

OASIS: Onboard Autonomous Science Investigation System for Opportunistic Rover Science

• • • • • • • • • • • • • • • •

• • • • • • • • • • • • • • • •

Rebecca Castano*

*Machine Learning and Instrument Autonomy
Jet Propulsion Laboratory
4800 Oak Grove Drive – MS 126-347
Pasadena, California 91109
e-mail: Rebecca.Castano@jpl.nasa.gov*

Tara Estlin

*Artificial Intelligence
Jet Propulsion Laboratory
4800 Oak Grove Drive – MS 126-347
Pasadena, California 91109
e-mail: Tara.Estlin@jpl.nasa.gov*

Robert C. Anderson

*Geophysics and Planetary Geosciences
Jet Propulsion Laboratory
4800 Oak Grove Drive – MS 264-621
Pasadena, California 91109
e-mail: Robert.C.Anderson@jpl.nasa.gov*

Daniel M. Gaines

*Artificial Intelligence
Jet Propulsion Laboratory
4800 Oak Grove Drive – MS 126-347
Pasadena, California 91109
e-mail: Daniel.M.Gaines@jpl.nasa.gov*

Andres Castano

*Machine Vision
Jet Propulsion Laboratory
4800 Oak Grove Drive – MS 125-209
Pasadena, California 91109
e-mail: Andres.Castano@jpl.nasa.gov*

Benjamin Bornstein

*Machine Learning and Instrument Autonomy
Jet Propulsion Laboratory
4800 Oak Grove Drive – MS 126-347
Pasadena, California 91109
e-mail: Benjamin.Bornstein@jpl.nasa.gov*

Caroline Chouinard

*Artificial Intelligence
Jet Propulsion Laboratory
4800 Oak Grove Drive – MS 125-209
Pasadena, California 91109
e-mail: Caroline.Chouinard@jpl.nasa.gov*

Michele Judd

*Science Division
Jet Propulsion Laboratory
4800 Oak Grove Drive – MS 183-335
Pasadena, California 91109
e-mail: Michele.Judd@jpl.nasa.gov*

Received 12 February 2006; accepted 12 February 2007

The Onboard Autonomous Science Investigation System has been developed to enable a rover to identify and react to serendipitous science opportunities. Using the FIDO rover in the Mars Yard at JPL, we have successfully demonstrated a fully autonomous opportunistic science system. The closed loop system tests included the rover acquiring image data, finding rocks in the image, analyzing rock properties and identifying rocks that merit further investigation. When the system on the rover alerts the rover to take additional measurements of interesting rocks, the planning and scheduling component determines if there are enough resources to meet this additional science data request. The rover is then instructed to either turn toward the rock, or to actually move closer to the rock to take an additional, close-up image. Prototype dust devil and cloud detection algorithms were delivered to an infusion task which refined the algorithms specifically for Mars Exploration Rovers (MER). These algorithms have been integrated into the MER flight software and were recently uploaded to the rovers on Mars. © 2007 Wiley Periodicals, Inc.

1. INTRODUCTION

The Mars Pathfinder and Mars Exploration Rover (MER) missions have demonstrated that mobile rovers are a viable and extremely useful option for exploring the surface of other planets. The MER rovers have traveled across kilometers of terrain and gath-

ered extensive scientific data, uncovering profound new insights into Mars' current and past environment, the history of its rocks, and the various roles and abundances of water. As a result of past successes, future missions are being planned that will send additional robotic explorers to Mars as well as to the moon and outer planets.

Surface rovers offer scientists the ability to move around a planetary surface and explore different areas of interest. The farther the rover can travel, the

*To whom correspondence should be addressed.

greater the opportunity exists for increased scientific discovery. Most mobile robot efforts at JPL and NASA have concentrated on navigation, manipulation, and control. For the MER mission, process automation has already proven valuable in engineering areas. For example, the rovers can drive autonomously using GESTALT (Maimone, Biesiadecki, Tunstel, Cheng, & Leger, 2006), a stereo-based hazard avoidance program that steers the rover away from rocks and steep hills. It can also keep track of its position using onboard visual odometry during some drives, which is more robust to slippage than wheel odometry (Cheng, Maimone, & Matthies, 2005). Due to advances in rover navigation, traverse ranges are increasing at a rate much faster than communications bandwidth. While the Sojourner rover traveled around 100 m in the entire mission, the drive record for the most distance covered in a single sol (Martian day) is over 220 m set by the MER Opportunity rover. As this trend in increased mobility continues, the quantity of data that can be returned to Earth per meter traversed is reduced. Thus, much of the terrain the rover observes on a long traverse may never be observed or examined by scientists. We present a system developed to maximize the quality of the science data transmitted to Earth through the use of onboard science. This system expands onboard automation beyond the engineering domain to the science domain.

The Onboard Autonomous Science Investigation System (OASIS) system has been developed to evaluate, and autonomously act upon, science data gathered by *in situ* spacecraft such as planetary landers and rovers (Castano et al., 2003; Castano et al., 2004; Castano et al., 2005; Castano et al., 2006). OASIS analyzes the geologic data gathered by the rover onboard. This analysis is used to identify terrain features of interest and additional science gathering opportunities. A planning and scheduling component of the system enables the rover to take advantage of the identified science opportunity by updating the command sequence to include the opportunistic measurements. OASIS currently works in a closed loop fashion with onboard control software (e.g., navigation and vision) and has the ability to autonomously perform the following sequence of steps: analyze gray scale images to find rocks, extract the properties of the rocks, identify rocks of interest, retask the rover to take additional imagery of the identified target and then allow the rover to continue on its original mission. We have conducted a number

of tests of the combined system and individual components. We describe results for the system in detecting and reacting to a science alert (identified science opportunity).

Several systems and components have been developed and demonstrated for autonomous rover science operations. Wagner et al. (2001) and Pedersen (2001) describe a system that was successful at autonomously identifying meteorites in Antarctica. Gullick, Morris, Ruzon, & Roush (2001) described techniques for analyzing field test data for the Marsrokhod rover. Gilmore et al. (2000) also presented several methods developed specifically for autonomous rover science. More recently, there has been a development of methods for autonomous science including classification of features and survey in association with the automated identification of life in the Atacama dessert (Smith, Niekum, Thompson, & Wettergreen, 2005; Thompson, Smith, & Wettergreen, 2005a; Thompson, Niekum, Smith, & Wettergreen, 2005b).

In this paper, we first discuss several motivating scenarios for on-board science and the OASIS system. We then explain the components of OASIS and the system itself after which we describe experimental testing with the system and assess the results. Finally, we address future work.

2. OVERVIEW OF THE OASIS SYSTEM

The OASIS system (Figure 1) consists of an analysis

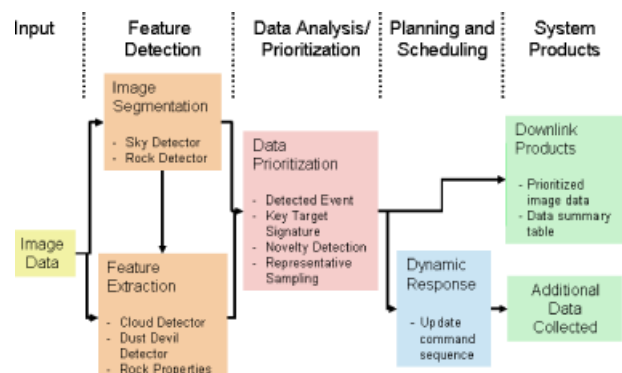


Figure 1. This diagram shows an overview of the OASIS system and how data flows between different system components.

capability for identification and prioritization of data and a planning and scheduling component to enable response to unanticipated opportunities. The analysis capability includes feature extraction as well as prioritization.

- **Feature detection:** Enables extraction of features of interest from collected images of the surrounding terrain. This module both locates rocks in the images and extracts rock properties (features) including shape, texture, and albedo. It also includes atmospheric analysis.
- **Data analysis and prioritization:** Analyzes the extracted features to assess the scientific value of the data and to generate new science goals.
- **Planning and scheduling:** Enables dynamic modification of the current rover command sequence (or plan) to accommodate new science requests from the data analysis and prioritization module. A continuous planning approach is used to iteratively adjust the plan as new goals arise. Opportunistic goals may be added to the plan as long as resource and other operational constraints are still met and all higher priority goals can be achieved.

2.1. Feature Detection

In this work, we will describe our methods for the analysis of scientific data acquired by a rover. Our techniques are applicable to a wide range of data modalities; however, our initial focus is on image analysis as images are commonly available and provide significant information about a scene.

The first step in image evaluation is the extraction of features of interest from the scene depicted. Features may be extracted from the original image or from a region after segmentation. Currently within OASIS there are two image segmentation modules and three feature extraction modules. Each of these modules applies general data analysis principals to identify and characterize image features that are representative of distinct scientific phenomena.

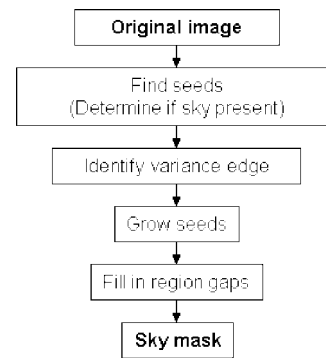


Figure 2. Flow diagram for sky detection algorithm.

2.1.1. Image Segmentation

2.1.1.1. Sky detector

Both to constrain geologic analysis to the ground and atmospheric analysis to the sky, it is beneficial to be able to distinguish the sky from the ground, i.e., to identify the horizon. Our approach to sky detection is based on region growing and is composed of four steps (Figure 2). First, we assess whether sky is present in the image or not. If so, seeds, areas of low variance, are identified. The lower bound of the sky is delineated by performing edge detection on the variance image. The seeds are then grown down to the variance edge. Finally, gaps in the sky region are filled. During this process we determine the skyline (the pixels at the interface of sky and ground) and the horizon (the image row below which there are no sky pixels). This algorithm had approximately a 90% accuracy in testing on 301 MER Opportunity rover images (errors were evenly distributed between false positives and misses).

2.1.1.2. Rock Detector

Detecting rocks in images is a valuable capability for autonomous planetary science. Rocks are excellent targets for compositional analysis with spectrometers. Their shape, size, and texture hold a wealth of geologic information. Computing the locations and distributions of rocks facilitates autonomous rover functions like adaptive target selection (Castano et al., 2006), selective image return (Castano et al., 2003), and autonomous site characterization (Thompson et al., 2005a).

Research in the past decade has produced a variety of strategies for detecting rocks. These include

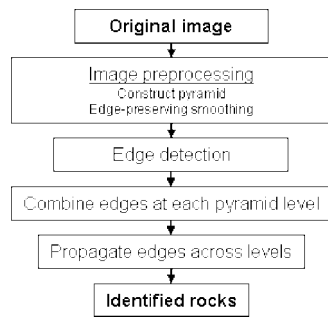


Figure 3. Flow diagram for rock finder algorithm.

stereo based techniques for finding rocks based on their protrusion from the ground plane, edge-based methods that find closed contours, template-based methods that look for characteristic pixel patterns, and methods that detect rocks using their shadows. Each approach has advantages for different conditions and mission requirements. For a formal comparison of the performance of these algorithms, see Thompson & Castano (2007).

From the many sensor modalities expected to be available on a rover, the rock detection algorithm initially used by the OASIS system is based on analysis of intensity of single grayscale images. The use of stereo is highly desirable for detection of large rocks but is unsuitable for detection of smaller rocks; further, it cannot be applied to images that do not come in stereo pairs, as is the case for the MER microscopic imager (MI). In contrast, rock detection on single grayscale images applies directly to analysis of low resolution hazard camera (hazcam), high resolution navigation camera (navcam), any spectral band (or combination) of the Panoramic camera (Pancam), and the MI camera. The detection of rocks is carried out by finding closed shapes in the image. The image is initially normalized, filtered with an edge preserving smoother (Tomasi & Manduchi, 1998) and its edges are enhanced using unmask sharpening. The edges of the resulting image are detected using both a Sobel and a Canny edge detectors (Trucco & Verri, 1998). For each result, we search for closed shapes (which presumably correspond to relatively small homogeneous regions) using an edgewalker. The results from both detectors are combined and output as a list of contours of the identified shapes. A flow diagram is shown in Figure 3. In testing on a set of 65 MER Spirit Pancam im-

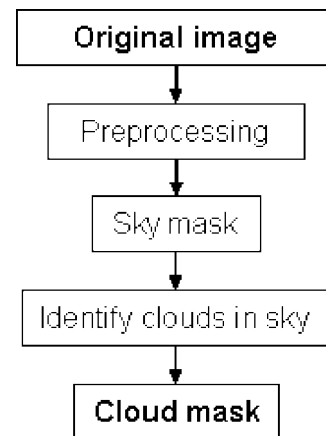


Figure 4. Flow diagram for cloud detection algorithm.

ages, 92% of the regions identified as rocks correctly were rocks. In a direct comparison to several other methods using MER Navcam images, this algorithm was at 90% while other rock identification algorithms were predominantly in the 65%–75% range. Testing with the FIDO field test rover in the JPL Mars Yard has yielded similar results. While having a very low false alarm rate, making it useful for automated target selection (see Section 3.3), it has a high miss rate (one of the lowest recall rates) and is best not used for estimating rock abundances.

2.1.2. Feature Extraction

2.1.2.1. Cloud Detector

Clouds are an atmospheric phenomena observed by the MER rovers on Mars. In detecting clouds, it is assumed that large variations in the intensity of the sky in the image correspond to clouds. Our approach to automating the detection of clouds is to first locate the sky (equivalently, the horizon) in an image and then determine if there are high variance regions in the sky (Figure 4). This algorithm, which operates on individual images (Figure 5), achieved over 93% accuracy in testing on 210 hand-labeled images taken by the Mars Exploration Rover Opportunity. There were three misses (false negatives) and eleven false positives. All of the three misses were labeled as a possible cloud (low confidence) by the scientist performing the labeling. No high confidence clouds were missed. For more details on the algorithms and experimental testing, see Castano

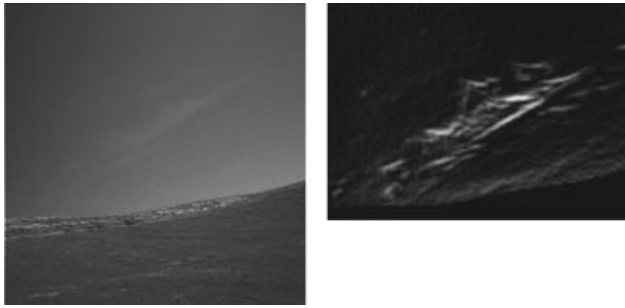


Figure 5. An example of cloud detection. Left image is the original image and right image is the result of the cloud detection algorithm.

et al. (2006) and Castano et al. (2007). Both the cloud detector and the dust devil detector described in the next section were uploaded on to the Mars Exploration Rovers. Preliminary experiments with the algorithms onboard have been successful.

2.1.2.2. Dust Devil Detector

A second type of dynamic atmospheric phenomena of interest on Mars is dust devils. The two most common methods for detecting dust devils are the comparison of two or more spectral bands of the scene and the motion detection using a temporal sequence. We selected the latter as it has application to grayscale as well as color imagery. In theory, detecting rapid motion in the scene is not equal to detecting dust devils. In practice, changes in a sequence of images taken over a short time period time at a scene on Mars are from dust devils. Dust devils are high dust opacity features on a dusty background and often have a faint signature in an image (Figure 6). The main challenge is to detect these often subtle features in the presence of significant image noise.

The algorithm (Figure 7) consists of a preprocessing step to reduce image noise followed by an image averaging. The difference between the average image and the input or test image is then computed. Noise effects are removed from the difference image and blob detection is performed on the remaining differences. A buffered bounding box is formed around each detection to ensure the full dust devil is captured. The dust devil algorithm was tested on 385 images, divided into 25 image sequences, acquired by the MER Spirit rover. The sequence lengths varied between 6 and 20 images. The algorithm achieved an 85% accuracy rate when the

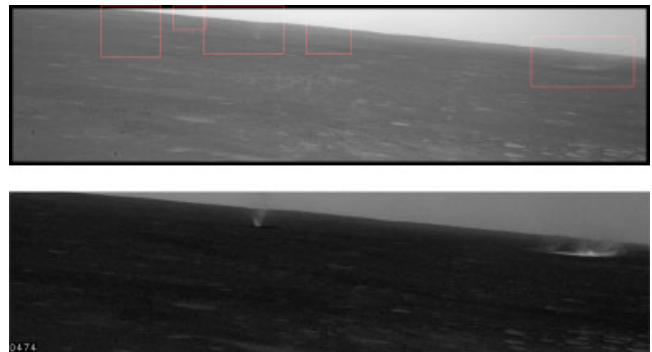


Figure 6. (Top) result of motion detection in an image. Two of the dust devils are observable (third and fifth box), while the other three occur later in the sequence. (Bottom) contrast adjusted image highlighting dust devils in scene. (enhanced image source: http://www.lpl.arizona.edu/~lemmon/mer_dd.html).

average image was determined using sets of four contiguous images. For more details on the algorithms and experimental testing, including results with different window sizes for the average image (i.e., different number of images used to form the average image), see Castano et al. (2006) and Castano et al. (2007).

2.1.2.3. Rock Properties

The primary focus of OASIS feature extraction and system testing has been on geological features, spe-

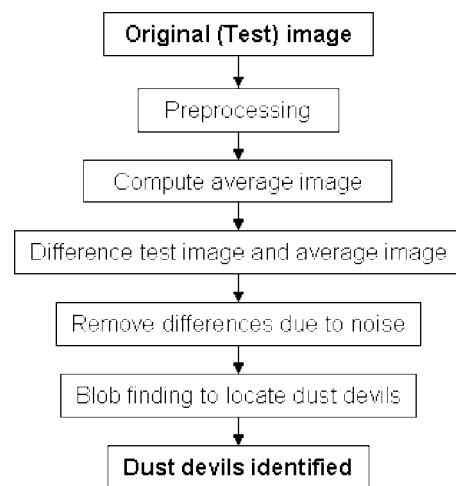


Figure 7. Flow diagram for dust devil detection algorithm.

cifically rock properties, rather than atmospheric phenomena. In terrestrial studies, the physical properties of rocks and soils are analyzed for information regarding their geologic history. Physical properties such as rock/grain size, shape, composition, sorting, and texture are common features of rocks and soils that can be used to infer their origin and transportation history. Using image data, the properties that OASIS currently estimates are albedo, texture, size, and shape.

The albedo of a rock is an indicator of the reflectance properties of a surface. OASIS measures albedo by computing the average gray-scale value of the pixels that comprise the image of the rock. The reflectance properties of a rock provide information about its mineralogical composition. Shadows and sun angle can both affect the gray-scale value of a pixel. Although this can be corrected by using the range data along with knowledge of both the sun angle and the camera orientation, the current system does not address these specific issues.

OASIS uses Gabor filters to estimate the visual texture of observed rocks (Castano, Mann, & Mjølness, 1999). Visual texture can provide valuable clues to both the mineral composition and geological history of a rock as well as soil (Mahmood, Mitchell, & Carrier, 1974).

One of the important properties of rocks on the surface is their size. Size can be used to identify sorting and geologic contacts. We model rocks as ellipses (if no range data is provided) or ellipsoids (if range data is available).

Inherent shape is another important and geologically informative feature of rocks. Although the shape of a rock is complex and often difficult to describe, significant geologic information can be extracted to better understand provenance (source of material) and environmental conditions. Various shape parameters are used to classify rocks in terrestrial studies (Wadell, 1932), including elongation (or aspect ratio), ruggedness (or angularity), and surface area. In OASIS, an ellipse is fit to the outline of the rock (Halir & Flusser, 1998). The eccentricity of the fit ellipse, as well as the error, is computed (Fox, Castano, & Anderson, 2002). In addition, the angularity of each rock is assessed using a measure of ruggedness (Hentschel & Page, 2002).

2.1.2.4. Boundary Detection

A rover on a planetary surface, such as Mars, represents a modern version of a field geologist on Earth.

One of the most important tasks for a geologist is to develop an understanding (spatial and temporal) of the field area. For most mapping projects, this involves going into the field area, identifying what types of rocks are present, identifying key landforms and landscapes, mapping geological contacts or boundaries, developing a geologic map, and creating a model based on historical interpretations and the dynamic processes that have shaped the landscape.

In the field, rocks exposed at the surface are the only record of the surface history; their physical appearance and location testify to their environmental settings in which they formed. Therefore, to gain an understanding of the basic geologic history of a region, one must be able to identify where the rocks on the surface originated from. To do this, one must identify and map geological contacts/boundaries in the field. The identification and mapping of geological boundaries can range from simplistic boundary detection, such as hills, plains, and river channels, to complex identification of different rock and clast types, to erosional and depositional histories of the landscape. For the field geologists, detail examination of units provides the data for mapping; for the rover, it is the critical hardware interface in conjunction with the scientist back on Earth that collects the data. For future long-range rovers on a planetary surface, it is critical for the rover's on-board software to be capable of identifying simple boundaries transitions during long traverses.

2.2. Data Analysis/Prioritization

After extraction of features, the information is used to effect a change either in the downlink data queue or in the rover activities. In this section, we describe four methods used to assess the priority of the data and the information it contains. The first method is used for clouds and dust devils while the remaining methods are currently used with rocks in OASIS.

2.2.1. Detected Event

The first type of reaction is to set a flag when an event of interest is detected. In the case of dust devils and clouds, the detection of an event in an image is used to flag that image or image sequence for downlink. Cloud detection images without clouds need not take up downlink bandwidth. Similarly, dust devil image sequences that do not contain dust

devils do not need to be transmitted to Earth. Other potential reactions include collecting an image sequence of the cloud or collecting additional types of data when a dust devil is detected.

2.2.2. Key Target Signature

The second technique recognizes pre-specified target signatures that have been identified by the science team as data of high interest. This prioritization method enables scientists to efficiently and easily stipulate the value and importance to assign to each feature. In OASIS, this method is used to prioritize rocks as a function of the distance of their extracted feature vector from the specified weighted feature vector. Scientists are given two ways to set the target signatures that will determine how the rocks are ranked. In the first method, the scientist can directly set the importance of specific feature values. For example, the scientist may choose to prioritize rocks based on two aspects of their shape, such as eccentricity and ellipse fit. The second manner in which scientists can specify a target signature is by selecting a rock with interesting properties from the set of already identified rocks. Rocks that resemble this particular rock in the selected properties are given a high priority. While we have emphasized rocks, one could envision using this method with other features, including identification of clouds or dust devils with particular properties.

2.2.3. Novelty Detection

The third technique, novelty detection, identifies unusual signatures that do not conform to the statistical norm for the region. We have developed three methods for detecting and prioritizing novel signatures, representing the three dominant flavors of machine learning approaches to novelty detection:

- distance-based,
- probability-based (i.e., “generative”),
- and discriminative.

The first novelty detection method is a distance-based k -means clustering approach. Again, we have applied this to rock data. Initially, all available rock data is clustered into a specified number (k) of classes. The novelty of any rock is then the distance of the rock feature vector to the nearest center of any

of the k clusters. The greater the rock’s distance is to the nearest center, the higher the novelty ranking assigned to the rock.

The second technique is a probability-based Gaussian mixture model, which attempts to model the probability density over the feature space. In this approach, the novelty of a rock is inversely proportional to the resulting probability of that rock being generated by the model learned on previous rock data.

The final method is a discrimination-based kernel one-class classifier approach. Here we treat all previous rock data as the “positive class” and learn the discriminant boundary that encloses all that data in the feature space. We essentially consider the previous rock data as a cloud scatter in some D -dimensional space, where D is the number of features. The algorithm learns the boundary of that cloud, so that future rock data that falls farther outside the cloud boundary is considered more novel.

2.2.4. Representative Sampling

The last prioritization algorithm, known as representative sampling, prioritizes data for downlink by ensuring that representative rocks of the traversed region are returned. One of the objectives for rover traverse science is to gain an understanding of the region being traversed. To meet this objective, the downlink back to Earth should include information on rocks that are typical for a region, and not just information on interesting and unusual rocks. A region is likely populated by several types of rocks with each rock type having a different abundance. If uniform sampling is employed for downlink image selection, as opposed to our autonomous on-board selection process, the downlinked set will be biased towards the dominant class of rock present. This situation may result in smaller classes not being represented at all in the downlinked data.

To provide an understanding of the typical characteristics of a region, rocks are first clustered into groups with similar properties. The data is then prioritized to ensure that representative rocks from each class are sampled. The rocks are clustered into groups based on the features extracted from the image data for each rock. To determine the classes, the property values are concatenated together to form a feature vector, and a weight is assigned to the importance of each property. Different weight assignments can be used as a function of the particular

properties that are of interest. For example, albedo and texture are typically used to distinguish types of rocks, but rock size may be used if sorting is of interest. Unsupervised clustering is then used to separate the feature vectors into similar classes. We currently employ k-means due to its relatively low computational requirements, although any unsupervised method could be used. For each class of rocks, we find the most representative rock in the class, i.e., the single rock in any image that is closest to the mean of the set. We give a high priority to the image containing this rock. The optimal number of classes can be determined using cross-validation techniques (Smyth, 1996).

2.3. Planning and Execution

When the data analysis software identifies science targets of interest (e.g., a novel rock), a science alert is generated. This results in a new science goal being passed to the planning and scheduling module which determines if the new goal can be accommodated. If it can be, the current rover command sequence is modified to collect new science data.

The OASIS planning and execution module (Estlin et al., 2007) is intended to run with little communication with ground. It accepts new science goals and then modifies the current rover command sequence (or plan) to try and achieve as many of the goals as possible while still respecting relevant state and resource constraints. This module also executes the current rover plan by dispatching commands to the rover's low-level control software and monitoring relevant state and resource information to identify potential problems or opportunities. If problems or new opportunities are detected, the system is designed to handle such situations by using replanning techniques to add, move, or delete plan activities.

2.3.1. Challenges for Onboard Planning and Execution

Rovers equipped with on-board planning and execution capabilities have the potential for increasing science return not only by dynamically handling opportunistic science events but also by making sequencing decisions on-board with access to current state and environment information. There are, however, a number of challenges in providing software to support these types of autonomous operations.

To generate and/or modify its own command

sequence for carrying out a set of science goals, the onboard software must reason about a rich model of science and engineering activities and the corresponding resource and temporal constraints. For example, the software has to predict power consumption of variable duration activities such as downlinks and traverses, keep track of available power levels, and ensure that generated plans do not exceed power limitations. Science activities may also have varying priorities that indicate their overall mission value. On-board planning and execution software must reason about these priorities and handle new science opportunities in a dynamic and efficient manner.

Sequence generation for rover surface missions also raises a number of interesting challenges regarding spatial reasoning capabilities. One of the dominating characteristics of rover operations is traverses to designated waypoints and science targets. This element is especially important in future missions that intend to explore large geographic areas. On-board planning and execution software needs to coordinate with several levels of rover navigation software to generate an efficient and achievable rover plan. In addition, it is difficult to predict duration and resource usage of rover navigation operations since rovers are traveling over unknown terrain. A large part of a rover schedule consists of rover moving to different target locations. If time and resource predictions for these moves are inaccurate, the onboard software must be able to continuously modify the schedule in order to accommodate the new information.

Further, on-board autonomy software must robustly handle the inherent uncertainty in rover pose estimation. The estimation of rover position is often a constant source of error. The Sojourner Rover produced a position error of roughly 5%–10% of distance traveled and an average heading drift of 13° per day of traverse (Mishkin et al., 1998). The MER rovers have seen large variations in position error, including Opportunity experiencing a slippage of close to 100% when the rover was caught in a sand dune (but thought it drove 40 m). Though visual odometry software has helped the MER rovers perform more accurate pose estimation than wheel odometry alone, it is typically only run in a limited fashion due to the additional time requirements.

2.3.2. Planning and Execution System Description

Planning and execution capabilities in OASIS are provided by an integration of the CASPER (Continuous Activity Scheduling, Planning, Execution and Re-planning) continuous planner and the TDL (Task Description Language) executive system (Estlin et al., 2007).

In our system framework, CASPER and TDL handle the following functionality:

- Creating an initial plan based on an input set of goals,
- maintaining resource, temporal and other rover operability constraints,
- executing a plan by interacting with basic and low-level rover control functionality (e.g., navigation, vision),
- monitoring plan execution to track plan activity and goal status,
- dynamically modifying the current plan based on plan activity, state, and resource updates,
- performing plan optimization to reason about soft constraints and goal priorities,
- handling dynamically identified science goals (called *science alerts*) that are generated through onboard data analysis.

Planning and scheduling capabilities for OASIS are provided by the CASPER continuous planning system (Chien, Knight, Stechert, Sherwood, & Rabideau, 2000). Based on an input set of science goals and the rover's current state, CASPER generates a sequence of activities that satisfies the goals while obeying relevant resource, state and temporal constraints, as well as operation (or flight) rules. Plans are produced using an *iterative repair* algorithm that classifies plan conflicts and resolves them individually by performing one or more plan modifications.¹ CASPER also monitors current rover state and the execution status of plan activities. As this information is acquired, CASPER updates future-plan projections. Based on this new information, new conflicts and/or opportunities may arise, requiring the planner to re-plan in order to accommodate the unexpected events. An example of a plan in the

¹Definitions of plan activities, goals, resources, and states, as well as the aforementioned constraints and rules, are encoded in a CASPER plan model.

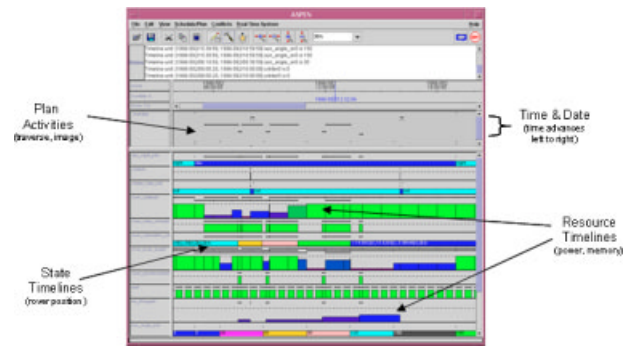


Figure 8. Sample rover plan displayed in planner GUI. Plan activities are shown in upper portion of window, where bars represent the start and end time of each activity. State and resource timelines are shown in bottom portion of the screen and show the effects of the plan as time progress. Time is depicted as advancing from left to right.

CASPER GUI that was executed during a rover demonstration is shown in Figure 8.

The executive functionality in OASIS is performed by the TDL executive system (Simmons & Apfelbaum, 1998). TDL was designed to perform task-level control for a robotic system and to mediate between a planning system and low-level robot control software. It expands abstract tasks into low-level commands, executes the commands, and monitors their execution. TDL also provides direct support for exception handling and fine-grained synchronization of subtasks. TDL is implemented as an extension of C++ that simplifies the development of robot control programs by including explicit syntactic support for task-level control capabilities. It uses a construct called a *task tree* to describe the tree structure that is produced when tasks are broken down into low-level commands.

Currently, OASIS has a separate planner and executive and, thus, this framework does share similarities to other three-layer architecture approaches (Gat, 1991; Bonasso et al., 1997; Alami, Chatila, Fleury, Ghallab, & Ingrand, 1998). However, as compared to these approaches where planning is typically done in a batch fashion and takes on the order of minutes to hours, this integration uses a continuous planning approach, where plans are updated and repaired in a matter of seconds. This enables the use of planning techniques at a finer timescale for tracking the progress of plan execution, quickly identifying potential problems in future parts of the

plan, and responding accordingly. As we expect minor portions of the plan to change frequently, we use a lightweight plan runner to dispatch activities to the executive a few seconds before the task's scheduled *start time*. This approach differs from the more common batch approach of turning the entire plan over to the executive for execution. Executive techniques are then used in only reactive situations or at times where procedural reasoning is preferred.

Another related approach directed towards rover command generation uses a contingency planner (CPS) to schedule rover operations where the generated plan allows both temporal flexibility and contains contingency branches at points where the rover activity is predicted to have above a certain probability of failure. (Bresina, Golden, Smith, & Washington, 1999). Contingent command sequences (or plans) are produced on the ground and only the executive, which executes the sequence, is on-board the rover. This approach enables planning to be performed without the strict processing constraints that are often present when running on-board a spacecraft. However, typically, only a limited number of contingencies can be preplanned due to the size and complexity issues. Further, it may be difficult to predict probable points of plan failure, especially when rovers drive through unknown terrain. The OASIS system approach provides more flexibility to handle new situations since the planner is on-board and can respond to a much large number of problem (or fortuitous) situations. Further, other work has shown the feasibility of using an automated planner on-board a spacecraft, even when only very limited processing and memory are available (Chien et al., 2005).

2.3.2.1. Science Alerts

To handle opportunistic science, we enabled the OASIS planning and execution module to recognize and respond to science alerts, which are new science opportunities detected by on-board data-analysis software. For example, if a rock is detected in navigation imagery that has a previously unseen shape or texture, a science alert may be generated to take additional measurements of that rock.

Science alerts can have different levels of reaction from the planning and execution system. The most basic reaction is to adjust the rover plan so that the rover holds at the current position and the flagged data is sent back to Earth for further analysis

at the next communication opportunity. The next level of reaction is to collect additional data at the current site before transmitting back to Earth. A further step is to have the rover alter its path to get closer to objects of interest before taking additional measurements. These operations provide new data that could not be obtained through analysis of the original image. Each of these levels of reaction has been demonstrated for the OASIS system in simulation and with rover hardware (more details are given in Section 4).

How the plan is modified to accommodate the alert depends on the type of alert that is considered. When a science alert is received that requires holding at the current position until data is communicated with Earth (called a *stop and call home* alert), the system alters the plan to remove or abort any activities that are not critical and wait for the next communication opportunity. If activities are currently executing, the planner requests the executive components of OASIS to abort them. If activities are scheduled in the future, the planner deletes them and resolves any inconsistencies created by these deletions.

To handle a science alert that requests additional measurements (called a *data sample request* alert), the planner must generate a plan that achieves the new goals without deleting existing activities or causing conflicts that cannot be resolved (e.g., scheduling more activities than can be executive over a certain time window). Data sample requests are also assumed to have an assigned priority, which is representative of their scientific value. This priority is currently assigned by the onboard data analysis system. A simple linear priority scheme is used to reflect a request's value and is chosen by the data prioritization algorithm being employed. For example, if the analysis algorithm is looking for rocks with a certain set of properties, then rocks with features that closely match the target property set will be assigned high priorities.

To handle a data sample request, the planner must be able to add a new science observation and a new move command to correctly place the rover in position to take the observation. This process is done through the use of planning optimization capabilities. Because it may not be possible to accommodate



Figure 9. Rocky 8 rover (left), FIDO rover (middle), Rocky 7 rover (right).

all alerts,² a science alert is represented as an optional planning goal with a particular priority. As an optional goal, its achievement is not mandatory but may improve the plan's optimization score if included in the plan. Before attempting to handle a science alert, CASPER protects the current plan by saving a copy before optimization. If CASPER can handle a new science alert (e.g., by adding additional science measurements) without causing other negative affects, such as resource over-subscriptions or the deletion of ground-specified (mandatory) science goals, then the new plans optimization score is evaluated. If the new plan has an optimization score that is higher than the original plan, then the new plan that accommodates the science alert is used. For more information on how CASPER reasons about plan optimization please see (Rabideau, Engelhardt, & Chien, 2000). CASPER also is given a time limit to search for a new plan that handles the science alert. If a new plan is not found under the time limit, the previous plan is restored and the science alert remains unsatisfied. This time limit can be adjusted based on user preferences and available processing time.

2.3.3. CLARATy Robotic Architecture

In order to test OASIS with rover hardware, the planning, scheduling, and execution component is integrated with the Coupled Layered Architecture for Robotic Autonomy (CLARATy) (Nesnas et al., 2003), which is being developed at JPL in response to

²In system testing, a number of different situations have been evaluated including allowing the data analysis component to generate a large set of data sample requests that cannot all be accommodated due to plan limitations on resources and time.

the need for a robotic control architecture that can support future mission autonomy requirements. CLARATy is a unified and reusable robotic software architecture that provides a large range of basic robotic functionality and simplifies the integration of new technologies on different robotic platforms. Through CLARATy, various OASIS components have been tested with several hardware platforms including Rocky 7, Rocky 8, and FIDO (shown in Figure 9). For the testing reported in this paper, the FIDO rover was primarily used and the OASIS rock-finder and planning and execution software were directly integrated with CLARATy and handled interaction with other key CLARATy elements that were required to run with rover hardware. Other pieces of OASIS (such as feature extraction) are in the process of being integrated but were not integrated for these tests. Through this integration, the OASIS system interacted with several technologies provided by CLARATy, which operate onboard the rover. These software components are locomotion, position estimation, navigation, and stereo vision processing.

The locomotion module coordinates a set of motors to steer and drive the rover's wheels, enabling the rover to move across the terrain. The FIDO rover, which was used in these tests, is a six-wheeled vehicle with all-wheel steering and driving enabling the rover to turn in place, drive in straight lines, drive in arcs, and perform crab maneuvers in which the rover moves in one direction while facing another.

As the rover traverses across the landscape, position estimation software attempts to estimate the rover's current location and orientation relative to a global and a local reference frame. Varying texture in the sand results in different amounts of wheel sink-

age which, in turn, results in variance in the effective wheel radius of the rover. In addition, the rover often encounters rocks of varying sizes, shapes, and textures which results in wheel slippage as it drives over the rocks. The position estimation algorithm in the current system uses an IMU (Inertial Measuring Unit) to estimate rover attitude (roll, pitch, and heading) and wheel odometry to estimate linear velocity. The IMU helps detect slippage that could not be detected with wheel odometry alone.

Navigation software is used to guide the rover to a goal location while avoiding obstacles. OASIS uses the Morphin navigation system developed at CMU (Urmson et al., 2003). As part of the hazard avoidance algorithm, the navigator acquires stereo images from the rover hazard cameras to detect obstacles in the rover's path. The front camera images are also passed to the OASIS rock detection process, running in a separate thread.

Both the navigator and the OASIS rock detection algorithm make use of the JPL stereo vision package to acquire range data for the hazard camera images. The navigator uses stereo processing to generate a point cloud representing the terrain around the rover. OASIS uses range data to identify the locations of identified rocks. Rocks are first located in a coordinate frame relative to the center of the rover and then translated into the global reference frame using the current position estimate.

3. ONBOARD SCIENCE FOR MAXIMIZING MISSION SCIENCE RETURN

On-board science or science autonomy refers to the capability of analyzing the scientific content of data on-board a spacecraft and using this information to autonomously take intelligent actions. The capability can be used to increase the science return of a mission by ensuring that high science content data reaches scientists on Earth. With the constrained computing resources, time, power, and downlink bandwidth on planetary rovers, onboard science can be used for efficient allocation of a rover's time and bandwidth resources. The OASIS system for on-board science can achieve this through opportunistic science during a traverse, campaign science, end-of-day science, and prioritization of data for downlink.

3.1. Opportunistic Traverse Science

One mission concept for future rover exploration is to focus the scientific investigation on multiple science sites that are located several rover traverse days away from each other. This scenario allows for a deep and concentrated exploration of each science site; however, in order to expeditiously drive the rover to each of the sites, scientists cannot perform detailed examinations of the terrain between each site. This is where traverse science, the capability of autonomously studying the terrain during the long traverse, can be beneficial.

OASIS can be used to identify and react to science opportunities along a traverse. As explained in the previous sections, the system can identify features that scientists have specified as important as well as novel features (see, for example, Figure 10) and then, if resources are available, autonomously take additional measurements of that feature.

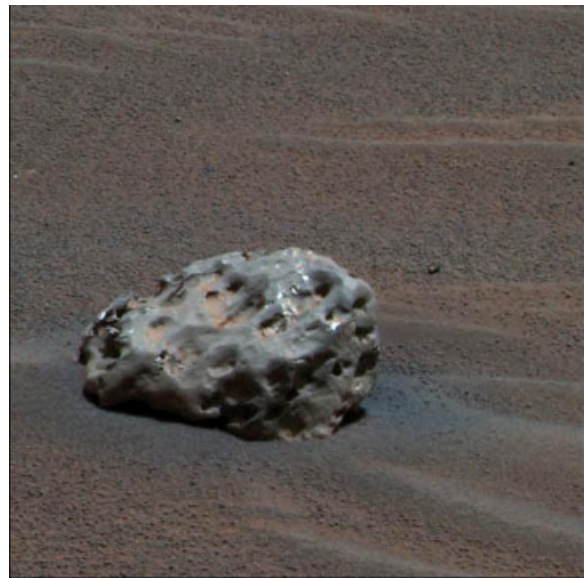


Figure 10. Image credit NASA/JPL/Cornell, JPL archive number PIA07269. On January 2, 2005, NASA's Mars Exploration Rover *Opportunity* found this iron meteorite on the surface of Mars—close to the heat shield that the spacecraft dropped on its descent to the surface. OASIS can train the rover to look for meteorites, or other types of rock, using the key target signature data analysis technique.

3.2. Campaign Science

OASIS can be used not only during a traverse, but also for campaigns to monitor for science events or features of interest. As previously described, prototype algorithms for dust-devil detection and cloud detection that were developed for OASIS have been refined for MER and integrated into the MER flight software. These algorithms allow selective transmission of data containing the features of interest. Ground testing indicates that a 4X increase in the number of images with science features can be expected through the use of these algorithms.

3.3. Automated Target Selection/End-of-Day Science

A third benefit of onboard science data analysis is the identification of targets both at the end of the day and during the course of a drive. A number of rover remote sensing instruments have a very narrow field-of-view and, thus, require selection of specific focused targets for sampling. Such instruments include mini-TES (thermal emission spectrometer), LIBS (laser induced breakdown spectrometer), and infrared point spectrometers. The typical scenario for selecting targets for these instruments is to manually identify the targets using data that has been previously downloaded. This means that targets can only be selected from the site for which data has already been downloaded the night before (thus, at the end of the day, the rover sits and waits until the next morning for instructions on which rocks to sample).

Rather than waiting until the next day, some samples could be collected by “blindly” targeting the instrument after a traverse day. However, by analyzing image data on-board, targets for these instruments can be identified automatically. As part of the OASIS system, we have implemented a method for automatically selecting rock targets for sampling at the end of a traverse. This could be used, for example, on the Mars Science Laboratory (MSL) to select targets for the ChemCam instrument (which includes a LIBS) to sample.

The approach is to first identify the rocks in the scene using the rock finder in the feature extraction component. Points on these identified rocks are then selected for targeting (Figure 11).

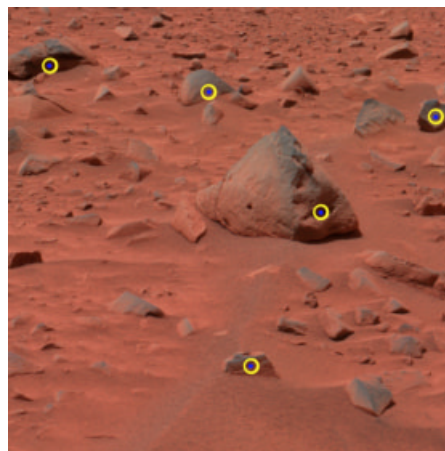


Figure 11. OASIS selects five potential targets for the ChemCam instrument to sample. Autonomously selecting targets vs blind sampling should greatly increase the chances of accurately targeting a rock.

3.4. Onboard Prioritization of Data

Finally, one of the primary motivations for the development of OASIS was to prioritize data for downlink. In this application, OASIS analyzes data the rover gathers, and then prioritizes the data based on criteria set by the science team. Three prioritization methods have been developed:

- *Representative sampling* (this method ensures that at least one image of every rock type encountered that sol is returned to Earth),
- *Key target signature* (the rover looks for science-specified rock types and gives images with those rocks a high priority for downlink), and
- *Novelty detection* (if the rover sees a rock type that it has not seen before, it assigns that rock’s image a high downlink priority).

At the next opportunity for transmitting data back to Earth, the data is already prioritized—ensuring that the most valuable data is sent first.

4. OASIS SYSTEM TESTING

To evaluate our system, we have performed a series of tests over several years both in simulation and using rover hardware in the JPL Mars Yard. These tests

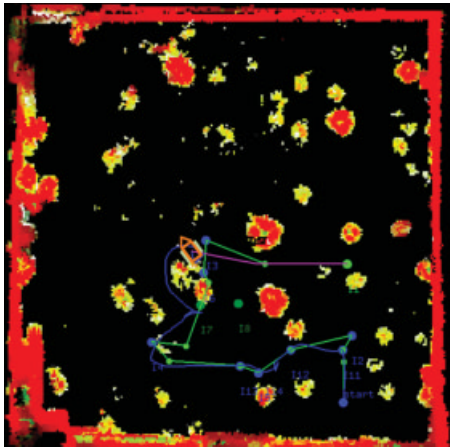


Figure 12. Sample plan shown in the Grid Visualization Tool (GriViT). Green lines show the planned path of the rover, blue lines show the real path, and pink lines show the path that is currently executing.

covered a wide range of scenarios that included the handling of multiple, prioritized science targets, limited time and resources, opportunistic science events, resource usage uncertainty causing under- or over-subscriptions of power and memory, large variations in traverse time, and unexpected obstacles blocking the rover's path.

Our testing scenarios typically consisted of a number of science targets specified at certain locations. A map was used that would represent a sample mission-site location where data would be gathered using multiple instruments at a number of locations. Figure 12 shows a sample scenario that was run as part of these tests. This particular map is of the JPL Mars Yard. The prespecified science targets represented targets that would be communicated by scientists on Earth. These targets were typically prioritized and for many scenarios constraints on time, power, or memory would limit the number of science targets that could be handled. A large focus of these tests was to improve system robustness and flexibility in a realistic environment. Towards that goal, we used a variety of target locations and consistently selected new science targets and/or new science target combinations that had not been previously tested.

Another primary scenario element was dynamically identifying and handling opportunistic science events. For these tests, we have concentrated on a particular type of event, which was finding rocks



Figure 13. Sample image that was taken in response to a science alert indication identification of a rock with white albedo on the JPL FIDO rover.

with distinct features. Specifically, we have tested the feature detection of rocks with certain albedo levels and shape features. These settings were an example of using the data analysis algorithm for target signature, where a particular terrain signature is identified as having a high interest level. If rocks were identified in hazard camera imagery that had a certain interest score, then a science alert was created and sent to the planner. Science alerts would typically come in during rover traverses to new locations, but it was also possible for them to come in while the rover was at a science target location due to a small lag caused by image processing time. If a science alert was detected, the planner attempted to modify the plan so an additional image of the rock of interest was acquired. A sample image that was taken in response to a science alert is shown in Figure 13.

4.1. Testing in Simulation

Since testing with rover hardware can be an expensive and time-intensive process, we ran a large number of tests for the planning and execution module in simulation using a relatively simple simulator. This simulator could execute rover sequence commands and simulate their effects at a coarse level of granularity. For instance, the simulator handled items such as rover position changes and energy usage over straight-line movements, but did not simulate obstacle avoidance or rover kinematics. Another capability that was used in simulation was triggering science alerts at preset or random times. This

capability helped in evaluating the software's capacity to correctly handle different opportunistic science scenarios.

To easily run and evaluate large numbers of tests, we also invested in a testing infrastructure, which allowed tests to be run offline and automatically gathered statistics, including items such as number of plan conflicts found and resolved, plan generation and replanning time, number of goals satisfied, overall plan traverse distance, and plan optimization scores. This testing infrastructure also enabled the automatic creation of mpeg movies that showed plan changes using snapshots of a plan visualization tool. This tool showed the results of plan generation and execution on an overhead map of the world, and could be used for both simulated and hardware testing. An example plan snapshot displayed by this tool is shown in Figure 12. Planning and execution results were evaluated by examining gathered statistics and by viewing created mpegs to flag incorrect or nonoptimal behavior.

4.2. Testing with Rover Hardware

In addition to testing in simulation, a large number of tests with all OASIS software modules were run in the JPL Mars Yard (shown in Figure 14) using different rover hardware platforms. For the past year and for the final demonstration, the FIDO rover (shown in Figure 9) was used for the majority of tests. System setup and the specific rover control software used for capabilities such as navigation and vision was described in Section 2.3.3.

Tests in the Mars Yard typically consisted of 20–50 m runs over a 100 m² area with many obstacles that cause deviations in the rover's path. Most rocks in the Mars Yard are dark in color, thus, for our albedo testing, we brought in a number of whiter rocks to trigger science alerts during rover



Figure 14. The JPL Mars Yard with terrain of various difficulties.

traverses. Science measurements using rover hardware were always images, since other instruments were not readily available (such as a spectrometer). However, different types of measurements were included when testing in simulation.

4.3. Lessons Learned

Integrating and testing with hardware and required control software (e.g., for navigation) introduced a number of challenges. In this section, we will briefly outline a few of the key lessons learned during this process. One primary challenge was that the testing and demonstration of our on-board science system required a number of underlying components to properly perform and support our software. An important lesson for running testing with rover hardware is that a significant amount of time must be allocated to tune and test supporting software and hardware components. Though the majority of supporting components were provided through CLAR-Aty, many had not been run through extensive testing in the environment that we wished to demonstrate OASIS. Some key components that we spent a large amount of time testing for our use include the FIDO rover cameras, navigation software, and position estimation software. This process was further complicated when supporting software had large numbers of tunable parameters and, since some supporting software had only been tested in certain environments or exercised in unit testing, our testing process also uncovered several problems (or bugs) in supporting software that had not been previously discovered. In the future, we hope to have more access to the people who developed the supporting technology and have direct support in correctly tuning their application for our needs. However, since this will not always be the case, we expect future work with hardware testing to consistently require time for tuning supporting technology.

Another lesson learned was the value of performing a large number of tests in simulation, even with a relatively simple simulator. To test the ability of the planning system to handle science alerts under varying conditions, we invested in a testing infrastructure that allowed large numbers of tests to be run offline using a simple simulator. This simulator tracked items such as resource usage and rover position, but did not simulate higher fidelity items such as obstacle avoidance or stereo image processing. During these tests certain factors were varied

such as the number of science alerts, available resources, rover speed, etc. This allowed the planning and execution software to be exercised on a large variety of situations and allowed many bugs to be caught before hardware testing was even performed. Furthermore, problematic situations that were discovered in hardware testing could often be easily reproduced and debugged by using a simulator. Simulated testing also provided an easy way to run large numbers of tests that would have been impossible to run exclusively on hardware, due to the time required to complete them. Overall, our testing in simulation made our system significantly more robust to variations in resource usage, rover position, activity execution time, and frequency of science alerts. Such robustness is particularly important when handling rover operations since factors such as traverse time, power usage, and even possible science opportunities are difficult to predict. We should note that, although testing using a simple simulator was very valuable, it would have also been valuable to perform some testing with higher fidelity simulation. For instance, when testing with rover hardware, obstacle avoidance software often performed differently depending on variations in rover position, sun angle, etc. Since we could not easily reproduce this behavior in our simple simulator, we had to rely on our hardware tests to ensure this behavior did not cause problems or unexpected situations for the OASIS software.

Furthermore, running with hardware often allowed a perspective that was difficult to attain through simulated testing. For example, the accuracy of rover turns towards new science opportunities was much easier to judge when running with hardware.

5. TESTING RESULTS

Here we report results from a set of ten runs with the FIDO rover that were conducted on four different (nonconsecutive) days. The runs had an average of 13 images. The criteria specified was based on albedo—identify rocks that were white. An example is shown in Figure 13. The goal was to detect targets if they are within approximately 3 m of the forward field of view of the rover as it proceeds along the traverse. Each run had between one and seven targets that the rover encountered within the goal detection range. Over the ten traverse test runs, this consisted of 40

total target rocks. Thirty-six of these targets were detected (85% detection rate). There were no false alarms during any of the ten runs.

The 85% detection rate indicates how many of the targets were identified at some time over the course of the traverse. A target may be imaged multiple times during the traverse as the rover approaches and drives by the rock. Over the ten runs, there were 82 instances of targets appearing in an image within the detection range. Individual targets were imaged between one and six times. Of the six targets that were missed, three of these appeared in a single image and three appeared in two images. No target that appeared in more than two images was missed.

5.1. OASIS Component Run Times on FIDO

As part of our testing, we gathered preliminary statistics on the run time of key OASIS components. While we have not spent significant time optimizing the performance of these components, the numbers provide a general idea of current performance and provide a reference to track future improvements. In a representative run, the OASIS rockfinder processed 11 hazcam images resulting in five science alerts being sent to the planner. The rockfinder was run on a 233 Mhz Pentium processor running Vx-Works 5.5 with 128 MB of RAM. Rockfinder took an average of 53 s to process each image and found about nine rocks per image with a total of 103 rocks being identified for the 11 images. The OASIS feature extraction component was run on a 930 Mhz Pentium processor running Linux 2.4 with 256 MB of RAM. For these tests, albedo and shape information were extracted for each rock. For the 11 images processed, feature extraction averaged 0.5 s per image. The planning and scheduling component ran on a 2.5 GHz Pentium processor running Linux 2.4 with 1 GB of RAM. In handling the five science alerts, the planner spent an average of 6 s generating a plan for each alert.

6. CONCLUSIONS AND FUTURE WORK

We have demonstrated an autonomous science system in the field conducting opportunistic science. By integrating data analysis and planning capabilities, the resulting system can operate in a closed-loop fashion. This framework enables new science targets to be addressed onboard with little or no communi-

cation with Earth. An important contribution of this work is closing the loop between the sensor data collection, science goal selection, and activity planning and scheduling. Current approaches require human analysis to determine goals and to manually convert the set of high-level science goals into low-level rover command sequences. By integrating these components onboard, we enable a rover to function autonomously, as if a scientist were always in communication. This type of capability should dramatically increase the science return of future rover missions.

In the future we will expand the features that the system can recognize. We also would like to incorporate information from other sources such as from orbital data in assessing the surface features such as boundaries. Currently, the system does not explicitly recognize the same target or feature from different viewpoints. This is an aspect of information that could be accumulated onboard. For example, observations of a feature as the rover approaches it could be used to update the information on the interest level of the feature. Similarly, when a feature is viewed from a different angle, this would be valuable information. In addition, information from different instruments could be incorporated into the onboard analysis to identify regions and individual features of high science interest.

The Deep Space Network will remain a constraining resource for future deep space missions as the number of high bandwidth missions increases. Traditional data compression can provide a valuable mechanism for increasing the amount of useful data returned; however, a limited amount of compression is possible before distortion levels become intolerably high. Science return can be maximized by returning the data with the highest science content possible. The use of onboard analysis to identify opportunities and select the data with the highest scientific interest will be a critical functionality to maximize science return on future deep space missions with high data volume instruments.

ACKNOWLEDGMENTS

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

was sponsored by the Mars Technology Program, the Intelligent Systems Program and the Interplanetary Network Technology Program.

Finally, our work this year could not proceed without the support and advice of scientists outside of the team, particularly Albert Haldemann (JPL) and Matt Golombek (JPL).

REFERENCES

- Alami, R., Chatila, R., Fleury, S., Ghallab, M., & Ingrand, F. (1998). An architecture for autonomy. *International Journal of Robotics Research* 17(4), 315–337.
- Bonasso, R., Firby, R., Gat, E., Kortenkamp, D., Miller, D., & Slack, M. (1997). Experiences with an architecture for intelligent, reactive agents. *Journal of Experimental and Theoretical Artificial Intelligence Research* 9(2/3), 237–256.
- Bresina, J., Golden, K., Smith, D., & Washington, R. (1999, June). Increased flexibility and robustness of Mars rovers. *Proceedings of the International Symposium on AI, Robotics, and Autonomous for Space (iSAIRAS)*, Noordwijk, Netherlands.
- Castano, A., Fukunaga, A., Biesiadecki, J., Neakrase, L., Whelley, P., Greeley, R. et al. (2007). Autonomous detection of dust devils and clouds at Mars. To appear *Machine Vision and Applications*.
- Castano, A., Fukunaga, A., Biesiadecki, J., Neakrase, L., Whelley, P., Greeley, et al. (2006, October). Autonomous detection of dust devils and clouds on Mars. *International Conference on Image Processing*, Atlanta, GA.
- Castano, R., Mann, T., & Mjolsness, E. (1999, July) Texture analysis for Mars rover images. In *Applications of Digital Image Processing XXII*, *Proceedings of SPIE*, Vol. 3808, pp. 162–173, Denver, CO.
- Castano, R., Anderson, R.C., Estlin, T., Decoste, D., Fisher, E., Gaines, D. et al. (2003, March). Rover traverse science for increased mission science return. *Proceedings of the 2003 IEEE Aerospace Conference*, Big Sky, MT.
- Castano, R., Judd, M., Estlin, T., Anderson, R.C., Scharenbroich, L., Song, L. et al. (2004, March). Autonomous onboard traverse science system. *Proceedings of the 2004 IEEE Aerospace Conference*, Big Sky, MT.
- Castano, R., Judd, M., Estlin, T., Anderson, R.C., Gaines, G., Castaño, A. et al. (2005, March). Current results from a rover science data analysis system. *Proceedings of the 2005 IEEE Aerospace Conference*, Big Sky, MT.
- Castano, R., Estlin, T., Gaines, G., Castaño, A., Chouinard, C., Bornstein, B. et al. (2006, March). Opportunistic rover science: finding and reacting to rocks, clouds and dust devils. *Proceedings of the 2006 IEEE Aerospace Conference*, Big Sky, MT.
- Cheng, Y., Maimone, M., & Matthies, L. (2005). Visual odometry on the Mars exploration rovers. *Proceed-*

- ings of International Conference Systems, Man, and Cybernetics, Vol. 1, pp. 903–910, Hawaii, HI.
- Chien, S., Sherwood, R., Tran, D., Cichy, B., Rabideau, G., Castano, R. et al. (2005). Using autonomy flight software to improve science return on Earth Observing One. *Journal of Aerospace Computing, Information, and Communication* 2(4), 196–216.
- Chien, S., Knight, R., Stechert, A., Sherwood, R., & Rabideau, G. (2000, April). Using iterative repair to improve the responsiveness of planning and scheduling. *Proceedings of the Fifth International Conference on Artificial Intelligence Planning and Scheduling*, Breckenridge, CO.
- Estlin, T., Gaines, D., Chouinard, C., Castano, R., Bornstein, B., Judd, M. et al. (2007, April). Increased Mars rover autonomy using AI planning, scheduling and execution. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2007)*, Rome, Italy.
- Fox, J., Castano, R., & Anderson, R.C. (2002, March). Onboard autonomous rock shape analysis for Mars rovers. *Proceedings of the 2002 IEEE Aerospace Conference*, Big Sky, MT.
- Gat, E. (1991). Integrating planning and reacting in a heterogeneous asynchronous architecture for mobile robots. *SIGART Bulletin* 2, 70–74.
- Gilmore, M., Castano, R., Mann, T., Anderson, R.C., Mjolsness, E., Manduchi, R. et al. (2000). Strategies for autonomous rovers at Mars. *Journal of Geophysical Research* 105(E12), 29223–29237.
- Gulick, V.C., Morris, R.L., Ruzon, M.A., & Roush, T.L. (2001). Autonomous image analysis during the 1999 Marsokhod rover field test. *Journal of Geophysical Research* 106(E4), 7745–7764.
- Halir, R., & Flusser, J. (1998). Numerically stable direct least-squares fitting of ellipses. *Proceedings of the 6th International Conference in Central Europe on Computer Graphics, Visualization and Interactive Digital Media (WSCG)*, Plzen, Czech Republic, 125–132.
- Hentschel, M.L., & Page, N.W. (2002). Selection of descriptors for particle shape characterization. *Particle & Particle Systems Characterization*, 20(1), 25–38.
- Mahmood, A., Mitchell, J.K., & Carrier, W.D. (1974). Grain orientation in lunar soil. *Lunar and Planetary Science Conference*, 5, Houston, TX, 2347–2354.
- Maimone, M., Biesiadecki, J., Tunstel, E., Cheng, Y., & Leger, C. (2006) Surface navigation and mobility intelligence on the Mars Exploration Rovers. In *Intelligence for Space Robotics*, (pp. 45–69). Albuquerque, NM: TSI Press.
- Mishkin, A., Morrison, J., Nguyen, T., Stone, H., Cooper, B., & Wilcox, B. (1998, Mar.). Experiences with operations and autonomy of the Mars Pathfinder micro-rover. *Proceedings of the 1998 IEEE Aerospace Conference*, Aspen, CO.
- Nenas, I.A., Wright, A., Bajracharya, M., Simmons, R., Estlin, T., & Kim, W.S. (2003, Apr.) CLARAty: an architecture for reusable robotic software. *SPIE Aerosense Conference*, Orlando, FL.
- Pedersen, L. (2001). *Robotic Rock Classification and Autonomous Exploration*, Ph.D. thesis, Robotics Institute, Carnegie Mellon University (CMU-RI-TR-01-14). Pittsburgh, PA.
- Rabideau, G., Engelhardt, B., & Chien, S. (2000, Apr.). Using generic preferences to incrementally improve plan quality. *Proceedings of the International Conference on Artificial Intelligence Planning Systems (AIPS 2000)*, Breckenridge, CO.
- Simmons, R. & Apfelbaum, D. (1998, October). A task description language for robot control. *Proceedings of the Intelligent Robots and Systems Conference*, Vancouver, Canada.
- Smith, T., Niekum, S., Thompson, D., & Wettergreen, D. (2005, March) Concepts for science autonomy during robotic traverse and survey. *Proceedings of IEEE Aerospace*, Big Sky, MT.
- Smyth, P. (1996). Clustering using Monte-Carlo cross-validation. *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, AAAI Press.
- Thompson, D.R., Smith, T., & Wettergreen, D. (2005a). Data mining during rover traverse: from images to geologic signatures. *International Symposium on Artificial Intelligence, Robotics and Automation in Space (iSAIRAS)*, Darmstadt, Germany.
- Thompson, D., Niekum, S., Smith, T., & Wettergreen, D., (2005b). Automatic detection and classification of features of geologic interest. *Proceedings of the 2005 IEEE Aerospace Conference*, Big Sky, MT.
- Thompson, D.R., & Castano, R. (2007, March) A performance comparison of rock detection algorithms for autonomous planetary geology. *Proceedings of the 2007 IEEE Aerospace Conference*, Big Sky, MT.
- Tomasi, C., & Manduchi, R. (1998, January) Bilateral Filtering for Gray and Color Images. In *International Conference on Computer Vision (ICCV)*, pp. 839–846, Bombay, India.
- Trucco, E., & Verri, A. (1998). *Introductory Techniques for 3D Computer Vision*, p. 343, Upper Saddle River, NJ: Prentice-Hall.
- Wadell, H. (1932). Volume, shape and roundness of rock particles. *Journal of Geology* 40: 443–451.
- Wagner, M.D., Apostolopoulos, D., Shillcutt, K., Shamah, B., Simmons, R.G., & Whittaker, W. (2001, May). The science autonomy system of the Nomad Robot. *Proceedings of the International Conference on Robotics and Automation (ICRA 2001)*, pp. 1742–1749, Seoul, Korea.