

Terrain Adaptive Navigation for Mars Rovers

Daniel M. Helmick, Anelia Angelova, Matthew Livianu, and Larry H. Matthies
Jet Propulsion Laboratory
4800 Oak Grove Dr.
Pasadena, CA 91106, USA
818-354-3226
firstname.lastname@jpl.nasa.gov

Abstract—A navigation system for Mars rovers in very rough terrain has been designed, implemented, and tested on a research rover in Mars analog terrain. This navigation system consists of several technologies that are integrated to increase the capabilities compared to current rover navigation algorithms. These technologies include: goodness maps and terrain triage, terrain classification, remote slip prediction, path planning, high-fidelity traversability analysis (HFTA), and slip-compensated path following. The focus of this paper is not on the component technologies, but rather on the integration of these components. Results from the onboard integration of several of the key technologies described here are shown. Additionally, the results from independent demonstrations of several of these technologies are shown. Future work will include the demonstration of the entire integrated system described here.^{1 2}

TABLE OF CONTENTS

1	INTRODUCTION	1
2	SYSTEM ARCHITECTURE	2
3	GOODNESS MAP/TERRAIN TRIAGE	3
4	TERRAIN CLASSIFICATION/SLIP PREDICTION ..	4
5	PATH PLANNING	5
6	HIGH-FIDELITY TRAVERSABILITY ANALYSIS ..	5
7	SLIP-COMPENSATED PATH FOLLOWING	7
8	RESULTS	7
9	CONCLUSIONS	9
10	FUTURE WORK	9
11	ACKNOWLEDGMENTS	10

1. INTRODUCTION

In this paper a navigation system designed for a Mars rover in rough terrain is described. The current navigation system running on the Mars Exploration Rovers (MER) [6], [16], was designed for relatively benign terrains and does not explicitly account for terrain types or potential slip when evaluating or executing paths.

It is well recognized that many scientifically interesting sites on Mars are in very rough terrains with the potential for significant slippage. 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 [5]. Therefore, it is important that the next generation of Mars rovers have the capability to autonomously navigate through these terrains, not only to increase the science return efficiency, but also to enable access to previously inaccessible science sites. The navigation system described here is designed to deal with these scenarios using more sophisticated (and thus more computationally expensive) terrain analysis; however, this system is also designed to converge to computational complexity similar to that of currently deployed navigation systems when the terrain is benign. This navigation system consists of several technologies that have been developed, integrated, and tested onboard research rovers in Mars analog terrains (see Figure 1). These technologies include: goodness maps and terrain triage, terrain classification, remote slip prediction, path planning, high-fidelity traversability analysis (HFTA), and slip-compensated path following.

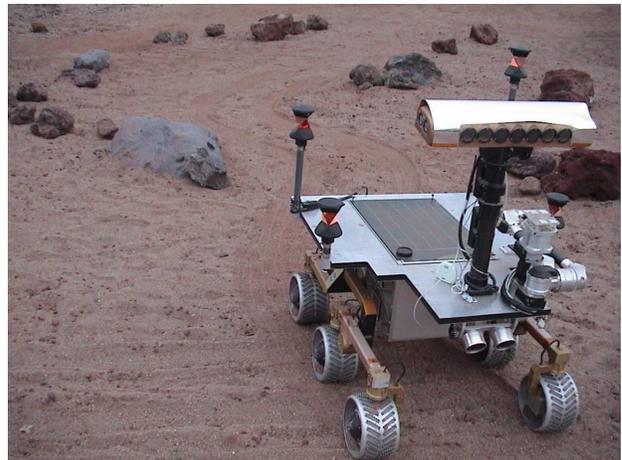


Figure 1. Rocky8 in the JPL Mars Yard

Section 2 discusses the system architecture as a whole, including design goals and operational assumptions. It explains how the subsystems interact to create a navigation system. It also discusses the interfaces between the component technologies.

Section 3 describes the generation of goodness maps from Navcam imagery [6], [16]. Also described in this section is the terrain triage algorithm. This is a technique to sub-divide

¹ 1-4244-0525-4/07/\$20.00/© 2007 IEEE

² IEEEC paper # 1668

the terrain into three categories based on the goodness calculation. The three categories are: definitely traversable, definitely not traversable, and uncertain. This categorization is then used to determine which parts of the future planned path need to be analyzed in more detail.

Section 4 discusses the remote slip prediction algorithm, which uses intensity and range data from stereo cameras to predict the slip of the rover on terrain at a distance. It implements learned, non-linear regression models that output rover slip, using terrain geometry from stereo imagery as input. This rover slip is then used to augment the cost map (essentially “1 - goodness map”). This section also briefly discusses the role of a terrain classifier in slip prediction.

Path planning uses the D* algorithm [21] to determine an optimal path to a goal through the cost map. This algorithm is briefly discussed in Section 5. It is beyond the scope of this paper to discuss this algorithm in detail, and the focus of this section is how the path planner fits into the system.

Section 6 explains the high-fidelity traversability analysis (HFTA) algorithm, which uses a sophisticated kinematic and dynamic forward simulation [10] of the rover following a path. It is designed to calculate a more accurate and realistic cost to traverse that path. The simulation includes a detailed geometric and mass model of the rover, terrain generated from onboard stereo imagery, a dynamic model of the wheel terrain interaction (with parameters based on the terrain classification outputs), and the same slip-compensated path following algorithm (see Section 7) that runs onboard the rover. The results of the HFTA are then used to refine the planned path of the rover through the uncertain region(s).

Once the final planned path is created, the slip-compensated path following algorithm (see Section 7) is invoked to enable the rover to actually follow this path regardless of the slip [9], [7]. This algorithm compares visual odometry (VO) (a technique that measures rover motion including slippage) [15] and vehicle kinematics (a technique that measures rover motion minus slippage) to estimate the location and the slippage of the rover; it then compensates for this slippage and accurately follows the desired path to the goal.

Results from the integration and demonstration of several of the key technologies onboard a research rover (Rocky8) in a Mars analog terrain (see Figure 1) are shown. Additionally, the results from independent demonstrations of several of these technologies are shown. Future work will include the demonstration of the entire integrated system on a research rover.

2. SYSTEM ARCHITECTURE

Figure 2 is a block diagram showing the system architecture described in this paper. The different colors represent functional groups of this architecture: red represents sensing, green represents mapping and terrain analysis, yellow repre-

sents path planning, and blue represents path following.

The top block is navigation camera (Navcam) imagery. The Navcams on the research rovers [11], the MER rovers [13], and the MSL [14] rover are all stereo cameras mounted on pan/tilt masts. These cameras allow the rover to take panoramic images from a high perspective, which decreases obscurations, thus enabling terrain sensing at further distances. Typical Navcam configurations are shown in Table 1. This configuration allows for stereo ranging at distances up to 50-100 meters [13]; and with the pan/tilt capability, range information spanning 360 degrees can be accurately registered into a single map. Alternatively, Pancams could be used in place of the Navcams (see Table 1). Using the MSL Pancams at their maximum zoom (shortest focal length), errors of less than 20 cm at 50 meters range should be possible [14]. Other techniques such as wide-baseline stereo could increase the range and the accuracy even further [17] [18].

The assumed operational scenario of this navigation architecture is that a goal is designated within stereo range of the rover. The maximum distance for goal designation is a function of stereo range error at that distance, vehicle pose estimation error at that distance, and acceptable path error. Given the MSL Pancam range error (see Table 1) and the fact that pose error is 1-2% of distance traveled [7], it is feasible that goals of up to 100 meters would be acceptable, but more conservative goals of 20 to 50 meters away are more likely.

This goal designation could be achieved in several ways. It could be designated by a human operator from Navcam imagery. It could be designated by a human operator from orbital imagery. It could be one of many “global waypoints” all designated from orbital imagery to define extremely long traverses (kilometers). In this case the constraint would be on the spacing of the waypoints to be within the stereo range of the rover. It also could be autonomously designated onboard using both orbital and local sensor data.

Once a goal is designated, a Navcam panorama is taken in the general direction of the goal (if a Navcam panorama has not already been taken for goal designation). Stereo is then done on each pair of the Navcam images and registered into a map using the pan/tilt angles. Once the stereo point cloud data are registered, a goodness map and a triage map are generated using planar statistics on the terrain geometry (as described in Section 3). The goodness map is then augmented with slip prediction costs (Section 4), re-triaged, and then passed to the path planner. The path planner plans an optimal path from the current rover location to the designated goal (Section 5). If all of the planned path goes through “definitely traversable” terrain, then the path is passed to the slip-compensated path follower. If any part of the path goes through “uncertain” regions of the triage map then the HFTA is performed on those regions to obtain a more realistic cost of traversal and to refine the path in that region. The path is then passed to the path follower and the rover follows the path until it reaches the



Figure 2. System Architecture Block Diagram

Table 1. MER/MSL Navcam and Pancam Configurations (range error assumes 0.25 pixel stereo correlation accuracy)

	MER/MSL Navcams	MER Pancams	MSL Pancams
baseline (meters)	0.20	0.30	0.20
camera resolution (pixels)	1024×1024	1024×1024	1200×1200
camera field of view (FOV)	45.0° × 45.0°	16.0° × 16.0°	6.0° × 6.0° - 50.0° × 50.0°
range error at 50 meters (meters)	2.5	0.60	0.20 (at max. zoom)

goal.

As mentioned in the introduction, one of the goals in the design of this architecture was to create a rover navigation system that would be capable of navigating through rough, high-slip terrains, but which would converge to the computational complexity of simpler algorithms in benign terrain. This is enabled using the terrain triage algorithm described in Section 3.

3. GOODNESS MAP/TERRAIN TRIAGE

A goodness map is a regularly spaced grid representing the local region around the rover. The map is populated using stereo data generated from Navcam imagery. In each cell of the map, a goodness value is calculated using the stereo

data that falls in and around that cell. A plane is then fit to a rover sized patch of cells. The goodness calculation involves metrics such as pitch and roll of the plane, roughness of the terrain (standard deviation of the plane fit), and step heights within the patch [6].

Terrain triage is a simple concept that is fundamental to reducing the computational complexity of this navigation architecture in benign terrains. The idea is to categorize the terrain into three categories: “definitely traversable”, “definitely not traversable”, and “uncertain”. This categorization is done by simply thresholding each of the goodness values of the goodness map and thus binning each of the cells into one of the three categories. This step is performed twice: once before the slip prediction and once after. The first time is to determine whether or not slip prediction needs to be

performed for a particular cell in the map. If a cell is categorized as “definitely not traversable” then there is no need to do slip prediction for that cell, because the slip prediction algorithm can only increase the cost of the cell. The second time terrain triage is performed is to incorporate the new costs from the slip prediction into the cost map. Now, when a path is planned through the cost map (as described in Section 5), if the planned path travels entirely through “definitely traversable” terrain, then this path is deemed acceptable and it is passed to the slip-compensated path follower without any further analysis. This will only happen in relatively benign terrain. If, however, the path passes through “uncertain” terrain, then the HFTA (see Section 6) is invoked on those sections of the path.

4. TERRAIN CLASSIFICATION/SLIP PREDICTION

An independent assessment of the terrain traversability is done in terms of rover slippage. Slip is a measure of the lack of progress or the lack of mobility of the rover on a certain terrain. It is defined as the difference between the commanded velocity and the actual achieved velocity in each DOF of the rover [7]. It is normalized by the commanded velocity [24] and for convenience will be expressed in percent.

Rover slippage has been recognized to be a significant limiting factor for the MER rovers while driving on steep slopes [4], [12]. Knowing the amount of slip beforehand and being able to detect areas of large slip will prevent the rover from getting stuck in dangerous terrain and will enable more intelligent path planning. Slip prediction is needed in addition to an obstacle detection mechanism because an area of large slip is a *non-geometric* type of obstacle and cannot be detected with a standard obstacle avoidance algorithm such as GESTALT [6].

Main Method and Architecture

While detecting rover slippage is relatively straightforward [7], [19], the main challenge here is that the rover slip needs to be known remotely, before the rover actually traverses a particular location, in order to enable safe avoidance of areas of large slip. We have proposed an algorithm which infers the amount of slip on the upcoming terrain using visual information and onboard sensors (e.g. a tilt sensor or the IMU) and have shown successful slip prediction results from only these remote sensors [1], [2]. The problem is approached by learning from previous examples. To tackle the problem of slip prediction from a distance, we subdivide it into first recognizing the soil type the rover is going to traverse and then predicting the amount of slip as a function of terrain geometry, i.e. slopes [1], [2].

The main architecture of the slip prediction algorithm is given in Figure 3. For clarity, we first describe the prediction part of the algorithm, assuming the terrain classifier and the slip models have already been learned. The slip prediction mod-

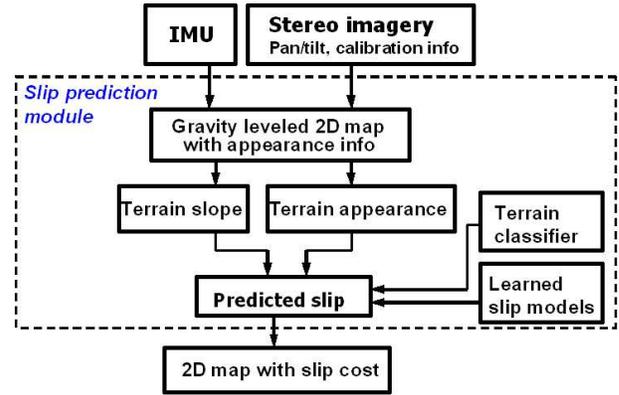


Figure 3. Slip Prediction Algorithm Framework

ule receives as input from the main module stereo pair imagery and rover attitude with respect to gravity. A map of the environment is built using the stereo range data registered with the color and texture information from the input images. In particular, each cell of the map contains information about terrain elevation and points to an image patch which has observed this cell. To predict slip in a map cell, the terrain classifier is applied to all the map cells in its neighborhood. A majority voting among their responses is used as the final terrain classification response at the desired rover location. Then, a locally linear fit in the cell’s neighborhood is performed to retrieve the local slope under the potential rover footprint. The slopes are decomposed into a longitudinal and a lateral slope with respect to the potential orientation of the rover. The two slope angles are used as inputs to a pre-learned nonlinear slip model for the particular terrain type determined by the terrain classification algorithm. The output of the module is the predicted slip for a given orientation of the rover and a slip related cost at a given location³. In our implementation, we provide the main module a mechanism to query slip at a desired map location.

During training, the rover collects appearance and geometry information about a particular location while it is observed by the rover from a distance. The corresponding slip of the rover is also measured when this location is being traversed. We use visual odometry (VO) between two consecutive steps to estimate the actual rover velocity. The commanded velocity of the rover is computed by using the full vehicle kinematics. The collected data pairs of visual information and slip measurements are given to the learning module which learns a terrain type classifier and independent slip models for each terrain type [1], [2]. More details of the two training components are given in the next two subsections. A rover position estimation is computed within the slip prediction module by accumulating the VO estimates. This is necessary to be able to map the current rover location to a location previously observed by the rover from a distance. The slip prediction mod-

³The slip related cost is a crude estimate of rover mobility without regards to particular robot orientation and is intended to be used in a D* path planning algorithm. It selects the maximum slip within a range of rover orientations.

ule can also use a Kalman filter position estimate based on merging multiple sensors, including VO, which can be provided by the main module [8].

Terrain Classification

Our current approach to terrain classification is based on processing visual appearance information, namely texture and color. We apply the texton based algorithm proposed in [22] which uses both color and texture simultaneously to learn to discriminate different terrain appearance patches. The algorithm proceeds as follows: initially the color R,G,B values in small pixel neighborhoods are collected and the most frequent features in the whole data are selected. Consequently, a histogram of the occurrence of any of the selected features within a patch corresponding to a map cell is built and compared by using a Nearest Neighbor classifier to a database of training patches [22]. Intuitively, a patch from a bedrock class will have a high frequency of pixels typical of previously observed bedrock patches, but it might also contain a small number of pixels which are typical of an unfamiliar to the system “rock” class which happens to be also shared with the soil and sand classes too (either of the terrain types patches might have small rocks dispersed in them). That is, this representation allows for building more complex appearance models and taking correct decisions given the observed statistics from the data.

Evaluation of the terrain classifier for this particular data domain has been provided in [2]. The terrain classification results are satisfactory and give initial successful slip prediction results [2]. The appearance-based terrain classifier can be improved by adding more sensors, both visual, e.g. multi-spectral imagery, or mechanical, e.g. vehicle vibrations. This is the topic of our current work.

Apart from being an instrumental part to the slip prediction module, the terrain classification provides important information to the HFTA algorithm (see Section 6). After the terrain type has been recognized, a canonical set of soil parameters associated with it are passed to the kinematic and dynamic simulation of the rover on the part of the terrain which has been deemed “uncertain”.

Learning the Slip Models

As each terrain type has a potentially different slip behavior [3], [24], we learn a slip model for each terrain independently. The slip models are built by learning a nonlinear approximation function which maps terrain slopes to the measured slip. The goal is to learn slip as a function of the terrain slopes: $S = S(x_{longit.}, x_{lateral})$. In our case we consider slip in the longitudinal direction only (parallel to the typical direction of travel), but the method can be trivially extended to learning of lateral slip or slip in yaw [1]. We have applied a Receptive Field Regression technique [23], but a standard nonlinear regression technique, such as a Neural Network, can also be used.

Implementation Details

The software architecture of the algorithm is designed to provide efficient slip prediction. Because terrain classification from visual information is generally time consuming, the focus has been on decreasing the amount of computation devoted to image processing related to terrain classification. In particular, our main idea is of evaluating the terrain type per map cell, rather than evaluating the terrain type in the whole image. This design concept can give significant advantages. Some speed-up can be achieved, as parts of the image do not belong to the map, e.g. the pixels above the horizon. Additionally, the terrain classifier will not be invoked if slip prediction is not needed in a certain area, e.g. an area which the main module has already marked as populated with obstacles or which is otherwise deemed uninteresting. Thirdly, a map cell at a close range covers a large part of the image compared to the ones at far ranges and can be processed selectively to speed up the processing without hurting the overall performance. More specifically, the map cell structure we use saves only its 3D location and pointers to images which have observed it (see Figure 4). When the terrain type needs to be predicted in a particular map cell, a projection of the map cell to the image is done and an image patch corresponding to this cell is retrieved.

Additionally, this paradigm allows for stereo imagery data to be received asynchronously or intermittently. In other words, one rover step can use multiple images, for example: when taking a panorama of the environment, if the rover is stalled and receives multiple identical images, if it does not receive imagery at all, etc. In any of these cases, the map is updated with new information, if such is available, and whenever a terrain classification is invoked, only the most recent terrain patch is used. The result of the terrain classification is saved with its corresponding confidence and might be combined with a potentially new evaluation if the confidence is insufficient. This is in contrast to processing fully all of the incoming images, extracting visual features and saving them to the map cells.

5. PATH PLANNING

Once the cost map is populated with information derived from terrain geometry and the slip prediction algorithm, as described above, an optimal path can be planned through this map from any start to any goal. We use a standard implementation of the D* algorithm [21] to plan this path. It is beyond the scope of this paper to go into the algorithmic details of this planner. In summary, it is a derivative of the well known A* search algorithm, with the capability to do efficient, incremental replanning.

6. HIGH-FIDELITY TRAVERSABILITY ANALYSIS

If any section of the path goes through “uncertain” regions of the triage map, then the HFTA algorithm is invoked onboard the rover. HFTA is a full kinematic and dynamic forward

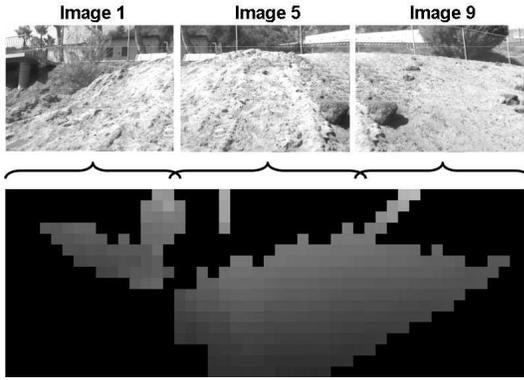


Figure 4. Schematic of the software design paradigm: each map cell keeps a pointer to an image which has observed it and the terrain classification is done only if needed. For example, the elevation map shown has been built from nine panoramic stereo image pairs, but effectively the visual information from only three images will need to be processed to fully classify the terrain.

simulation of the rover following a path. It is designed to find the lowest cost path through these “uncertain” regions.

The simulation infrastructure is provided by ROAMS [10]. ROAMS is a kinematic/dynamic simulation for rovers interacting with terrain. A detailed geometric and mass model is used to represent the rover in the simulation. This model includes all 15 degrees-of-freedom (DOFs) of the mobility system of the actual rover used in the experiments, Rocky8 (which is described in greater detail in Section 8). This includes the 12 active DOFs (six steering and six drive) and the 3 passive DOFs (rocker and two bogies). Each of the links connecting these DOFs has a mass and center of gravity (CG). Each of the active DOFs has a dynamic model representing a motor, which can be commanded in the same way as the actual motors on the rover.

The terrain is modeled geometrically using a mesh. For HFTA, this mesh is generated from stereo data, so it represents a realistic geometric model of the terrain around the rover each time this algorithm is used. As with any stereo data from a single point of view, obscurations will occur that cause “range shadowing.” This is a well-known effect and the terrain model will simply linearly interpolate over these shadows, essentially creating a ramp between the top of object creating the shadow and the terrain visible on the far side of the object. This ramp is actually the worst case scenario of the obscured terrain so it is a conservative assumption.

Mechanical soil properties can also be associated with the terrain. The terrain is again gridded, with the capability of assigning to each cell independent values for the cohesion, friction angle, and density of the soil in that cell. This enables the ability to represent non-homogeneous terrain at arbitrary resolution. For the HFTA algorithm, the terrain classifier (described in Section 4) predicts the terrain type of each of the

cells. Then, a canonical set of soil properties (derived experimentally or analytically) is associated with each of the terrain classes.

These mechanical soil properties are used in a dynamic wheel/soil interaction model used to determine the rover sinkage and slippage [20]. This model calculates and resolves the 18 forces (3 at each of the 6 wheels) of the vehicle interaction with the environment. This results in an accurate calculation of the net motion of the rover, including sinkage and slippage.

The combination of the stereo data generated terrain geometry and the terrain classifier generated soil properties creates a realistic model of the rover traversing a realistic model of the terrain.

For the forward simulation, the same slip-compensated path following algorithm that is used to control the actual rover [7], is used to control the simulated rover. So a path is passed to the simulation, the path-following algorithm follows the path over the sensor generated terrain that models the slippage of the simulated rover over this terrain. While this is being simulated, several metrics are being recorded that will enable the assessment of the traversability of that particular path.

The most important of these metrics is the energy required for the rover to traverse the given path. The first half of Equation 1 shows the energy calculation as the integration of the product of wheel torque and wheel velocity. Because the wheel terrain interaction (and thus vehicle slip) is being modeled, and because the slip is being compensated for, this metric penalizes for both high slip terrain (because the wheels must turn for a longer period of time to reach the goal) and for rough terrain (because the wheel torques are higher on locally steeper terrain, i.e. rocks, gullies, etc.). This metric also accounts for both forward and backward slip, where forward slip actually decreases the cost of traversal because the wheels turn for a shorter period of time to reach the goal. The path cost is calculated using:

$$C = W_e \left(\sum_{n=1}^6 \int T_n \cdot \omega_n dt \right) + W_p \left(\int (\| p_{des} - p_{act} \|) dt \right) \quad (1)$$

where T_n and ω_n are the torque and speed of wheel n , respectively; p_{des} and p_{act} are the desired and actual rover positions, respectively; and W_e and W_p are weighting terms.

Another metric that is used is path error (the second half of Equation 1). Because the slip-compensated path follower is being run in the simulation, path error is an indication of terrain that is more difficult to traverse, even when compensating for slip.

Other metrics that can be used are minimum ground clearance, minimum distance to extreme hazards, and maximum ground interaction forces. Ground clearance is the distance from the ground to the underbelly of the rover. This is a po-

tential hazard because it can “high center” the rover which can be a very serious threat to mobility. This type of forward simulation is the only way to accurately estimate the minimum ground clearance of a path. Distance to extreme hazards (such as large drop-offs, high-slip areas, wheel traps, etc.) is another metric to determine the risk of traversing a path and can be used to evaluate the relative cost of a path. Large ground interaction forces that can be caused by traversing over rough, stiff terrain (such as very rocky terrain) could also be used as a metric to increase the lifetime of the vehicle by selecting paths that place lower demands on the actuators and structure of the vehicle.

When the HFTA algorithm is invoked, the input is a single path through an “uncertain” region. The algorithm then randomly perturbs this single path to create multiple paths with the same start and finish points. It then runs the forward simulation of the rover through each of the paths, calculating a single scalar value of cost for each of the paths. It then outputs the path of the lowest cost.

7. SLIP-COMPENSATED PATH FOLLOWING

When the final path is created, the slip-compensated path following algorithm is invoked to enable the rover to actually follow this path regardless of the slip. This algorithm compares visual odometry (a technique that measures rover motion including slippage) and vehicle kinematics (a technique that measures rover motion minus slippage) to estimate the location and the slippage of the rover; it then compensates for this slippage and accurately follows the desired path to the goal.

This system is described in detail in [7], [9].

8. RESULTS

Slip Prediction Results

We have tested the slip prediction module independently on a LAGR⁴ vehicle. The dataset is collected outdoors in a natural park on five different terrains, including sand, soil, gravel, asphalt and woodchips. The slip models and the terrain classifier have been trained on 3000 frames. The summary results are presented in [2], and are the results of slip prediction over 2000 test frames, non-intersecting with the training data, while the rover traverses any of the above mentioned terrains. The average slip prediction error achieved is about 21%, which is a satisfactory result, given the amount of noise in the measured slip. Moreover, misclassification in the terrain type contributes to a large part of the error. In particular, if the terrain type were correctly classified, the prediction error decreases to about 11% on average. Terrain misclassification errors mainly occur between visually similar terrains such as sand and soil, especially for the parts of the soil areas which were covered with dust. See [2] for detailed results.

⁴LAGR stands for Learning Applied to Ground Robots and is an experimental all-terrain vehicle program funded by DARPA.

The final slip prediction is performed on the whole forthcoming map and a slip related cost is calculated. The cost based on the predicted slip is handed down to the terrain triage algorithm which further determines areas of the terrain which need more refined terrain traversability assessment and is finally given to a path planner. Figure 5 shows the map generated by the rover driving in deep sand and on upslope soil terrain. The corresponding terrain classification and slip prediction results are also shown for each map cell. As seen, it has been predicted that driving on flat sand incurs a large cost and the rover should prefer the neighboring woodchip terrain. Driving upslope on soil terrain causes about 40% slip which is mapped to a slip cost in the mid-ranges and which will further invoke the HFTA algorithm to determine the safest path. Also note that most terrain classification errors occur at ranges larger than 6m, where the image patch corresponding to a map cell is of very small size (for the LAGR rover configuration).

Integrated System Results

An integrated system including Navcam panorama capture, goal designation, goodness/triage map generation, path planning, and slip-compensated path following was performed onboard a research rover (Rocky8) in the Mars yard (see Figure 1). Rocky8 is a research platform that is used by many Mars Technology Program (MTP) tasks to demonstrate the software and algorithms on relevant hardware in a relevant environment (see [7] for a more detailed description of the rover). It is similar in design to the MER rovers and the future MSL rover.

For this demonstration the rover started at one side of a rock field on a sandy slope (see Figure 10). It first took a panoramic image consisting of ten stereo Navcam pairs spaced 10° apart, resulting in a 90° panorama. Then a user designated a goal from this panoramic imagery (see Figure 6). In this test the goal was designated at (12.0, -4.0) meters in the initial rover frame.

The point clouds are registered using the mast pan/tilt angles, and the rover roll and pitch (from the IMU) are used to gravity level the data. A goodness map and a triage map were generated from the stereo data (see Figures 7 and 8). These maps used 10 cm cell sizes. Then a path was planned through the goodness map using the D* algorithm.

Once the path was planned, it was passed to the slip-compensated path follower. The path follower compensated for slippage and successfully reached the goal. It was running Visual Odometry at 1 Hz and vehicle kinematics at 4 Hz, and resulted in a continuous motion from the start point to the goal. The results of the traverse can be seen in Figure 9. As can be seen, the rover traverses the path and arrives at the goal (see Figure 10). The actual path in this plot is the output of Visual Odometry. For a validation of the accuracy of Visual Odometry see [7],[16]. The maximum rover roll during this traverse was 16.3° and the maximum rover pitch was 7.2°.

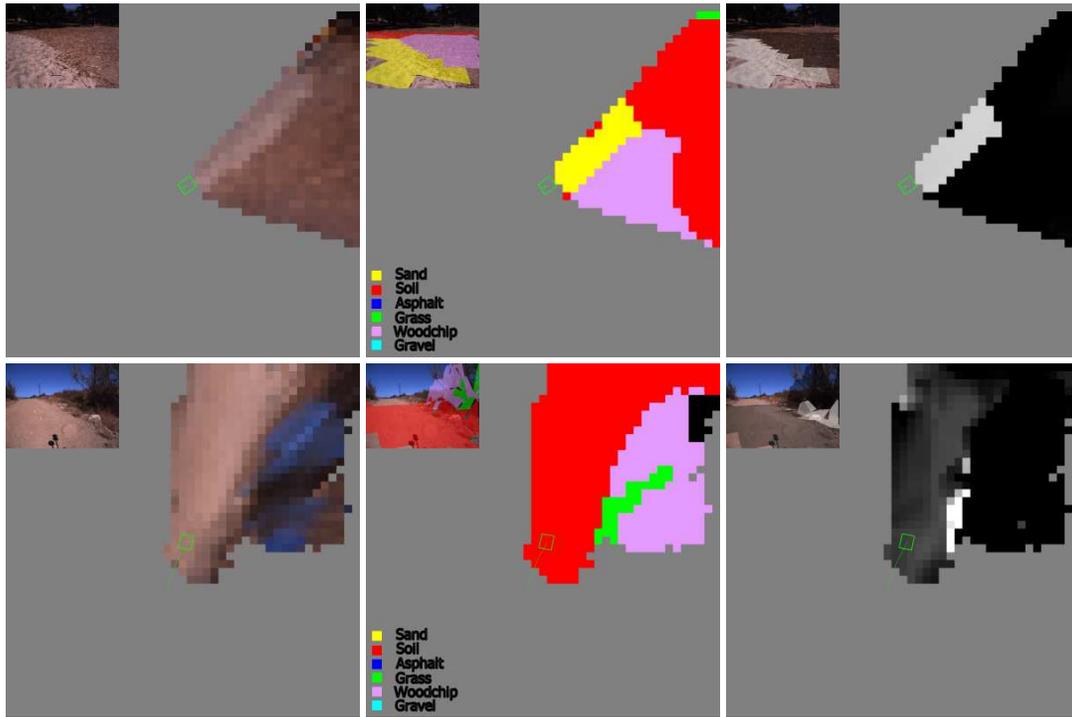


Figure 5. Example results of continuous slip prediction on the map as the rover drives (the path and the final rover position are marked in green). An input image and the corresponding map containing average color per cell (left), the automatic terrain classification (middle), the final slip cost based on the predicted slip (right) (brighter means larger cost). Top: the rover is traversing flat sandy terrain. Bottom: the rover is driving upslope on soil terrain.

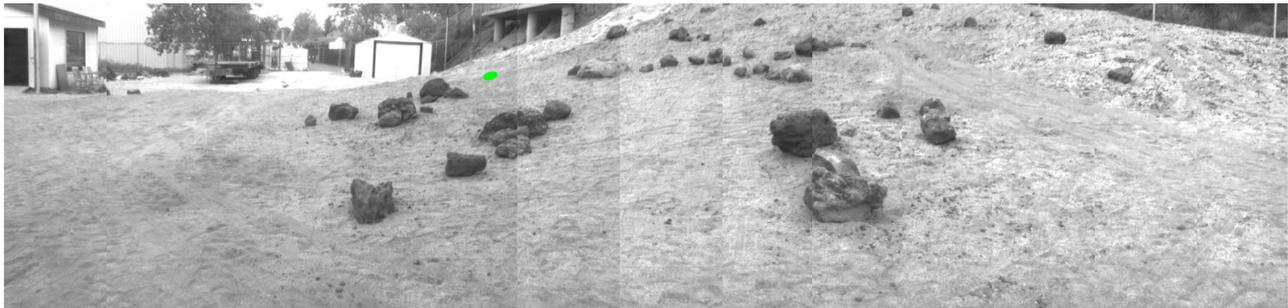


Figure 6. Panorama Taken by the Rover with the Designated Goal

HFTA Results

The HFTA algorithm was run on an analytical (i.e. not derived from stereo data) terrain consisting of a large hemisphere. It was run offboard, so it was not integrated with the rest of the system. The forward simulation of the rover was run four times over four different paths (see Figure 11). Table 2 shows the energy and path error metrics for each of the paths. It is important to note that the absolute accuracy of these metrics is not important because they are only used to compare relative path costs to determine the lowest cost path of the set. As can be seen, going directly over the hemisphere (path 1) is the most costly path in terms of energy and traveling on the side of the hemisphere (paths 2 and 3) is the most costly in terms of path error. Traveling on the flat ground around the hemisphere has low cost for both energy and path

error, and is clearly the most desirable path of the four. For this terrain, these results are supported by intuition. When the terrain complexity increases this algorithm will be able to distinguish subtle differences between paths that would not be possible using simpler methods. On average, this algorithm, takes approximately 4 seconds per meter of path analyzed. Decreasing this runtime will be a focus of future work.

Table 2. Path and Energy Costs for HFTA Simulation

Path	Energy (J)	Path Error (m-s)
1	5101.46	1.91
2	3324.51	12.32
3	1587.44	12.79
4	938.52	2.21

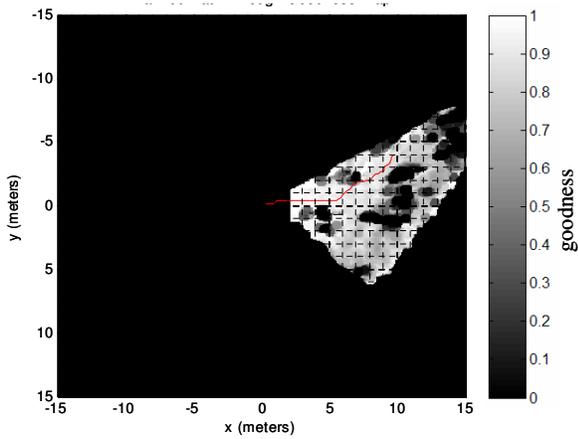


Figure 7. Goodness Map with Planned Path

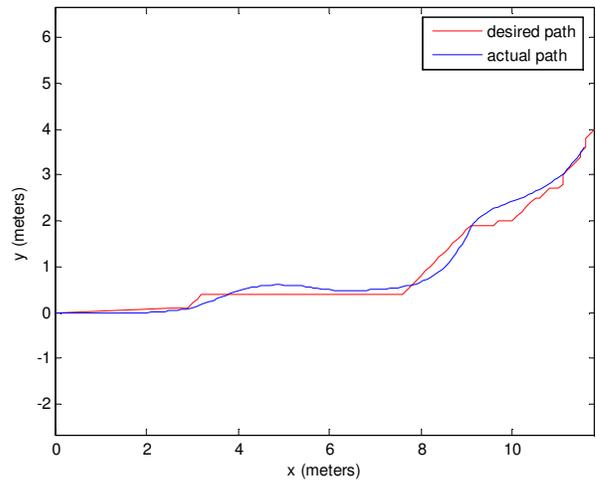


Figure 9. Slip-Compensated Path Follower Results

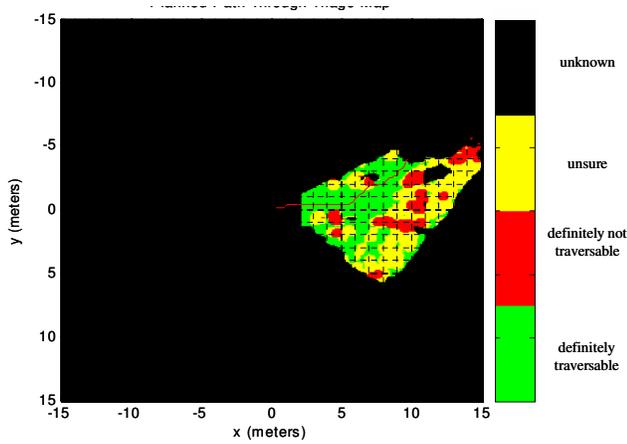


Figure 8. Triage Map with Planned Path

9. CONCLUSIONS

We show results from individual tests of various subsystems including: slip prediction and HFTA. Results from the slip prediction subsystem show that we are able to successfully predict slip at a distance using data gathered in the field from a rover. Results from the HFTA algorithm show that we can compare traversability of different paths by measuring metrics from a dynamic forward simulation of the rover.

We also show results from experiments of a subset of the described technologies integrated onboard a research rover in the Mars yard. These results demonstrate the feasibility of this approach as an end-to-end navigation system in a realistic Mars analog terrain.

10. FUTURE WORK

Future work will include the integration of the terrain classification, the slip prediction, and the HFTA algorithms into the onboard research rover software, and to demonstrate the entire integrated system in the Mars yard or in the field.



Figure 10. Rocky8 in the JPL Mars Yard at the Goal

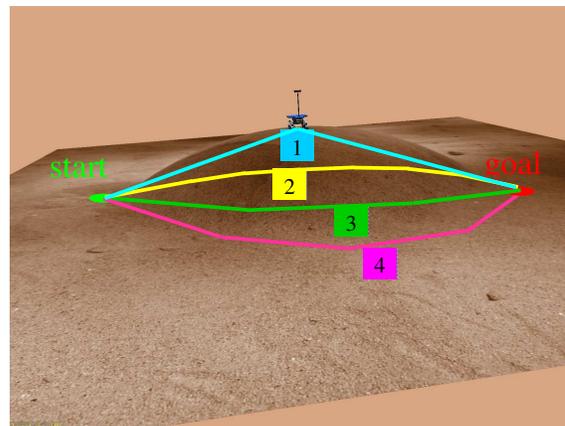


Figure 11. Paths Evaluated with HFTA

11. 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.

REFERENCES

- [1] A. Angelova, L. Matthies, D. Helmick, G. Sibley, P. Perona, "Learning to predict slip for ground robots," *International Conference on Robotics and Automation*, 2006
- [2] A. Angelova, L. Matthies, D. Helmick, P. Perona, "Slip prediction using visual information," *Robotics: Science and Systems Conference*, 2006
- [3] M. Bekker, *Introduction to Terrain-vehicle Systems*, Univ. of Michigan Press, 1969
- [4] J. Biesiadecki, E. Baumgartner, R. Bonitz, B. Cooper, F. Hartman, C. Leger, M. Maimone, S. Maxwell, A. Trebi-Ollennu, E. Tunstel, J. Wright, "Mars Exploration Rover Surface Operations: Driving Opportunity at Meridiani Planum," *IEEE Conference on Systems, Man, and Cybernetics*, 2005
- [5] A. Diaz, C. Elachi, et al., "A Roadmap for the Robotic and Human Exploration of Mars," May, 2005 (<http://science.hq.nasa.gov/strategy/road%5farch.html>).
- [6] S. Goldberg, M. Maimone, and L. Matthies, "Stereo Vision and Rover Navigation Software for Planetary Exploration," *IEEE Aerospace Conference*, Volume 5, Big Sky, Montana, March 2002.
- [7] D. M. Helmick, Y. Cheng, S. Roumeliotis, D. Clouse, L. Matthies, "Path following using visual odometry for a Mars rover in high-slip environments," *IEEE Aerospace Conference*, Big Sky, Montana, 2004
- [8] D. M. Helmick, S. Roumeliotis, Y. Cheng, D. Clouse, M. Bajracharya, L. Matthies, "Slip compensation for a Mars rover," *IEEE International Conference on Intelligent Robots and Systems*, 2005
- [9] D. M. Helmick, S. I. Roumeliotis, Y. Cheng, D. Clouse, M. Bajracharya, L. Matthies, "Slip-Compensated Path Following for Planetary Exploration Rovers," *Journal of Advanced Robotics*, October, 2006.
- [10] A. Jain, J. Balaram, J. Cameron, J. Guineau, C. Lim, M. Pomerantz, G. Sohl, "Recent Developments in the ROAMS Planetary Rover Simulation Environment", *IEEE 2004 Aerospace Conference*, Big Sky, Montana, March 6-13, 2004.
- [11] W. S. Kim, A. I. Ansar, and R. D. Steele, "Rover Mast Calibration, Exact Camera Pointing, and Camera Hand-off for Visual Target Tracking," *IEEE 12th Int. Conf. on Advanced Robotics (ICAR'05)*, July 2005.
- [12] C. Leger, A. Trebi-Ollennu, J. Wright, S. Maxwell, R. Bonitz, J. Biesiadecki, F. Hartman, B. Cooper, E. Baumgartner, M. Maimone, "Mars Exploration Rover Surface Operations: Driving Spirit at Gusev Crater", *IEEE Conference on Systems, Man, and Cybernetics*, 2005
- [13] J. N. Maki, J. F. Bell III, K. E. Herkenhoff, S. W. Squyres, A. Kiely, M. Klimesh, M. Schwochert, T. Litwin, R. Willson, A. Johnson, M. Maimone, E. Baumgartner, A. Collins, M. Wadsworth, S. T. Elliot, A. Dingizian, D. Brown, E. C. Hagerott, L. Scherr, R. Deen, D. Alexander, and J. Lorre, "The Mars Exploration Rover Engineering Cameras," *J. Geophys. Res.*, Vol 108, No. E12, 2003, 12-1 to 12-24.
- [14] M. C. Malin, J. F. Bell, J. Cameron, W. E. Dietrich, K. S. Edgett, B. Hallet, K. E. Herkenhoff, M. T. Lemmon, T. J. Parker, R. J. Sullivan, D. Y. Sumner, P. C. Thomas, E. E. Wohl, M. A. Ravine, M. A. Caplinger, and J. N. Maki, "The Mast Cameras and Mars Descent Imager (MARDI) for the 2009 Mars Science Laboratory," *36th Lunar and Planetary Science Conference*, 2005.
- [15] L. Matthies, S. Schafer, "Error modeling in stereo navigation," *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 3, June, 1987
- [16] M. W. Maimone and J. J. Biesiadecki, "The Mars Exploration Rover Surface Mobility Flight Software: Driving Ambition," *IEEE Aerospace Conference*, Big Sky, Montana, March 2006.
- [17] Clark F. Olson, Habib Abi-Rached, Ming Ye, Jonathan P. Hendrich, "Wide-Baseline Stereo Vision for Mars Rovers," *In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1302-1307, 2003.
- [18] Clark F. Olson and Habib Abi-Rached, "Wide-Baseline Stereo Experiments in Natural Terrain," *In Proceedings of the 12th International Conference on Advanced Robotics*, pages 376-383, July 2005.
- [19] G. Reina, L. Ojeda, A. Milella, J. Borenstein, "Wheel slippage and sinkage detection for planetary rovers," *IEEE/ASME Transactions on Mechatronics*, 2006
- [20] G. Sohl and A. Jain, "Wheel-Terrain Contact Modeling in the Roams Planetary Rover Simulation," *Fifth ASME International Conference on Multibody Systems, Nonlinear Dynamics and Control*, Long Beach, CA, September 2005.
- [21] A. Stentz, "The Focussed D* Algorithm for Real-Time Replanning", *Proceedings of IJCAI-95*, August 1995.
- [22] M. Varma, A. Zisserman, "Texture classification: Are filter banks necessary?," *Conf. on Computer Vision and Pattern Recognition*, 2003
- [23] S. Vijayakumar, A. D'Souza, S. Schaal, "Incremental online learning in high dimensions," *Neural Computation*, **12**, 2602-2634, 2005
- [24] J. Wong *Theory of Ground Vehicles*, John Wiley & Sons Inc., 1993



Daniel Helmick received his B.S. degree in Mechanical Engineering from Virginia Polytechnic Institute and State University and his M.S. in Mechanical Engineering with a specialization in controls from Georgia Institute of Technology in 1996 and 1999 respectively. Since June 1999 he has been working at

the Jet Propulsion Laboratory on robotics research projects involving vision/sensor based control of robots, state estimation, manipulation, and navigation and mobility algorithms. He has worked on robotic vehicles covering a wide range of functionality, including: Mars research rovers for rough terrain mobility; small, tracked robots for urban mobility; a cryobot for ice penetration; and reconfigurable wheeled robots for Mars exploration. His research interests include: sensor-based control of robots, sensor fusion and state estimation, and rover navigation and mobility.



Anelia Angelova has graduated with a Master of Science degree in Computer Science from the Department of Mathematics and Informatics, Sofia University, Bulgaria and a Master of Science degree in Computer Science from the Department of Computer Science, California Institute of Technology in 2000 and

2004 respectively. She worked as a software engineer for ADA Bulgaria from 1999 until 2001. She is currently a PhD student at California Institute of Technology. Her research interests are in the area of computer vision and machine learning.



Matthew Livianu graduated from Harvey Mudd College in 2004 with a B.S. in Computer Science learning AI and designing, building, and programming various mobile robots that found fire, followed walls, and navigated using GPS. He worked at Raytheon Company SAS performing signal integrity analysis for

space hardware. Currently, he is a Master's student at Georgia Tech, studying systems and controls and computer engineering. His research is focused on intelligent controls for autonomous mobile robotics.



Larry Matthies obtained a Ph.D. degree in Computer Science from Carnegie Mellon University in 1989, then moved to the Jet Propulsion Laboratory, where he is currently a Principal Member of Technical Staff and supervisor of the Machine Vision Group. His research has

focused on terrain sensing and obstacle avoidance algorithms for autonomous navigation of robotic vehicles. At JPL, he pioneered the development of real-time algorithms for stereo vision-based obstacle detection and he contributed to the development of the structured light sensor used by the Sojourner Mars rover. He has also developed algorithms for visual motion estimation from image sequences, 3-D scene reconstruction from image sequences, real-time terrain classification using multispectral imagers, and environmental mapping using sonar and stereo vision sensors. His group currently has research projects on computer vision for robotic vehicles sponsored by NASA, DARPA, and the U.S. Army; these projects include work on navigation of Mars rovers, asteroid and comet landers, and Earth-based robotic vehicles for urban and cross-country missions. He is a member of the editorial board of the Autonomous Robots journal and an adjunct member of the Computer Science Department at the University of Southern California. He has also been an invited speaker at the Frontiers of Engineering Symposium organized by the National Academy of Engineering.