

# Increased Mars Rover Autonomy using AI Planning, Scheduling and Execution

Tara Estlin, Daniel Gaines, Caroline Chouinard, Rebecca Castano,  
Benjamin Bornstein, Michele Judd, Issa Nesnas, and Robert Anderson

**Abstract**—This paper presents technology for performing autonomous commanding of a planetary rover. Through the use of AI planning, scheduling and execution techniques, the OASIS autonomous science system provides capabilities for the automated generation of a rover activity plan based on science priorities, the handling of opportunistic science, including new science targets identified by onboard data analysis software, other dynamic decision-making such as modifying the rover activity plan in response to problems or other state and resource changes. We first describe some of the particular challenges this work has begun to address and then describe our system approach. Finally, we report on our experience testing this software with a Mars rover prototype.

## I. INTRODUCTION

THE Mars Exploration Rovers (MER 2003) have traveled many kilometers over the Martian soil and have outperformed all expectations by lasting an order of magnitude longer than their original mission goal. Both rovers have now lasted over 1000 sols (or Martian days), whereas their primary mission was for 90 sols. The longevity of these vehicles will have significant effects on future mission goals, such as objectives for the Mars Science Laboratory rover mission which is scheduled to fly in 2009. Current mission operations for MER require a team of mission engineers to plan rover activities for each sol. Though many planning activities have been significantly streamlined as compared to days early in the mission, the large majority of rover activities are sequenced manually with some aid from ground software packages that help model rover behavior.

Common objectives for future rover missions to Mars include the handling of opportunistic science and long-range, or *over the horizon*, driving. Though some software already exists onboard the MER rovers to support these goals, additional autonomous capabilities will be required. To handle newly-identified science targets, onboard software will need to schedule new rover activities based on specific target data and current rover state. To handle long-range traverse, onboard software will need to closely monitor rover state and resources, as well as dynamically modify rover activity start-times and orderings since rovers will have to

navigate through unknown terrain.

Some autonomous capabilities for rovers have been well tested on the Mars surface. For example, the MER navigation algorithm has been used to drive the rovers up to 140 meters in one sol. Future rover missions will likely continue to use this capability and increase allowable distances for autonomous driving. This increase will also create new requirements for other onboard autonomy software that can closely monitor rover state and resources (to ensure both the safety of the rover and the most efficient, use of such resources). Further onboard software will need to react to unforeseen opportunities during drives that could be missed if the rover could not handle opportunistic science events. Onboard data analysis is already being used on MER to automatically detect dust devils and other work is being done to develop analysis software that can detect a large variety of image features [1, 2].

This paper presents the planning, scheduling, and execution component of the OASIS autonomous science system [1]. This component provides capabilities for initial plan generation, plan execution and monitoring, and re-planning in response to new science opportunities and/or changes in rover state or environment. This work has been focused on supporting Mars exploration and science goals and specifically on supporting the concepts of opportunistic science and long-distance traverse. In particular, this paper provides an overview of how the OASIS planning and execution system has been tested both through simulation and with actual rover hardware in the JPL Mars Yard. Numerous tests have been run showing that OASIS can robustly respond and re-plan for a variety of situations, including the handling of new science goals, dynamic changes in resources levels, unknown obstacles, and autonomous targeting based on data analysis results. OASIS is also integrated with a large set of robotic control software that provides Mars mission-relevant capabilities such as autonomous navigation, vision, and manipulation.

Past OASIS publications have given a high-level system description or details on the data analysis components of OASIS [1]. This paper focuses and provides details on the planning and execution elements of the system.

## II. CHALLENGES FOR ONBOARD AUTONOMY

Rovers that possess extended autonomy capabilities, such as onboard command generation and re-planning, have the potential for not only supporting the more efficient use of

Manuscript received September 15, 2006. This work was supported by the NASA Mars Technology Program and NASA Intelligent Systems Program.

All authors are with the Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109 USA (email: [firstname.lastname@jpl.nasa.gov](mailto:firstname.lastname@jpl.nasa.gov), i.e., [Tara.Estlin@jpl.nasa.gov](mailto:Tara.Estlin@jpl.nasa.gov)).



Fig. 1. View of the Dome Fuji target on the Mars surface taken by the MER Spirit rover.

rover resources over longer periods but also for enabling data collection opportunities that might have never been realized had the autonomy software not been onboard. New missions are being designed that will require rovers to support more autonomous endeavors such as long-range traversals, complex science experiments, opportunistic science handling, and lengthy mission durations. To address these goals, autonomy software designers face a number of challenges. In this section, we consider a few key challenges for using planning and execution techniques to provide extended rover-autonomy capabilities.

To generate or modify a rover activity plan for carrying out a set of science goals, the onboard planning and execution software will need to reason about a rich model of resource and temporal constraints. For example, it will need 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.

Sequence generation for rover missions raises a number of interesting challenges regarding spatial reasoning. One of the dominating characteristics of rover operations is drives to designated waypoints and science targets through either partially-known or unknown terrain. (Figure 1 shows a view of Martian terrain captured during early parts of the MER mission.) Onboard planning and execution needs to effectively coordinate with several layers of navigation and other spatial reasoning software in order to generate correct and efficient rover plans. This coordination typically includes querying a path planner for route options, using position estimators to track rover progress and re-plan if progress is impaired, and monitoring navigation and obstacle avoidance software that may cause the rover to move off the predicted route.

Another predominant challenge in developing autonomy software for rover operations is robustly handling the inherent uncertainty of rover behavior. Since rovers are typically driving in unknown terrain, important state knowledge, such as rover position or activity duration, is often difficult to accurately predict. The MER rovers have seen wheel slippage cause significant position errors, such as the 21% error accrued when traversing Eagle crater and the 100% slippage error that occurred when Opportunity got stuck in a Purgatory dune [3]. Rover resource levels will also fluctuate based on how certain operations interact with the terrain, including drives and instrument placements. Some resources (such as power) may be affected by other environment variables such as sun angle and atmosphere dust opacity. The difficulty of operating in unknown terrain is further compounded by the tight resource and state constraints a rover faces. For instance, for much of 2006, the MER Spirit rover had such a low power level (due to the angle of the sun in Martian winter), that it could only perform a small number of stationary operations per sol.

Onboard autonomy software should also be able to effectively reason about plan quality and science. Over the course of a mission, a rover will be asked to perform a large variety of science operations. Different priorities will exist for these measurements and priorities may fluctuate throughout the mission. When resources are over-taxed, the rover should also be capable of making science/resource trade-offs in an effort to produce the highest science return. Engineering objectives may also exist that should be considered such as leaving the rover at a desirable tilt for acquiring solar panel energy or including observations that take images of rover hardware to help evaluate system state.

### III. OASIS AUTONOMOUS SCIENCE SYSTEM

To address the issues outlined in the previous section, we have developed a planning, scheduling and execution system that can autonomously command a planetary rover. This software is part of a larger framework called OASIS (Onboard Autonomous Science Investigation System) [1], which is shown in Figure 2. OASIS provides a number of capabilities for performing rover autonomous science, including image and data analysis, planning, execution and interaction with robotic control. This paper focuses on the planning, scheduling and execution element of this framework. Other components will only be briefly described, but are further detailed in related publications.

#### A. OASIS Planning, Scheduling and Execution

In the OASIS framework, planning, scheduling and execution techniques are used to provide rover-plan generation, execution, monitoring, and re-planning in response to problems and new science opportunities. To provide these capabilities, OASIS closely integrates the CASPER (Continuous Activity Scheduling, Planning, Execution and Re-planning) planning system and the TDL (Task Description Language) executive system. The following capabilities are provided:

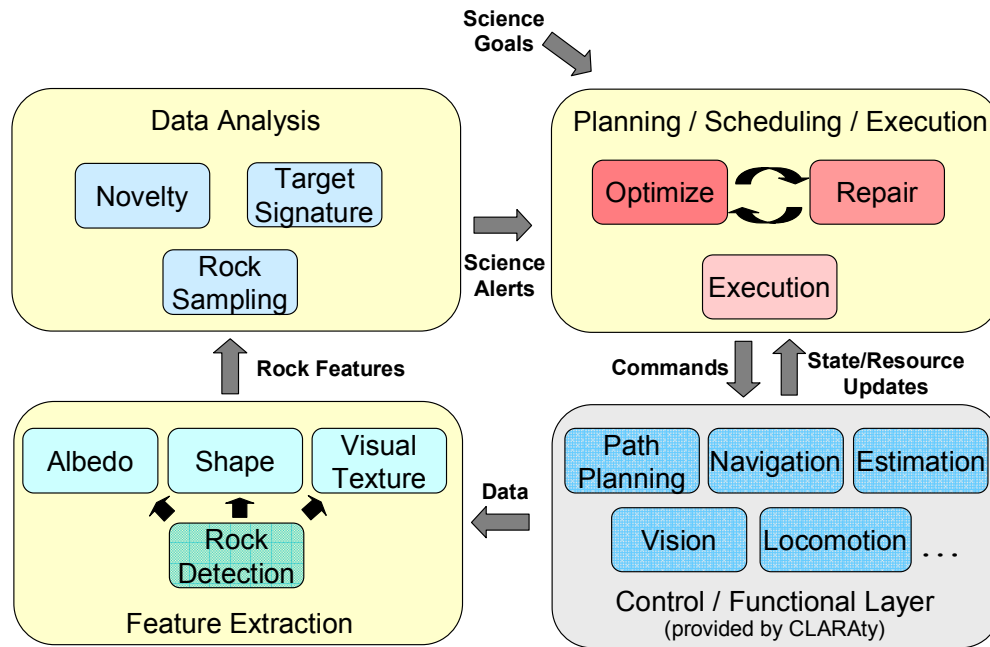


Fig 2. OASIS (Onboard Autonomous Science Investigation System) framework. This diagram shows how different decision-making capabilities interact within OASIS (shown in the yellow boxes) and how OASIS interacts with low-level control software (shown in the gray box).

- Automated plan generation based on an input set of science and engineering goals
- Maintenance of resource, state, and temporal constraints
- Plan execution through the interaction with basic rover control functionality
- Monitoring of plan execution to ensure plan activities are successfully executed
- Dynamic modification of the plan based on activity, state or resource updates or new goal information
- Plan optimization for reasoning about plan quality
- Automated handling of newly identified science goals

Planning in OASIS is provided by the CASPER continuous planning system [4]. For initial plan generation, CASPER is given a model of rover operations and an input set of science and engineering goals (e.g., take a panoramic image at a certain azimuth and elevation). CASPER generates a sequence of activities that satisfies as many goals as possible while obeying relevant resource, state and temporal constraints. For instance, plans may have strict limits on energy usage and science activities may be required to occur during certain time windows. Plan activities are grounded in time, enabling CASPER to quickly determine plan conflicts and solutions. Conflicts occur when a plan constraint has been violated where this constraint could be temporal or involve a resource, state or activity parameter. Plans are produced using an *iterative repair* algorithm that classifies conflicts and resolves them individually by performing one or more plan modifications.

To provide re-planning capabilities, CASPER monitors current rover state and the execution status of plan activities.

As this information is acquired, CASPER updates future-plan projections. These updates may cause new conflicts and/or opportunities to arise, requiring CASPER to replan to accommodate the new data. A simple re-planning situation is shown in Figure 3. Here, a rover science activity (of drilling for sample collection) took more power than originally estimated, which causes a resource conflict. CASPER fixes the conflict by deleting a low-priority science activity to ensure that enough time remains to execute a later, high-priority science activity and a required communications link.

To reason about science goal priorities and other plan quality measures we use the CASPER optimization framework to search for a higher quality plan. User-defined preferences are used to compute plan quality based on how well the plan satisfies these preferences. An overall plan score is computed based on the preference specification. Plan optimization works in an iterative fashion (similar to plan repair) and searches for plan modifications that could potentially improve the overall plan score.

Plan execution in OASIS is handled by the TDL execution system [5]. TDL was designed to handle intermediate control activities for a robotic system and to mediate between a planning system and low-level robot control software. It expands activities into lower-level commands, executes the commands, and monitors their execution. It also provides support for exception handling and fine-grained timing control such as synchronization of subtasks. TDL is implemented as an extension of C++ and uses a construct called a *task tree* to describe the tree structure that is produced when tasks are broken down into lower-level commands. In OASIS, TDL handles expanding high-level activities into lower-level commands (based on current state

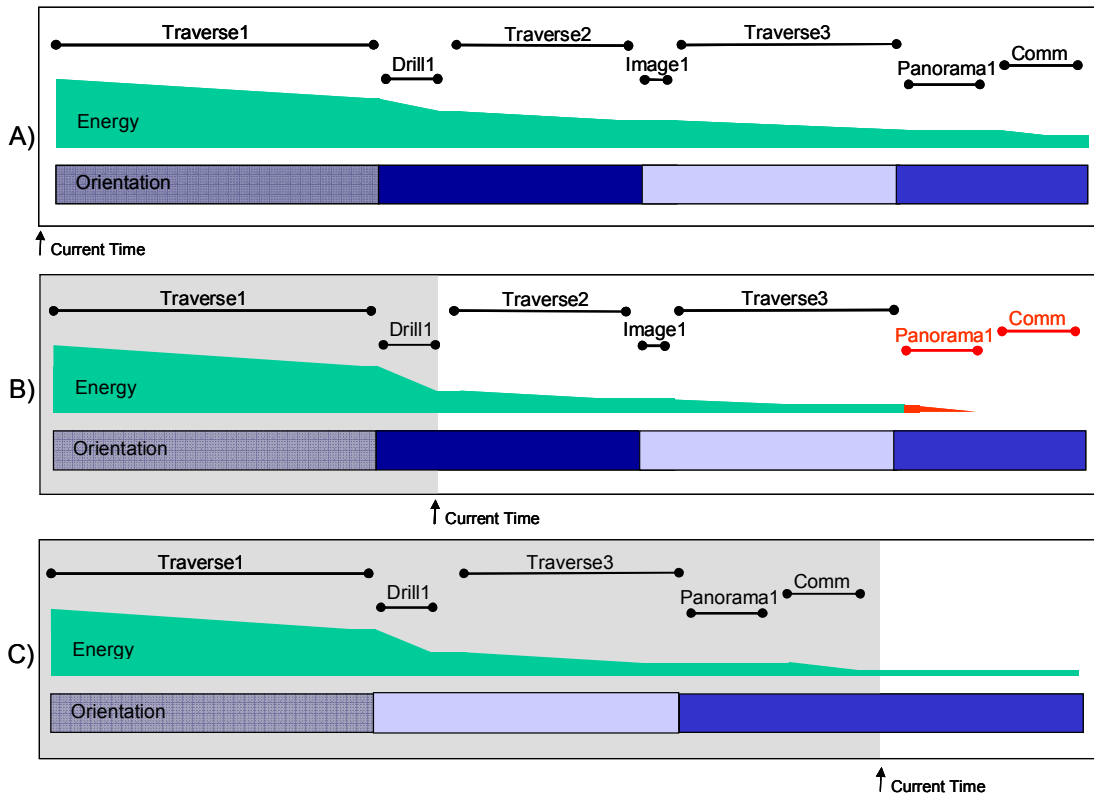


Fig. 3. Example re-planning scenario. A) The initial plan generated by CASPER which drives the rover to three locations, performs a science operation at each location, and finally performs a critical end-of-day communications link with Earth. B) The plan after partial execution. The first science activity has taken more energy than expected, which has caused several resource conflicts with later activities. C) The plan after re-planning has occurred to resolve the conflict. The second science operation and drive were dynamically removed so the last (higher priority) science operation and communication activity can still be achieved.

information), executing commands by dispatching them to functional-level control software, relaying updates on activity status and state/resource data to CASPER, tracking plan execution progress, and relaying identified problems to CASPER.

#### B. OASIS Feature Extraction and Data Analysis

The OASIS Feature Extraction and Data Analysis modules (shown in Figure 2) are responsible for analyzing onboard image data from rover cameras, identifying interesting or novel parts of that data, and generating new science goals to gather additional measurements when new science targets have been identified. Analyzed images could be taken by either rover science or engineering cameras and could be shot during a variety of rover activities including obstacle avoidance imaging, high-resolution science imaging, and imaging done for long-range route planning.

As shown in Figure 2, new science data is first processed by the Feature Extraction component. Images are broken down by first locating individual rocks, and second, by extracting a set of rock properties (or features) from each identified rock. Extracted rock properties (e.g., shape, albedo, visual texture, size) are then passed to the Data Analysis component of the system. This component consists

of several prioritization algorithms, which analyze the data by searching for items such as rocks with features that match pre-known signatures of interest (as identified by scientists on Earth), or novel rocks (i.e., outliers) that have not been seen in past activities. If the analysis component detects new science opportunities of significant interest, it will generate a *science alert* that is sent to the planner, which signifies that new measurements should be taken of the identified target.

#### C. CLARAty Robotic Architecture

The systems described in this paper are also integrated with the Coupled Layered Architecture for Robotic Autonomy (CLARAty) [6], which provides a large range of basic robotic functionality and simplifies the integration of new technologies on different robotic platforms. Through CLARAty, the OASIS planning and execution system has been tested with several JPL rover platforms, including the FIDO rover, which is shown in Figure 4.

To run realistic scenarios with rover hardware, a number of supporting pieces of software were used. These components were provided through CLARAty and could run on the relevant hardware platforms. This software includes the Morphin navigation system, which enables a rover to avoid obstacles and navigate to specified waypoints. It also



Fig. 4. The FIDO rover, which was used for testing.

includes a position estimation algorithm, which integrates IMU (Inertial Measuring Unit) measurements with wheel odometry to estimate rover position and attitude (roll, pitch and heading). Other used algorithms include mobility and stereo processing as well as control functions for mast pan/tilt and camera operation.

#### IV. OPPORTUNISTIC TRAVERSE SCIENCE

To handle opportunistic science during rover traverses, we enabled the OASIS planner, CASPER, to recognize and respond to *science alerts*, which as described above, are new science opportunities detected by the onboard OASIS data analysis software. For example, if a rock is detected in navigation imagery and the rock has a previously unseen color or texture, a science alert can be generated to take additional measurements of that rock.

CASPER uses its optimization framework to decide how to respond to science alerts. Because it may not be possible to accommodate all alerts (due to other constraints or activities), a science alert is represented as an optional goal, which indicates 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 the quality has not increased after a certain time limit, the previous plan is restored. If CASPER can handle a new science alert (e.g., by adding additional science measurements) without causing other negative affects (e.g., resource over-subscriptions or the deletion of higher-priority science goals) then the new plan that accommodates the science alert is used.

Science alerts can have different levels of reaction in OASIS. 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 at the next communication opportunity. The next level of reaction is to collect additional data at the current site before transmitting back to Earth. Further steps include having the rover alter its path to get closer to objects of interest before taking additional measurements and taking a close-contact measurement.

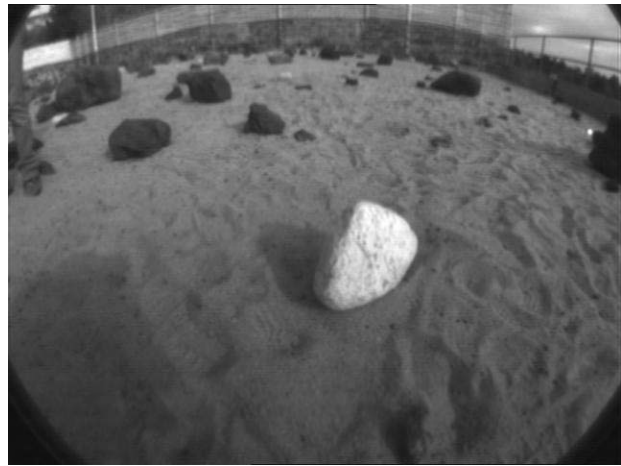


Fig. 5. Image taken in response to a science alert on the JPL FIDO rover. In this test, science alerts were generated for “high albedo” (or light-colored) rocks.

The first three levels of response (calling Earth, taking additional measurements from the rover current's position, and driving closer to the object of interest to take measurements) have been implemented and significantly tested with hardware. A sample image taken in response to a science alert during testing is shown in Figure 5.

#### V. AUTONOMOUS TARGETING

Another objective for future rover missions is to provide autonomous targeting for measurements of science targets that cannot easily be selected in advance. A number of remote sensing instruments for rovers have a very narrow field-of-view (FOV) and thus require very specific targets for correct sampling. Examples of such instruments include the MER mission Mini-Thermal Emission Spectrometer (or mini-TES) and the ChemCam spectrometer, which performs Laser-Induced Breakdown Spectroscopy and is planned to be flown as part of the Mars Science Laboratory (MSL) 2009 rover mission. Targeting of these instruments by mission personnel requires a lengthy planning process, since images of the targeted areas will need to be first analyzed on Earth.

One scenario suggested for the MSL ChemCam instrument is to autonomously select and shoot a number of targets selected by onboard analysis software. New measurements would then be autonomously scheduled onboard the rover. To address this scenario, we applied our planning and execution system to both execute a set of images from which targets could be selected and to incorporate new science measurements based on results from data analysis. Similar techniques could be applied as were used to support traverse science, where measurements were only added if they did not disrupt other high priority activities or require an excess of resources. The OASIS planner can also handle these new measurement requests at different frequencies; in other words, requests could either be sent as available (in an iterative fashion as image data is



analyzed) or they could be sent as one batch set.

## VI. ADDITIONAL PLAN IMPROVEMENTS AND FAULT HANDLING

Another key area where plan optimization techniques were used to improve the rover plan was when extra time or resources became dynamically available during plan execution. Since traverse times and resource usage are difficult to predict, it is often the case that a rover operation takes less time or power than expected. For instance, a traverse could take much less time than expected due to benign terrain. For these cases, the planning optimization framework was used to dynamically add additional science goals to the plan that could not be fit in the original plan due to time and resource constraints. This capability enables the scenario where scientists on the ground specify a number of prioritized science goals, but not all of them may be achievable due to limited rover resources. However, some goals (that were not included in the initial plan based on resource estimates) may eventually be fit into the plan if resource usage is lower than predicted.

On the converse side, rover traverse activities and some science activities may take longer or more resources than expected. Just as terrain could be more benign than originally estimated, it could also be significantly more difficult. In this scenario, the OASIS planning and execution can dynamically adjust the plan to ensure that resources are not over-subscribed during plan execution. The CASPER planner can often delete a low priority science activity during plan execution in order to ensure that higher priority science and engineering activities will be correctly executed. Further, it can also be detected that an activity is behind schedule during execution. If it falls significantly behind schedule, the TDL executive can fail the activity and request that CASPER adjust the plan to better accommodate the current situation. If new obstacles have been detected, CASPER may re-order (or delete) science targets to ensure more high priority targets are correctly sampled.

## VII. SYSTEM TESTING WITH HARDWARE AND SIMULATION

To evaluate our system we have performed a large series of tests both in simulation and using rover hardware. These tests 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 often consisted of a random 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 6 shows a sample scenario that was run as part of these tests. This particular map is of the JPL Mars Yard. The pre-specified science targets (shown in Figure 6 as the larger circles) represented targets that would be communicated by scientists on Earth. These targets were

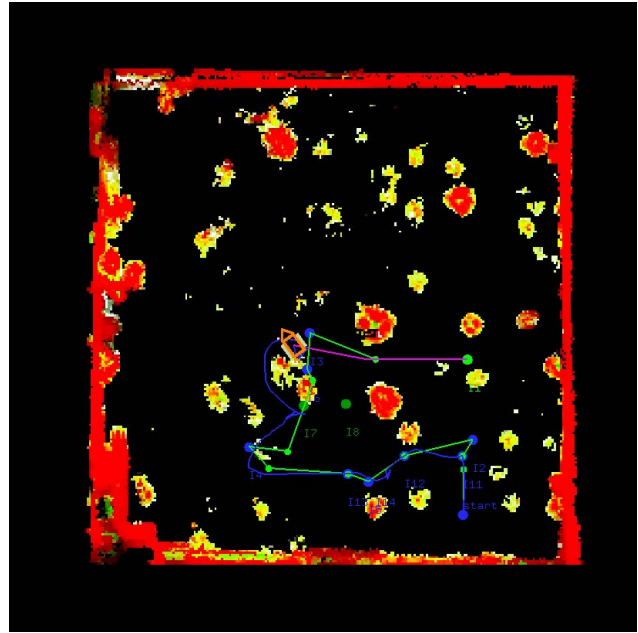


Fig. 6. Sample plan used in testing. Green lines show the planned path of the rover; blue lines shown the real path; and pink lines show the path that is currently executing.

typically prioritized and 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.

A primary scenario element was dynamically identifying and handling opportunistic science events. For these tests, we used onboard data analysis software to generate science alerts based on a target rock signature. Various types of signatures were used, but they typically corresponded to a combination of target albedo (brightness) levels, shape characteristics, and size estimations. If rocks were identified in rover camera imagery that had a high score for these features, then a science alert was created and sent to the planner. Science alerts often happened during rover traverses to new locations, but they were also used for testing autonomous targeting at the end of a traverse. If a science alert was detected, the planner attempted to modify the plan so an additional image of the rock of interest would be acquired. A sample image that was taken in response to a science alert was shown in Figure 5.

Other important scenario elements included adding or deleting ground-specified science targets based on resource under- or over-subscriptions. For instance, in some tests, the rover covered distances more quickly than estimated and the planner was able to add additional science targets that could not be fit into the original plan. Conversely, in other tests, the rover used more energy than expected during traverses or science activities, which often caused a power over-

subscription, where enough power was not being preserved for later plan activities. The planner resolved this situation by deleting some lower priority science targets. Unexpected energy drops during a traverse could also be handled by the executive, which detects the shortfall and stops the current traverse if there is not enough energy to complete it. In all cases, the planning and execution system attempts to preserve as many high priority science targets as possible while still adhering to required constraints.

#### A. Testing in Simulation

Since testing with rover hardware can be an expensive and time-intensive process, we ran a large number of tests 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 multiple science alerts at pre-set or random times. This capability helped in evaluating the planner'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 statistics automatically gathered, including information such as number of plan conflicts found and resolved, plan generation and re-planning time, number of goals satisfied, overall plan traverse distance and plan optimization scores. This testing infrastructure also could create mpeg movies of runs that showed plan changes using snapshots of a plan visualization tool.

#### B. Testing with Hardware

In addition to testing in simulation, a large number of tests have been run in the JPL Mars Yard using different rover hardware platforms. For the past few years, the FIDO rover (shown in Figure 4) was used for the majority of tests. FIDO is an advanced technology prototype rover similar to the MER rovers. FIDO's mobility sub-system consists of a six-wheel rocker-bogie suspension capable of traversing over obstacles up to 30 cm in height.

All OASIS software has been designed to run onboard the rover, however during testing, only functional-level CLARAty modules, such as navigation and vision, and the OASIS rockfinding software were run onboard FIDO. Other modules, including the planning and execution module and the analysis module, were run on offboard workstations that communicated with the rovers using Wireless Ethernet, since a port of these components to the onboard operating system (VxWorks) was not complete.

Tests in the Mars Yard typically consisted of 20-50 meter runs over a 100 square meter area with a range of obstacles that caused deviations in the rover's path. Science measurements using rover hardware were images from one of three sets of cameras on the rover (hazard cameras, navigation cameras and panoramic cameras). Other

instruments, such as spectrometers, were not readily available and thus not directly incorporated into hardware tests. However different types of measurements were included when testing in simulation.

#### C. Testing and Demonstration Summary

Testing in simulation and with real hardware provided important steps in the evaluation of our system. Many bugs were caught early through simulated testing, but others did not surface until significant runs had been performed on rover hardware. Furthermore, running with hardware often allowed a perspective that was difficult to attain through simulated testing. For example, the effects of rover slippage were much easier to visualize and test when using hardware.

A number of live demonstrations of our system have been performed, including a several hour long demonstration which showed the system successfully handling a random combination of science targets and science alerts (that had not been used in previous testing) and resulted in over 40 meters of autonomous driving. This demonstration consisted of several runs that showed scenario elements such as handling new science alerts, dynamically adding new ground-specified science when time became available, and deleting low priority science targets in a later run where more power was used than originally estimated. The OASIS software operated correctly in all tested cases.

Another live demonstration showed a combination of traverse science and autonomous targeting. Autonomous targeting was supported by adding a targeting subplan that starts with a FIDO navigation camera panorama that is analyzed online for new targets. Autonomous, new measurements were then scheduled and taken by the high-resolution FIDO panoramic cameras, which were being used as an example limited FOV instrument.

## VIII. RELATED WORK

A number of planning and executive systems have been used for robotic applications. One approach directed towards rover command generation used a Contingent Planner/Scheduler (CPS) that was developed to schedule rover science operations using a Contingent Rover Language (CRL) [7]. CRL allows both temporal flexibility and contingency branches in command sequences. Contingent sequences are produced by the CPS planner and then interpreted by an executive, which executes the final plan by choosing sequence branches based on current conditions. As compared to OASIS, only the executive is onboard the rover; planning is a ground-based operation and does not involve re-planning. Since only a limited number of contingencies can be anticipated and incorporated into the plan, CRL does not provide as much flexibility as OASIS when adjusting the sequence in response to unexpected events.

The LAAS-CNRS lab robotic control architecture [8] also uses onboard planning and execution to create initial plans and to provide re-planning capabilities. However, as

compared to CASPER, the IxTeT planner uses a partial order CSP-based planning approach, which can require larger amounts of time for re-planning since a valid plan must be found at every search step. Further this system has not addressed the handling of opportunistic science and interacting with an onboard data analysis system.

Other similar approaches include Atlantis [9] and 3T [10], which used a deliberative planner and an executive on top of a set of reactive controllers. These approaches only use a batch planning approach in a limited fashion do not provide online re-planning or support for opportunistic science.

The Autonomous Sciencecraft Experiment (ASE) [11] has demonstrated the capability of planning and data analysis systems to coordinate the behavior of the EO-1 Earth orbiting satellite. The Remote Agent Experiment (RAX) [12] demonstrated the ability of an AI planning and execution system to generate and execute plans onboard the NASA Deep Space One (DS1) spacecraft. RAX, however, used a batch approach to planning and could not dynamically re-plan. Further, since RAX and ASE were applied to spacecraft, neither handle the large uncertainty inherent in surface navigation and science.

#### IX. OTHER CURRENT AND FUTURE WORK

Towards the goal of enabling our planning and execution system to better reason how well science measurements characterize the surrounding environment, we are in the process of extending OASIS to evaluate how well a set of observations spatially covers a specified area of terrain [13]. Scientific observations can be ranked through our algorithm by their ability to improve coverage of a large geological feature or area. This quality measure is one of many that could be considered in future rover missions, but it has important significance in enabling our system to balance the science goals of detailed study vs. broad coverage.

In future work, we plan to extend our capabilities for opportunistic science handling to include adding observations for different types of science instruments and performing close-contact measurements for high priority alerts. We also intend to extend our system to handle area surveying, where all rocks (or other terrain features) within a certain area would be properly examined and catalogued.

#### X. CONCLUSION

This paper discussed a number of challenges for using planning, scheduling and execution techniques to provide autonomous rover capabilities for future NASA missions. We described our system approach and explained how it provides capabilities for sequence generation, execution, monitoring, re-planning, sequence optimization, opportunistic science handling, and autonomous targeting. Through a series of tests in simulation and on rover hardware, we have demonstrated our system's ability to robustly respond to unexpected problems and to take advantage of unforeseen opportunities, thus achieving high utilization of rover resources.

#### ACKNOWLEDGMENT

This work was performed by the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

#### REFERENCES

- [1] R. Castano, T. Estlin, D. Gaines, A. Castano, C. Chouinard, B. Bornstein, R. C. Anderson, S. Chien, Al. Fukanga and M. Judd, "Opportunistic Rover Science: Finding and Reacting to Rocks, Clouds, and Dust Devils," *Proceedings of the 2006 IEEE Aerospace Conference*. Big Sky, Montana, March 2006
- [2] D. Thompson, T. Smith and D. Wettergreen, "Data Mining During Rover Traverse: From Images to Geological Signatures," *Proceedings of the Eighth Int'l Symposium on AI, Robotics and Automation in Space*, Sept 2005.
- [3] R. Li, K. Di, et al., "Incremental Bundle Adjustment Techniques Using Networked Overhead and Ground Imagery for Long-Range Autonomous Mars Location," *Proceedings of the Eighth Int'l Symposium on AI, Robotics and Automation in Space*, Sept 2005.
- [4] S. Chien, R. Knight, A. Stechert, R. Sherwood, and G. Rabideau, "Using Iterative Repair to Improve the Responsiveness of Planning and Scheduling," *Proceedings of the Fifth Int'l Conference on AI Planning and Scheduling*, Breckenridge, CO, April 2000.
- [5] R. Simmons and D. Apfelbaum, "A Task Description Language for Robot Control," *Proceedings of the Intelligent Robots and Systems Conference*, Vancouver, CA, October 1998.
- [6] I.A. Nenas, A. Wright, M. Bajracharya, R. Simmons, T. Estlin, Won Soo Kim, "[CLARAty: An Architecture for Reusable Robotic Software](#)," SPIE Aerosense Conference, Orlando, Florida, April 2003.
- [7] J. Bresina, K. Golden, D. Smith, and R. Washington, "Increased Flexibility and Robustness of Mars Rovers," *Proceedings of the International Symposium, on AI, Robotics and Automation for Space*, Noordwijk, The Netherlands, June 1999.
- [8] R. Alami, R. Chautila, S. Fleury, M. Ghallab, and F. Ingrand, "An Architecture for Autonomy," *International Journal of Robotics Research*, 17(4) April, 1998.
- [9] E. Gat, "Integrating planning and reacting in a heterogeneous asynchronous architecture for mobile robots," in *SIGART Bulletin* 2, 1991.
- [10] R. Bonasso, R. Firby, E. Gat, D. Kortenkamp, D. Miller, and M. Slack, "Experiences with an Architecture for Intelligent, Reactive Agents," *Journal of Experimental and Theoretical Artificial Intelligence Research*, 9(1), 1997.
- [11] S. Chien, R. Sherwood, et al., "Using Autonomy Flight Software To Improve Science Return on Earth Observing One," *Journal of Aerospace Computing, Information, and Communication* 2005.
- [12] A. Jonsson, P. Morris, N. Muscettola, K. Rajan, and B. Smith, "Planning in Interplanetary Space: Theory and Practice," *Proceedings of the Fifth International Conference on Artificial Intelligence Planning Systems*, Breckenridge, CO, April 2000.
- [13] D. Gaines, T. Estlin and C. Chouinard, "Spatial Coverage Planning and Optimization for a Planetary Exploration Rover," *Proceedings of the Fifth International Workshop on Planning and Scheduling for Space*, Baltimore, MD, Oct 2006.