

# INTELLIGENT ROVER DECISION-MAKING IN RESPONSE TO EXOGENOUS EVENTS

C. Chouinard<sup>(1)</sup>, T. Estlin<sup>(1)</sup>, D. Gaines<sup>(1)</sup>, F. Fisher<sup>(1)</sup>

<sup>(1)</sup>Jet Propulsion Laboratory  
California Institute of Technology  
4800 Oak Grove Dr.  
Pasadena, CA 91109-8099  
{firstname.lastname}@jpl.nasa.gov

**Abstract:** *This paper presents an introduction to the CLEAR (Closed Loop Execution for Autonomous Rovers) system which performs rover command generation and re-planning due to both inconsistencies with time and resource estimations and opportunities encountered during execution. CLEAR also makes use of onboard feature extraction and data analysis tools and is able to dynamically adjust the rover's current plan to achieve opportunistic science. The challenges faced maintaining domain specific information (e.g., science priorities) in an uncertain environment are also presented along with the successes demonstrated with several methods of system testing.*

## 1. INTRODUCTION

The command sequences used by the first Martian exploratory rover to traverse the unexplored terrain were limited in flexibility and resulted in large amounts of intermittent rover idle time. The rover traveled slowly and cautiously only a few meters each day at best. Today the twin Mars Exploration Rovers (MER) have traveled many kilometers across the Martian terrain using autonomous navigation [1]. As navigation autonomy has enabled rovers to accomplish more science than previously possible, decision-making autonomy can also enable a rover to increase its science by making dynamic decisions in an unpredictable environment and by capturing new, scientifically interesting data while maintaining primary objectives.

A spacecraft mission to Mars typically begins each day with a team of scientists and engineers deciding what will need to be accomplished in the next day on Mars and generating a command sequence for the rover to follow. A rover may be required to visit several goal targets, which are pre-determined scientifically interesting, and perform some science action at each target. The rover may take an image of an unusual rock, or use the rock abrasion tool (RAT) to clean or grind away the surface of a rock, or take a spectrometer read of the soil over a possible ancient sea bed [2].

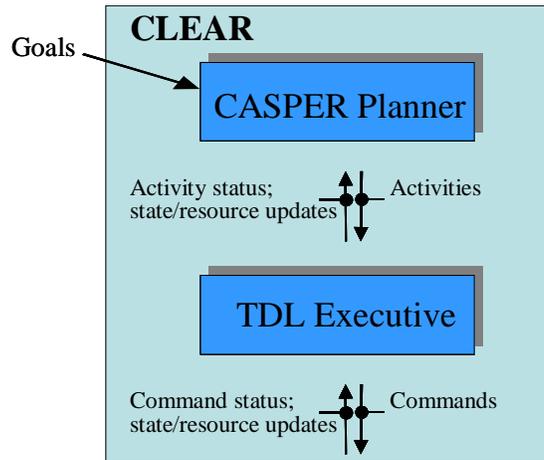
The set of science goals and the traverses to each target,

along with all other daily activities, make up the rover's daily plan (or command sequence). Each activity in the plan can only be an estimation of what is expected to occur, either taking an optimistic or pessimistic view of operations. If it is expected that the environmental conditions will be favorable to the rover, that all traverses and sciences will finish quickly or on time, then the plan can be optimistically tightly packed with very little buffer for activities to start and finish and more activities can be accomplished in the day.

However, the terrain may be more difficult to drive across than expected and images may not be compressed as much as estimated, so activities can often take longer than expected or may use more resources than originally estimated. One oversubscription of time or resources in a plan can have a rippling effect on the future activities in the plan. Thus a pessimistic activity model of estimation can be used which can protect against problems by allowing activities to have plenty of buffer within and surrounding them. However, a sparse initial plan may result.

Without onboard planning and decision-making capabilities, the daily plan must be generated using pessimistic estimations for both activity durations and resource usage to reduce the chance that conflicts will occur when the plan is executed. The CLEAR (Closed Loop Execution for Autonomous Rovers) system attempts a reasonably optimistic initial plan, while monitoring time and resources to react to oversubscriptions. It continuously looks for opportunities to improve the plan by adding-in activities that were originally impossible to achieve based on expected time and resource usage and also by adding-in opportunistic science activities. This continuous adjustment of the plan due to problems or opportunities allows CLEAR to use both optimistic and pessimistic reasoning strategies.

The CLEAR system combines planning and execution techniques to autonomously generate and adjust rover command sequences in order to achieve science goals. The interaction between the planner and executive



## Real or Simulated Rover Control Software and Hardware

Fig. 1: CLEAR Planning and Execution System

allows CLEAR to share knowledge between planning and executive processing, determine local versus global plan changes, and anticipate a large class of problems. CLEAR differs from MER (Mars Exploration Rovers) in that it is an onboard decision-making system that has dynamic plan modification capabilities while MER uses a ground team to manually create the rover's daily plan and adjust the following day's plan due to new information received each day.

In addition to the initial set of science goals in a plan, there may be new data that triggers a "science alert". Science alerts are potentially new goals that are discovered while traversing between two science targets. [3] An onboard data analysis tool alerts CLEAR that an opportunity exists to gather valuable science data, and the planner attempts to fit this new goal into the plan without disrupting the current state. The following section describes the whole system and details how it achieves the balance of optimistic and pessimistic operations.

### 2. SYSTEM DESCRIPTION

The CLEAR system generates an initial plan for a rover, manages state and resource constraints over time, monitors plan execution progress, and continuously searches for opportunities to improve the current plan. It has been expanded from a previous version to handle science alerts that are detected through the use of onboard feature extraction and data analysis components from the OASIS (Onboard Autonomous Science Investigation System) system [4]. CLEAR fits into the OASIS structure by providing planning and execution capabilities.

CLEAR combines the Continuous Activity Scheduling, Planning, Execution, and Replanning (CASPER) tool [5] with an executive written in the Task Description Language (TDL) [6] to model activities and the rover world, schedule the activities, and monitor their progress over the given timeframe. (See Fig. 1.)

- The **CASPER** tool is used to model a spacecraft's resources and states while also defining domain constraints and hardware functionality. The continuous planner generates a sequence of tasks and monitors the status of executing tasks. If unexpected events occur, CASPER can react accordingly, as needed and as defined
- The **TDL executive** monitors specific task execution, including all related subtasks. It expands abstract tasks into lower-level commands, executes the commands, and monitors their execution. It also provides direct support for exception handling and fine-grained synchronization of subtasks.

In order to interact with low-level rover hardware and necessary control software, CLEAR is integrated with the Coupled Layered Architecture for Robotic Autonomy (CLARAty) [7], which is being developed at JPL in response to the need for a robotic control architecture that can support future mission autonomy requirements. CLARAty provides a large range of basic robotic functionality and simplifies the integration of new technologies on different robotic platforms. For this work CLARAty has provided software for obstacle avoidance, navigation, vision, locomotion, and rover pose estimation. Through CLARAty, the CLEAR

system has been tested with several JPL rover platforms, including Rocky 7, Rocky 8, and FIDO.

As shown in Fig. 1, the planner maintains the scheduled plan activities and sends them to the executive. The planner also listens to the executive for activity status and state and resource updates which may cause a need for plan adjustment. Conflicts that arise are resolved in the planner and the new plan is related to the executive. The executive receives the activity from the planner and translates it into the appropriate rover commands, often breaking up one activity into several smaller tasks which will be monitored for progress and completion. The executive receives command status updates as well as state and resource information and can attempt to resolve local problems with smaller tasks, but will notify the planner when the task cannot be accomplished and a global, planner-level fix is required. The idea of separate control allows the planner to work on optimizing the future state of the plan while the executive monitors the progress of the current state. When there is a conflict with the previously planned activities' progress, the global planner works on a resolution [8], [9].

## 2.1 Modeling the World

The CLEAR system uses CASPER and TDL to model the rover world, including its basic environment, its state and resource constraints, and the activities it will be expected to perform. The CASPER model includes information on the environment, such as the timeframe of each plan, the types of instruments available on the rover, as well as information on rover constraints which may detail the minimum and maximum usage of onboard resources or transitions from possible execution states to renewal states at particular times of day.

Each activity model estimates the predicted amount of resources and time which will be used, while also stating which hardware components will be required while performing the science. Some components may be atomic and cannot be used by two activities at the same time, while others are aggregate and may be used by a limited number of activities at any given time.

The TDL model also contains information about the rover and its environment. This includes constraint information, such as what preconditions must be true for an activity to begin execution, ways to decompose activities into commands based on current state, and exception handlers for some situations.

## 2.2 Initial Plan Generation using DFBnB

In our scenarios, and typically in a rover's day, there are several different science goals to accomplish. These targets often have a certain priority associated with them, determined by the scientists. There are several methods of deciding which goals to include in the plan, based on priority and other cost function parameters such as distance between targets and the sun angle at certain times of day. In our current system, we use a "strict priority" method as a model for generating the initial plan and the type of action to take when the plan must change. A strict priority method states that higher priority goals are always more desirable than lower priority goals, no matter how many lower priority targets could be included in a plan for even one higher priority target. Other rules or methodologies could also easily be adopted and have been used for previous tests.

To guarantee an optimal initial plan, based on our own specified criteria, CLEAR uses Depth First Branch and Bound (DFBnB) to order the set of science goals. To respect the strict priority rule, plans with the most number of high priority targets are scored the highest. Bounding occurs when the priorities of the remaining goals to be added to the search tree's "branch" are not as high as the priorities of the goals of the best plan found so far, when the accumulated distance cost of the "branch" is higher than the best plan found, and when the current "branch" oversubscribes time and/or resources. The result is a conflict-free plan with the highest priority targets included. Target ordering uses the shortest distance, fits in a limited plan timeframe, and uses only the amount of resources that are initially allowed. The DFBnB algorithm in CLEAR has been adjusted for many different bounding criteria over its development process. Temporal constraints on specific goals (which will be discussed in the future work section) are also being added to the DFBnB algorithm.

## 2.3 Conflict and Goal Management during Plan Execution

Conflicts in the plan are inconsistencies with the planner's model of activities and resources. Conflicts can occur when the planned state of the rover is updated with new information which is inconsistent with the expected state. For example, a conflict may arise if the energy used by a RAT operation is greater than modeled, or if the rover stops during its traverse before reaching the intended position due to an unforeseen obstacle. The first conflict is with resource usage and the later is with expected orientation. Conflicts must be eliminated from the plan immediately. The CLEAR system does this by selecting an appropriate repair strategy (adding, moving, and/or deleting an activity or

a series of activities). The decisions made are based on the conflict types and the built-in heuristics.

When a deletion of a science goal is required due to unforeseen events, the goal's status is changed, but it is not permanently deleted from the plan. The goal becomes "requested". Goals that could not be initially included in the plan due to time or resource limitations are also in this state. When favorable events occur which free up resources and time, these goals may be added back to the plan through the use of the continuous optimization procedure. CLEAR is always trying to improve the score of its current plan by adding-back science goals or adding-in newly discovered goals. As each activity in the plan finishes, the state of the plan may be a little more or a little less accommodating to new goals. To ensure conflicts on hard constraints are always resolved, an unconflicted plan's score is always better than a conflicted plan. The optimization cycle works to repair conflicts as they occur and improve the plan whenever possible. Continuous optimization makes most efficient use of the rover's varying resources and uncertain environment.

#### 2.4 Handling Science Alerts



Fig. 2: Sample image that was taken in response to a science alert on the JPL FIDO rover

Science alerts are new goals discovered while navigating between the already planned science targets. Science alerts can either be fit into the current plan or the rover can be stopped and instructed to wait for ground instructions in the case of very significant discoveries. We limit discussion to the former type. Onboard data analysis uses the rover's camera images to detect interesting rocks. When an interesting rock is discovered, an alert is sent to the planner. The planner then attempts to add-in the appropriate traverse and image activities needed to achieve the opportunistic

science while maintaining the state of the existing plan. See Fig. 2 for an example of an image taken in response to a science alert from a detected rock. In this example, the analysis system was set to detect rocks of light albedo.

In our current system implementation, we have specified that science alerts must have a priority lower than any initial (or ground-specified) science target. Science alerts are treated differently, because they are considered supplemental science. Further, science alerts must be handled in a short timeframe. If they are not planned for quickly, the rover could move far past them, creating a more difficult problem to solve. Also, the current constraints and state of the plan may make adding-in the science alert infeasible. For this reason, science alerts have an expiration time. If the planner cannot add the science alert's activities into the tightly-packed plan quickly, the alert is removed from the list of potential goals. The rover has its objectives and will inform the scientists of these alerts, if desired, but the autonomous system only re-plans the current state to achieve these novel goals if it can do so quickly and easily. If they are unachievable, CLEAR allows the continuous optimization cycle to work on other types of plan improvement.

#### 2.5 Handling Resource/Time Oversubscriptions

Time oversubscriptions are handled by either the executive or the planner. The executive receives data directly from the rover and passes it to the planner. For some activities, such as traverses, the executive can monitor the activity's progress through state or resource updates. For instance, since the distance between any two goals is known, the distance remaining in any given traverse activity can be calculated based on the current position. In the CLEAR system, we can also define a percentage of tolerable progress. If the rover is not making at least ninety percent progress, the system may decide to give up on the current drive and maybe even the target goal. The executive attempts to resolve conflicts with a traverse activity until either the goal has been reached or the tolerable progress has not been achieved. In the later case, the planner must resolve the problem. The planner dynamically adds a new traverse activity to the plan if there is time, or makes a decision to rearrange activities based on any new terrain knowledge, or may have to delete a low priority target to accommodate the requirement to achieve all higher priority goals.

Oversubscriptions of resources can also trigger the need for repair. A dig command may use more energy than

estimated due to tougher soil. If the plan was generated with an optimistic view of operations, this single resource oversubscription could cause problems for the future activities in the plan, and a decision must be made as to what modifications will need to be made. Our current implementation would respect the strict priority rule in the case of a required deletion and adjust the plan accordingly. CASPER then handles any oversubscriptions from the activity's execution.

## 2.5 Handling Resource/Time Undersubscriptions

The planner may also receive data from the rover which allows an opportunity to beneficially alter the plan. If the initial plan was not able to include all the desired goals due to resource or time constraints, the rover may make up enough time or resources throughout the day to allow one or more goals to be added into the plan. This situation occurs often when either the planner's model overestimates the average time or resources the rover will use to accomplish each activity or the lower priority goals to be added into the plan are relatively close to the existing planned path and only a small deviation is needed to drive to the new goal. Adding-back goals to the plan allows the system to be somewhat pessimistic at the start, but to potentially achieve the goals that would have been planned for with an initially optimistic view. Similarly to the dynamically added science goals, science alerts are also added into the rover's plan if the conditions are favorable.

## 3. SYSTEM TESTING

The CLEAR system has been tested both in simulation and on real hardware. Each testing method presents its own advantages and different levels of ability. In simulation, control over the scenario events and the rover behavior allows very rigorous testing of complex problems and helps prepare CLEAR for real rover hardware testing. Running CLEAR on the research rovers in JPL's Mars Yard introduces the unpredictability of natural elements into the testing process.

### 3.1 Simulated Testing

To test CLEAR in simulation, first a set of "random" scenarios and events was generated. Each scenario consisted of several goals in random locations and a controlled variation of the modeled world parameters including rover speed, resource depletion rates, and timing data. The random "events" included in the test cases were a list of science alerts which were simulated to arrive from the onboard data analysis tool. Other

events included sudden and gradual drops in resource usage.

A generic testing framework runs each scenario automatically and gathers logging information. Statistics, including timing information for generating the initial plan, for adding-back goals through optimization, and for satisfying new science alert goals, are automatically collected from the logs after each scenario completes. These simple statistics help identify problems within the separate components of CLEAR.

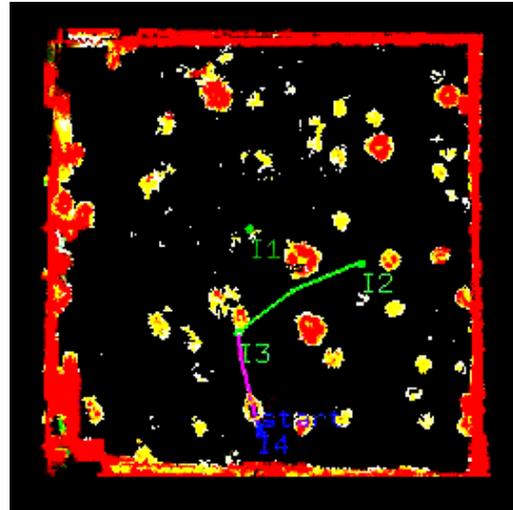


Fig. 3 : GriViT Visualizing a Simulated Rover

The Grid Visualization Tool (GriViT), shown in Fig. 3, captures images of the rover's progress over the entire execution timeframe. Planned paths and goals are indicated in green, blue represents the past, and magenta paths are currently executing. Obstacles and their severity are indicated in varying colors along the red, yellow and green spectrum where red and yellow should be avoided, and while green is not "free" space, it also does not present a danger to the rover, but may be used by the navigation algorithm to calculate the best possible next step. The sequential image files are later combined to create a visual movie of the rover moving along its planned path. Each image displays goals and path information, movement of the rover along the path, and the receipt and potential addition of science alerts. For debugging purposes, GriViT can also display conflict information.

Because it is so easy to run and analyze large numbers of tests, simulated tests are also very good at finding rare errors which occur only once in every 25 to 50 tests. In simulation there is no need for daylight, nor dry weather, nor is there a need to monitor the rover for



Fig. 4: Rocky 8 rover (left), FIDO rover (middle), Rocky 7 rover (right)

possible hardware issues. Conditions are always ideal. It is not possible to test every scenario and event, but it is easy to manipulate the random parameters to create general cases and solve problems within the CLEAR system. It is important to log the appropriate data for each run, since it is sometimes difficult to recreate the same problems found in one particular scenario. Understanding what could happen is as important as understanding what has happened.

### 3.2 Hardware Testing

CLEAR has been extensively tested and formally demonstrated on the Fido, Rocky7, and Rocky8 (see Fig. 4) rovers in JPL's Mars Yard over the past few years. Many tests have included over 40 meter traverses with numerous science goals (up to 13) and often with as many science alerts. Some tests have run several hours long. In the essence of time, the latest demonstration in January 2005 was limited to 30 minutes. CLEAR modeled the rover world with several goals to start with, but with a time limitation which would not allow all the goals to be included in the initial plan. Several science alerts were detected and successfully handled while driving between targets, and the rover made up enough time during its traverses that the continuous optimization procedure was able to add-back the low-priority goal that was originally not included in the plan. Fig. 5 displays an example of this scenario in simulation using GriViT. The blue path indicates the rover navigating its way between waypoints and goals. The real rovers use their cameras and an obstacle avoidance algorithm to best decide their path between points. The planner and executive command the rover to the major waypoints, not every step in between, hence the difference between the planned path and the actual path traversed.

## 4. RELATED WORK

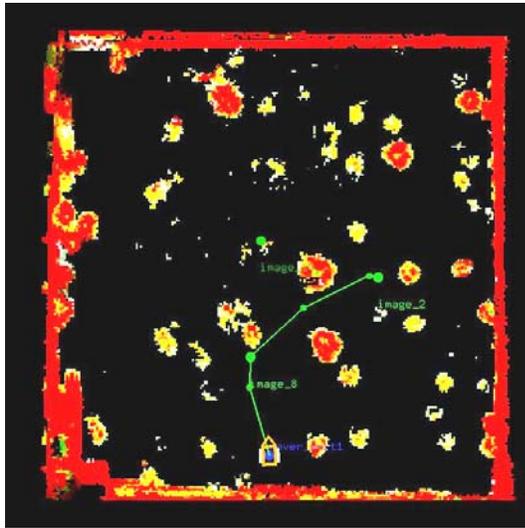
A number of planning and executive systems have been successfully used for robotic applications and have

similarities to the approach described in this paper. Most of these approaches have used some combination of planning and execution, however they differ in not only the behavior of these individual components, but also in how these systems interface with each other and with other system modules.

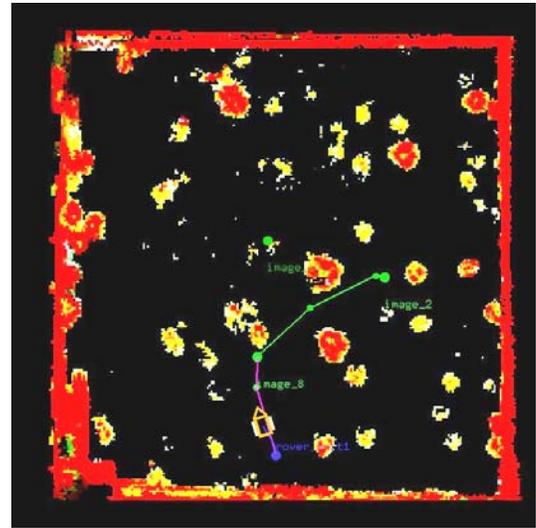
The Autonomous Sciencecraft Experiment (ASE) [10] has demonstrated the capability of planning and data analysis systems to autonomously coordinate behavior of the EO-1 Earth orbiting satellite. ASE can also detect and respond to new science events; however it uses very different detection and analysis algorithms. The Remote Agent Experiment (RAX) [11] was flown on the NASA Deep Space One (DS1) mission. It demonstrated the ability of an AI planning, execution and diagnosis system to respond to high-level spacecraft goals by generating and executing plans onboard the spacecraft. However, RAX did not incorporate data analysis to identify new science targets and used a batch approach to planning. Furthermore, since RAX and ASE were applied to spacecraft, neither handles issues associated with the uncertainty of surface navigation.

Another approach directed towards rover command generation uses a Contingent Planner/Scheduler (CPS) that was developed to schedule rover-scientific operations using a Contingent Rover Language (CRL) [12]. CRL allows both temporal flexibility and contingency branches in rover command sequences. Contingent sequences are produced by the CPS planner and then are interpreted by an executive, which executes the final plan by choosing sequence branches based on current rover conditions. In this approach, only the executive is onboard the rover; planning is intended to be a ground-based operation. Since only a limited number of contingencies can be anticipated, our approach provides more onboard flexibility to new situations. In the CRL approach, if a situation occurs onboard for which there is not a pre-planned contingency, the rover must be halted to wait for communication with ground.

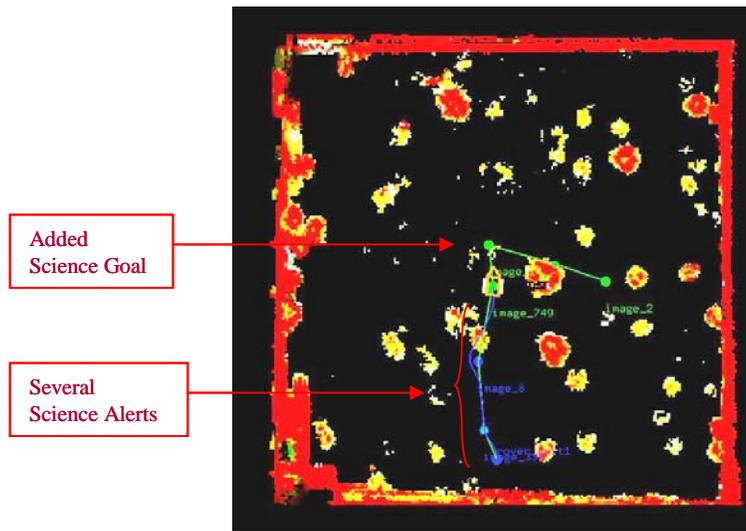
## 5. FUTURE WORK



A. Initial Plan, Limited Resources



B. Rover Navigating between Goals, Searching for Opportunistic Science



C. New Goals Added into Plan, Optimization Adds-Back Science Goal

Fig. 4 : Formal Demonstration on FIDO Rover

There have been a number of exciting additions to CLEAR recently which have yet to be tested on the real rovers. Previously, all science alerts kept the rover in the same physical location, only turning to face the new target. CLEAR now has the added capability to change the planned path and actually plan for the rover to drive closer to the new target to better accommodate the data collection at the new goal. These drive-to science alerts introduce new problems, including where to insert them into the planned path, since the current time might not be the best position for them. CLEAR inserts them into their most optimal ordering based on shortest path distance, in most cases. If the new goal is close enough to an already scheduled goal, CLEAR may use the same

location for both goals, and then the higher priority goal will be achieved first.

Temporal constraints on goals have also been added to both the initial plan generation and the actual execution of the plan. Strict priority is still respected, where the high priority target must be included in the plan before any other goals. However, once execution of the plan begins, if the temporally constrained, high priority goal has passed its latest possible start time, CLEAR deletes the goal from the list of possible goals and allows other, lower priority goals to be added into the plan through the optimization process, if possible. An expansion of this idea is to relax the strict priority algorithm

specifically for plans with temporally constrained activities. If the highest priority target not included in the current plan is constrained to start at 14:00 and the current time is 10:00, we may want to try to add other lower priority targets that could fit in the plan now, rather than wait for the opportunity to add-in the high priority target. With this expansion comes the need to define a timing tolerance for how close in time to the higher priority target we can be to allow this relaxed behavior.

## 6. CONCLUSIONS

CLEAR has successfully demonstrated autonomous planning, scheduling and execution capabilities for rovers in both a simulated world and a Martian mock-up terrain facility here at JPL. The onboard decision-making system generates an optimal initial plan, respecting a strict priority ideal; executes activities and monitors their progress; re-plans when state and resource estimations are under and over-subscribed; and robustly handles multiple types of opportunistic science. A balance between optimistic and pessimistic operations allows CLEAR to take advantage of all opportunities and to react appropriately to an uncertain environment, thus achieving a greater amount of science data when possible, and making most efficient use of rover resources.

## 7. REFERENCES

1. Mars Exploration Rover Mission, Jet Propulsion Laboratory. (March 02, 2005). *Mars Exploration Rover Mission: Technology*. Retrieved June 17, 2005, from [http://marsrovers.jpl.nasa.gov/technology/is\\_autonomou\\_s\\_mobility.html](http://marsrovers.jpl.nasa.gov/technology/is_autonomou_s_mobility.html)
2. Mars Exploration Rover Mission, Jet Propulsion Laboratory. (June 13, 2005). *Mars Exploration Rover Mission: The Mission*. Retrieved June 17, 2005, from [http://marsrovers.jpl.nasa.gov/mission/spacecraft\\_surface\\_instru.html](http://marsrovers.jpl.nasa.gov/mission/spacecraft_surface_instru.html)
3. R. Castano, M. Judd, T. Estlin, R. Anderson, D. Gaines, A. Castano, B. Bornstein, T. Stough, and K. Wagstaff. "Current Results from a Rover Science Data Analysis System," *Proceedings of the 2005 IEEE Aerospace Conference*, March 2005.
4. R. Castano, M. Judd, T. Estlin, R. C. Anderson, L. Scharenbroich, L. Song, D. Gaines, F. Fisher, D. Mazzoni, and A. Castano, "Autonomous Onboard Traverse Science System," *Proceedings of the 2004 IEEE Aerospace Conference*. Big Sky, Montana, March 2004.

5. 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 International Conference on Artificial Intelligence Planning and Scheduling*, Breckenridge, CO, April 2000.

6. R. Simmons and D. Apfelbaum, "A Task Description Language for Robot Control," *Proceedings of the Intelligent Robots and Systems Conference*, Vancouver, CA, October 1998.

7. I.A. Nesnas, 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.

8. T. Estlin, D. Gaines, C. Chouinard, F. Fisher, R. Castano, M. Judd, R. Anderson and, I. Nesnas, "Continuous Planning and Execution for an Autonomous Rover," *Proceedings of the 2005 IEEE Aerospace Conference*, Big Sky, Montana, March 2005.

9. F. Fisher, T. Estlin, D. Gaines, S. Schaffer, C. Chouinard, R. Knight, "CLEaR: Closed Loop Execution and Recovery – A Framework for Unified Planning and Execution," *Technology and Science IND News Issue 16*, pp. 15-20, September 2002.

10. S. Chien, R. Sherwood, D. Tran, B. Cichy, G. Rabideau, R. Castano, A. Davies, D. Mandel, S. Frye, B. Trout, S. Shulman, D. Boyer, *Using Autonomy Flight Software to Improve Science Return on Earth Observing One*. Journal of Aerospace Computing, Information, and Communication, April 2005.

11. N. Muscettola, P. P. Nayak, B. Pell, and B. C. Williams, *Remote Agent: To Boldly Go Where No AI System Has Gone Before*. Artificial Intelligence, 103(1-2):5--47, 1998.

12. 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. ACKNOWLEDGEMENTS

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