction angle). The purpose of this forward simulation is to evaluate, with high-fidelity, the cost of the rover traversing a path. This cost evaluation is then used to find the lowest cost path through the “uncertain” regions, which is then used to modify the original planned path.

Once the final path is determined, it is used by the slip-compensated path follower which closes the loop on visual odometry, compensates for slippage, and precisely follows the path [7].

### 3. Experimental Design

The experimental design consists of three major components: the rover hardware (see Figure 1), onboard software, and the Mars analog terrain (see Figure 3).

The rover hardware was designed to allow for rapid software development and convenient, practical usage while still remaining representative of flight hardware.
that has been and will be used on planetary missions in the future. Mechanically, it is a six-wheeled rover with a rocker/bogie suspension that is kinematically very similar to Sojourner, MER, and MSL [9,10]. The onboard avionics consist of a standard PC-104 stack with a 1.8GHz Pentium-M processor, commercial-off-the-shelf (COTS) distributed motor controllers, fourteen brushless motors to actuate six steerable/drivable wheels and a pan/tilt mast, and eight cameras (six b/w and two color). The cameras are configured to make four stereo pairs consisting of front and rear “hazard avoidance” camera pairs, and “navigation” and “panorama” camera pairs on the mast.

The onboard software runs as several separate processes on a standard Debian Linux distribution. One process is dedicated to coordinated motor commanding, kinematics, and slip-compensated path following and runs at ~20 Hz. Another process is dedicated to image capture, stereo processing, and visual odometry and runs at ~1 Hz. Several other processes are used during each planning cycle which occurs every 10-50 meters of traversal and consists of goodness map generation, terrain classification, slip prediction, path planning, and HFTA.

The Mars analog terrain is a widely used facility at JPL called the Mars Yard. It is a large (~55m x 45m) area consisting of multiple terrain types (various rock sizes and distributions, multiple sand/soil types, and several areas of bedrock simulant) and various terrain geometries (including flat ground and slopes up to ~30º).

4. Results

Results from experiments onboard a research rover in a Mars analog environment are shown here. Two experiments were run: the first was a demonstration of the complete TANav system and the second was a control experiment that only used terrain geometry in its path planning. The comparison of these two experiments demonstrates the advantages that can be gained by using the TANav system. Both experiments started with the same stereo panorama from the mast. In both cases a goodness map was generated from this panorama. For the TANav experiment the local terrain was classified, and slip was estimated for the surrounding area. Then a slip-augmented cost map was generated and a path was planned through this map. Next, terrain triage and the HFTA algorithm were performed to refine this path. For the control experiment a path was planned through the geometric goodness map. Finally, for both experiments, the rover performed slip-compensated path following on the corresponding path. Results from each of these steps are described below.

4.1. Stereo Panorama

The experiments began with the rover taking a panorama of the surrounding area with its panoramic cameras. This panorama consisted of five stereo pair color images each with approximately 20% overlap between images (see Figure 3). From this panorama, a user designated the desired goal in the rover frame (x is forward and y is to the right) of the traverse (see Figure 4). In this case the goal was chosen to be at x=9.2 and y=2.6 meters.

4.2. Goodness Map

Five point clouds were generated from each stereo pair of the panorama. These point clouds were merged and transformed to the rover frame using the pan/tilt angles of each of the image pairs. This merged point cloud was then used to generate a goodness map (using the algorithm described in Section 2 and [4,11]) with 10 cm grid cells covering an area 15 x 30 meters in front of the rover (see Figure 5). As can be seen in the goodness map, geometric hazards (the rock field to the left and the steep slopes to the right as can be seen in the panorama, Figure 3) are accounted for in this map with lower goodness (red) cells and the flatter areas without rocks are shown with higher goodness (green).

4.3. Terrain Classification

Terrain classification was then performed onboard the rover on the panoramic imagery using the algorithm described in Section 2 and [5]. The results of this terrain classification are shown in Figure 6. As can be seen, the classifier was trained (using different panorama camera images from different perspectives of the Mars Yard) using three terrain classes (bedrock, soil, and rock). There are two large patches of bedrock on the right hand side that have accurately been classified (see the right hand side of Figure 6 for an image space overlay) and the patches of rock on the left are also accurately classified. For the entire panorama, the classification accuracy is 88.5% correct. The classification accuracy is calculated as (Nc/Nt), where Nc is the number of terrain cells where the classification result matched the hand-labeled class, and Nt is the total number of hand-labeled terrain cells. Each terrain cell is 20 x 20 cm.
Figure 3 – Color panorama of Mars Yard taken by Pluto panoramic cameras (flat terrain on the left and slopes reaching 25° on the right)

Figure 4 – Overhead projection of panorama with start and goal points

Figure 5 – Geometric goodness map

Figure 6 – Terrain classification results (overhead map on the left and image overlay (from the right-most panoramic image) on the right)
Figure 7 – Slip goodness map

Figure 8 – Slip-augmented goodness map

Figure 9 – Planned path with TANav

Figure 10 – Planned path without TANav

Figure 11 – Terrain triage map

Figure 12 – Randomly generated paths through “uncertain” region
Table 1 – HFTA evaluated path costs

<table>
<thead>
<tr>
<th>Iteration</th>
<th>energy (J)</th>
<th>path error (m-s)</th>
<th>weighted cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>181.229</td>
<td>3.06218</td>
<td>211.8508</td>
</tr>
<tr>
<td>2</td>
<td>233.865</td>
<td>2.10263</td>
<td>254.8913</td>
</tr>
<tr>
<td>3</td>
<td>181.902</td>
<td>1.70714</td>
<td>198.9734</td>
</tr>
<tr>
<td>4</td>
<td>154.766</td>
<td>7.40014</td>
<td>228.7674</td>
</tr>
<tr>
<td>5</td>
<td>185.901</td>
<td>3.52733</td>
<td>221.1743</td>
</tr>
<tr>
<td>6</td>
<td>226.213</td>
<td>3.30995</td>
<td>259.3125</td>
</tr>
<tr>
<td>7</td>
<td>183.962</td>
<td>4.77294</td>
<td>231.6914</td>
</tr>
<tr>
<td>8</td>
<td>178.86</td>
<td>5.18531</td>
<td>230.7131</td>
</tr>
<tr>
<td>9</td>
<td>162.279</td>
<td>6.58719</td>
<td>228.1509</td>
</tr>
<tr>
<td>10</td>
<td>181.338</td>
<td>8.03111</td>
<td>261.6691</td>
</tr>
<tr>
<td>11</td>
<td>233.865</td>
<td>2.63038</td>
<td>260.1688</td>
</tr>
<tr>
<td>12</td>
<td>222.272</td>
<td>2.62189</td>
<td>248.4909</td>
</tr>
<tr>
<td>13</td>
<td>182.854</td>
<td>7.78614</td>
<td>260.7154</td>
</tr>
<tr>
<td>14</td>
<td>162.238</td>
<td>6.31429</td>
<td>225.3809</td>
</tr>
</tbody>
</table>

Figure 13 – Stereo-generated HFTA terrain

Figure 14 – Path following results with TANav

Figure 15 – Path following results without TANav
4.4. Slip Prediction

Slip prediction was then performed onboard the rover. Slip models for the three terrain classes were learned offboard using training data from many traverses over the Mars Yard terrain. Using these learned models and the results of the terrain classification and the stereo range data (to remotely determine terrain slopes) as inputs, a slip goodness map was generated (see Figure 7). This slip goodness map predicts a normalized value of rover slip (where lower goodness means higher slip and vice versa) for each grid in the map for which there is range and terrain class data available.

As can be seen in the slip goodness map, the lower sloped terrain on the left (of both the rocks and the soil) shows higher goodness values (lower slip) and the higher slope terrain on the right (corresponding to the areas of soil) show lower goodness values (higher slip). The two areas of bedrock on the right (in the near and far field) show distinctly higher goodness even though the slopes are high (20-25°). This corresponds to the learned models of bedrock which exhibit extremely small slip responses even for high slopes.

4.5. Slip-Augmented Goodness Map

A slip-augmented goodness map (see Figure 8) was generated by using a linear combination of the geometric goodness map and the slip goodness map. It is important to note that the slip goodness map cannot increase the goodness of any cell of the geometric goodness map (and vice versa) so that geometric and slip hazards are both preserved in the final slip-augmented goodness map.

4.6. Path Planning

Given the slip-augmented cost map (1 - slip-augmented goodness map), the user designated goal, and the current location of the rover, an optimal path was planned for the TANav experiment (see Figure 9). For the control experiment, a path was planned using just the geometric cost map (see Figure 10). As can be seen, the path using the geometric cost map goes through the lower slope terrain on the left before curving up to the goal at the end of the path. The path using the slip-augmented cost map takes a higher slope, lower slip path to the right, over the bedrock, and then curves down to the goal at the end of the path.

4.7. Terrain Triage

The terrain triage algorithm was then applied to the slip-augmented goodness map to create a triage map (see Figure 11). The part of the path circled in blue passes through an “uncertain” region of the triage map, therefore this length of the path was used as an input to the HFTA algorithm.

4.8. High Fidelity Traversability Analysis

Using the beginning and end points of the length of the “uncertain” path, many paths were randomly generated using several simple heuristic rules (monotonic growth in x and y, limited distance per step, etc.) that improved coverage of the space between the beginning and end points (see Figure 12). Each of these paths was then run through the HFTA forward simulator onboard the rover (over the stereo generated terrain shown in Figure 13) with the cost results from each path shown in Table 1. The path shown in red (in the table and the figure) had the lowest combined cost and therefore was the final path sent to the slip-compensated path following algorithm.

4.9. Slip-Compensated Path Following

Each of the two paths generated above (with and without TANav) were followed by the slip-compensation path follower in the Mars Yard. The results from each of these traverses are shown in Figure 14 and Figure 15. In each of these figures the red line represents the planned path, the green line represents the output from wheel odometry and the blue line represents the actual path traversed. The difference between wheel odometry and visual odometry is the slippage of the vehicle over the traverse. The traverse of the path using TANav experienced a total of 1.78 meters of slippage (lateral and longitudinal) and successfully reached the goal. The traverse of the path that didn’t use TANav experienced 4.78 meters of slippage before the rover stalled due to excess slippage without ever having made it to the goal.

5. Conclusions

We have shown here the results from an onboard demonstration of the TANav system compared to those of a navigation system that doesn’t use information about terrain properties. As can be seen from these results, the TANav system and the general
concept of using slip estimation and terrain properties in a navigation system has the potential to significantly increase the robustness of rover navigation in high-slip environments. TANav not only improved the performance of the system (by decreasing the total slip of the rover over the traverse) but it also enabled access to the goal by finding a path that was actually traversable. This is opposed to the system that did not use TANav, which resulted in a non-viable path that was not detected until traversal, which could result in a mission ending scenario. Although this is not a rigorous evaluation of this system, it is an end-to-end proof-of-concept demonstration of this technology with very promising results.

6. Future Work

There are several areas of potential improvements that would be good next steps for this research. One of these is to wrap the HFTA algorithm with a black box optimization algorithm such as Nelder-Mead [12] in place of the random path generator. A second area of improvement would be to integrate directional slip prediction (which is currently done in the slip prediction step) into the path planning algorithm, which would enable higher-level behaviors to emerge such as planning a path with switch-backs up a slope.

7. Acknowledgments

The research described in this publication was carried out at the Jet Propulsion Laboratory, California Institute of Technology under contract from the National Aeronautics and Space Administration (NASA), with funding from the Mars Technology Program, NASA Science Mission Directorate. Special thanks go to Max Bajracharya who implemented a very large portion of the onboard software system infrastructure as well as having made large contributions to the avionics design. Thanks also go to Dan Clouse who played an important role in the implementation of the HFTA algorithm. Additional thanks go to Andrew Howard and Steve Goldberg who implemented many of the vision algorithms (including stereo and visual odometry) and image capture code.

8. References