Data Visualization for Effective Rover Sequencing

Frank R. Hartman
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA, USA
Frank.Hartman@jpl.nasa.gov

Brian Cooper, Chris Leger,
Scott Maxwell, John Wright and Jeng Yen
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA, USA
Brian.Cooper, Chris.Leger, Scott.Maxwell,
John.Wright, Jeng.Yen @jpl.nasa.gov

Abstract - *The Rover Sequencing and Visualization Program is a suite of tools for the commanding of planetary rovers which are subject to significant light time delay and thus are unsuitable for tele-operation. The two main components of the program are the Rover Sequence Editor and HyperDrive. This paper will focus on HyperDrive, the immersive visualization component of the system. HyperDrive fuses multiple data types returned from the vehicle in order to facilitate an operator understanding of the current environment and past rover performance, so that safe effective command sequences for successful future rover activities may be generated on a tight tactical timeline. Multiple display and task specific interaction modalities are provided to most efficiently present relevant spatial and time series data to the sequence builder.*

Keywords: Spacecraft commanding, surface operations, planetary rover, sequence visualization, data fusion, augmented reality.

1 Introduction

The Rover Sequencing and Visualization Program (RSVP) is the current generation of rover commanding tools in use by the Mars Exploration Rover mission at the Jet Propulsion Laboratory. RSVP is based on the Rover Control Workstation which was developed and used during the Mars Pathfinder mission [1],[2] and on several visualization technology development efforts conducted during and after the Pathfinder mission [3],[4].

RSVP is a distributed heterogeneous collection of applications using the Parallel Virtual Machine (PVM) API to communicate between themselves. The Rover Sequence Editor (RoSE) is the master task and is charged with maintaining the master copy of the sequence under development and distributing that sequence to the other components so that they may provide their own display modes and methods of modifying the sequence. This paper will primarily discuss HyperDrive, which is the component tasked with providing a three-dimensional display for sequence items that have a strong spatial component such as arm (instrument deployment device or IDD) and mobility commands [5]. We will use the term RSVP when referring to the entire application and the term HyperDrive when referring specifically to the visualization component of it. All sequence builders on the MER mission use RSVP with the HyperDrive component being used primarily by the rover planners who are charged with generating all arm and motion sequences with some additional use in the generation of imaging commands.

HyperDrive is based on providing several immersive views into a unified three-dimensional database. This database primarily consists of CAD models of the MER rover with instrument deployment device, linearized image data, three-dimensional terrain models derived from stereo image data, a spatial representation of the command sequence under development, and time ordered histories of rover state values which may originate from received telemetry or ground based kinematic simulation. HyperDrive also maintains interfaces with ground versions of the flight software as run on the vehicle. This allows the simulation to provide the highest fidelity available in determining how the on-board IDD or mobility software will respond to the commands being issued in the current environment. By building a unified model of the rover and its environment and modeling our data classes to reflect real world interactions, the process of adding new data or interfaces has been greatly simplified.

2 Data Sources and Interfaces

There are several main data types that are essential elements of the HyperDrive visualizations. In our operations process there are tight delivery time and accuracy requirements on many of these products and a separate subsystem with responsibility for each.

2.1 Terrain

Three-dimensional terrain meshes are one of the main sources of data to the HyperDrive application. The first piece of data that a rover sequence builder will load will generally be the terrain mesh. The terrain mesh will provide instant context for the rover once its current kinematic state has been applied to the CAD model and in this way a much more rapid understanding of the rover's pose and potential interactions with the surrounding landscape can be gained than by viewing the image data alone. Generation of terrain meshes from imagery is made possible by the uses of stereo cameras on the rover and is covered in greater detail in [6]. Generally, the terrain meshes are texture mapped with the camera imagery used to generate them, but other colorations are possible to

show various attributes of the terrain to the user. One example of this, coloration based on traversability, will be discussed in the autonav section of this paper.

2.2 CAD Data

One of the fundamental features of the HyperDrive visualizations is the placement of an accurate rover CAD model into the correct position on the terrain mesh. This is the primary method by which the rover planner can achieve an awareness of the rover's situation in relation to its environment. This is the primary visual cue provided and all other visualizations refer to this context. CAD data of the rover is not delivered on a daily basis but is subject to strict requirements on accuracy particularly when planning arm operations where very tight clearances are possible. The model of the rover used in the visualizations is necessarily a simplification of the CAD model used for manufacturing and mechanical design due to graphics performance issues but during its creation the essential kinematic features and outer envelopes of parts were maintained in order to assure that the visualization would not indicate sufficient clearance when in fact there was an interference problem with a given motion sequence.

2.3 Commands

The command sequence under construction is of primary importance to the rover planner. The master copy of a sequence is maintained by the rover sequence editor and transmitted to HyperDrive in an xml dialect called Rover Markup Language (RML). After parsing the RML command sequence HyperDrive generates three-dimensional glyphs for mobility and arm commands and places them in the spatial location where they will execute. HyperDrive thus provides a three-dimensional visual programming language for motion of the Mars Exploration Rovers. This command sequence can then be simulated with varying degrees of fidelity to understand its likely performance on Mars and to check for any possible error conditions.

2.4 Rover State Information

Rover state information is represented within HyperDrive as a state vector class and a state history class that is a linearly interpolated dense time history of state vectors. Of primary importance are the kinematic parameters of the vehicle but the classes are sufficiently general to represent any data type that it is desired to track, including software parameters, actuator current histories, etc. A primary design decision was to represent information telemetered from the vehicle and simulation results by the same classes and to define a common xml specification (RKSML) to use for their file I/O. This decision has allowed us to use common modes for displaying and querying these state information datasets and to present a unified interface for assessing both past vehicle performance and future, simulated performance. We can then compare our simulation to the results of the command sequence execution in order to refine the simulation's accuracy [7].

2.5 Ephemeredes

Another data source that is incorporated into the visualization are the ephemeris kernels provided by JPL's NAIF/Spice system. These data are used to correctly locate the sun, earth, and orbiting relay spacecraft in relation to the terrain and rover and to perform conversions between the multiple time systems in use by the mission. A primary use for this information is in correctly positioning the sun in order to do lighting or shadowing predictions as shown in Figure 3. These shadow calculations have been very well received by the science team, particularly for predicting lighting condition for the Microscopic Imager.

3 Display Modes

HyperDrive provides three main viewing modes into it's unified three-dimensional database. These modes are designed to provide the operator with a high level of comprehension of the rover's state and its current environment and to allow rapid assessment of potential hazards and other terrain features that will impact the structure of commanded rover activities.

The central mode that is displayed inside the main GUI is the "God's Eye" or "Flying Camera" view. The main feature of this view is that it allows the operator to view the rover and terrain from any arbitrary viewing position. This view is essential to gain an understanding of the rover in its terrain and to break out of the fixed outward-looking imagery that is returned by the rover. Figure 1 shows an overhead view of a planned traverse in Eagle crater that would be impossible to create solely using the raw imagery returned from the rover's (then) current viewpoint.

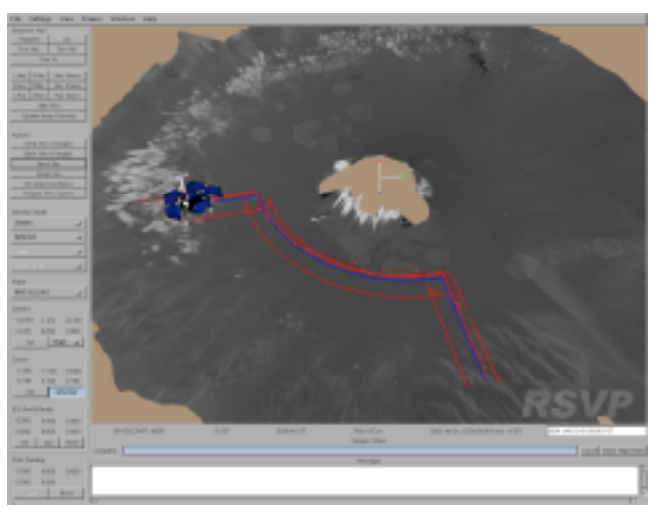

Figure 1. Flying Camera View

There are limitations to the performance of the stereo processing algorithms that are used to create the terrain meshes including but not confined to the limited range of the stereo correlation process. The "augmented reality" or CARD (Computer Aided Remote Driving) view is designed to provide a viewing mode that overcomes some

of these limitations. The CARD view is based on the linearized image data, which is then overlaid with rendered imagery from the three-dimensional database excluding the terrain mesh. In this way the CARD view is always synchronized to the other application views but provides superior detail and access to the raw image data. This view is primarily utilized with liquid crystal stereo viewing and it is this mode that is most useful for planning long distance traverses and allowing the rover planners to gain a much more intuitive kinesthetic understanding of the terrain's contours.

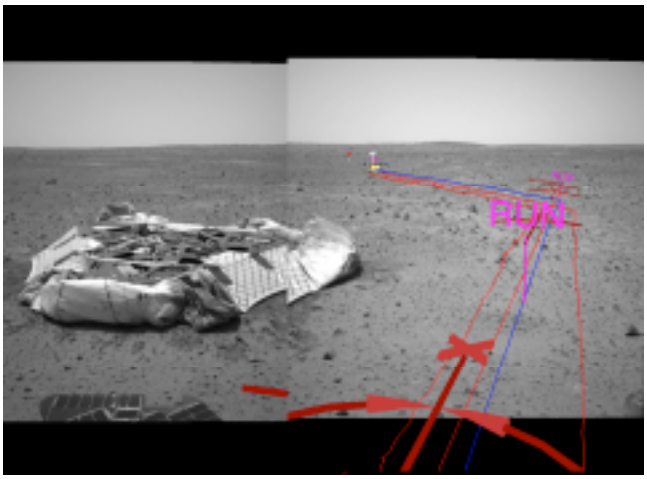

Figure 2. Augmented Reality View

A third type of view provided by HyperDrive is the simulated camera view. This view is based on the same database but is rendered from the viewpoint of one of the nine cameras located on the vehicle [8]. The main purpose of this view is in predicting the image data to be returned by a given sequence. For example Figure 3 shows a fully simulated view looking out one of the front Hazcam's towards a commanded arm position. The shadows are correctly calculated for the time of day that the arm motion sequence was to be run.

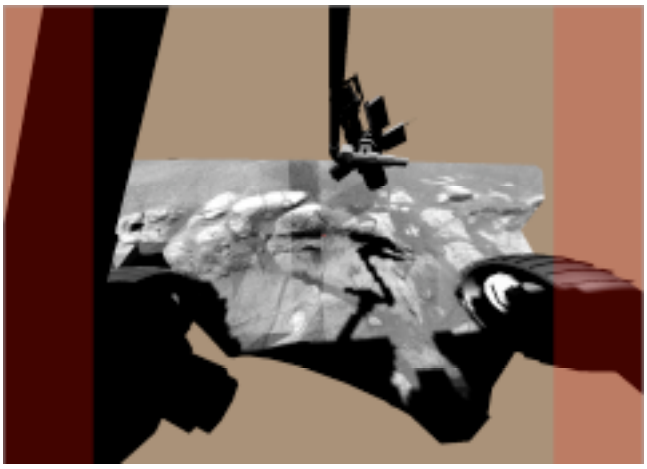

Figure 3. Simulated Front Hazcam view with shadowing

4 Task Specific Modes and Tools

4.1 General Issues

In order to build any type of mobility or arm sequence there is a minimal set of data that must be loaded into HyperDrive to establish the scene and the state of the rover within it. This will allow the multiple coordinate frames that may be used for commanding to be correctly instantiated so that the transforms between them needed to correctly designate may be carried out. The most important piece of information is the state vector that defines the initial condition for the current sequence build. This is absolutely critical to generate the correct designation for the various coordinates (in multiple frames) that will go into an arm or motion sequence. Once the state of the rover is established the next step is to load a terrain mesh to provide context for the current state and the sol's planned sequence. Finally, science targets defining the requested arm placement or in some cases mobility position are loaded. These targets are generated by the science team upstream of the sequence build process using the Science Activity Planner [9]. Once this basic set of data are loaded, the rover planner will continue on to build any one of the sequence types discussed in the following paragraphs.

4.2 Low-Level Driving

A low-level drive sequence consists of a series of driving primitives that are in effect "blind." This type of sequence will generally be constructed from ARC and the various flavors of TURN commands. The TURN commands do servo to an azimuth based on IMU data but otherwise these commands simply turn the wheels the required numbers of times to achieve the commanded distance under perfect conditions. Imagery may be acquired during the traverse, but no image-based position update or hazard detection is processed during the execution of a low-level drive.

When building this type of drive the sequencer can be considered to be drawing geometric figures on the terrain using the command sequence. Having a good visual representation of the terrain to be traversed is essential in this mode both for the safety of the vehicle and to understand the likely slip environment that will modify the end state of the vehicle. Slip data were collected on a test fixture and provided a starting point for the rover planner to adjust the sequence based on the anticipated slip. Figure 4 shows the starting condition (Blue) and the ending condition of a planned low-level drive in Eagle crater. Also displayed are glyphs representing the STOW command and several calls to sub-sequences. Science targets for the previous sol are shown in yellow.

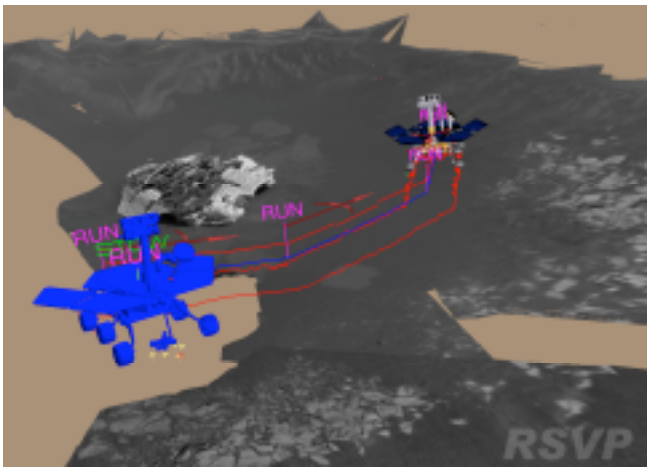

Figure 4. Mer-B Sol 21, Visually building a simple drive sequence. Blue rover represents initial conditions.

4.3 Autonav Driving

Autonav refers to the rover's internal autonomous navigation and path selection system [10]. Building this type of sequence shares many features of the the low-level drive type but also has some important differences. Designation and correct registration of all data is still critical in order to assure that the correct target coordinates are generated and included in the command sequence. Using autonav the behavior of the vehicle is no longer constrained to follow a specific path created by the rover planner. The vehicle is given a series of goal points and radial tolerances to those points and is free to respond to input from its imaging system in choosing how to traverse to those points. To facilitate building an autonomous drive sequence HyperDrive provides the sequence builder with a map of the terrain that is color-coded based on how the autonav terrain evaluation algorithm will assess it's traversability. Figure 5 shows an autonomous drive sequence built on a terrain color mapped from green to red in order of increasing mobility hazard. The dart glyphs represent the goal points for GOTO_WAYPOINT commands.

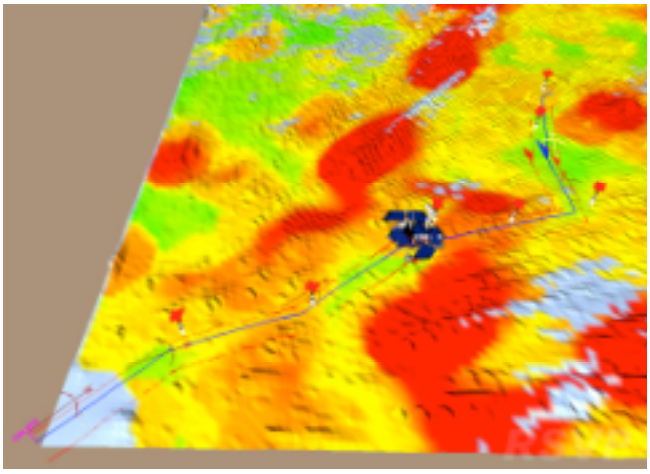

Figure 5. Mer-A Sol 63, Terrain coloration based on the Flight Software's evaluation of traversability

4.4 Arm Placement

Generation of arm placement sequences can be one of the most complex tasks in the sequence build process. Arm sequences can run into many hundreds of commands and ensuring a safe, error free command load is the rover planner's responsibility. HyperDrive provides an arm sequencing mode that has processed many thousands of commands with sub-centimeter end-to-end system placement accuracy. The essential feature of this mode is that a copy of the on-board flight software is invoked during every ground simulation of an arm sequence. This ensures that a sequence that will generate an error on-board will be caught during the sequence build process and corrected before upload.

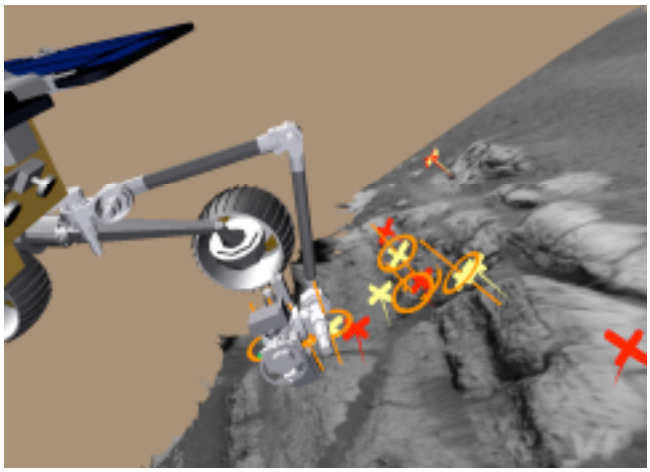

Figure 6. Mer-B Sol 28, View of an arm sequence under developement showing science targets (yellow and red) and command icons (orange)

An important part of HyperDrive's analysis of an arm sequence under development is to predict any collisions between the arm and itself or the arm and the vehicle that would be caused by execution of that sequence. This is accomplished be interfacing with the IDD flight software's collision checking functionality in order to exactly model the collision prediction that will be executed on-board the vehicle. The flight software's collision database can be displayed to the rover planner in order to facilitate his understanding of the collision constraints while developing an arm sequence. Figure 7 shows the arm position and collision volumes during a recent trench measurement. In order to retract the arm the desired amount, the left front wheel collision volume was disabled during the command sequence. The HyperDrive tool was used to verify whether there was sufficient space to do this and still maintain clearance between the arm and the vehicle. This would not be possible without a very high fidelity CAD-based model to use in our displays.

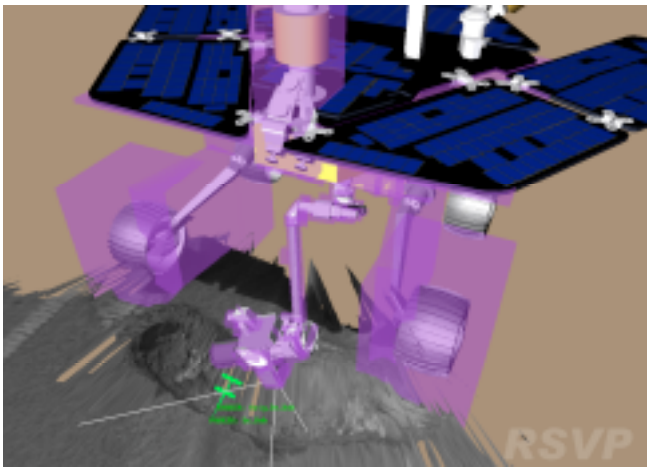

Figure 7. The violet polyhedra represent the flight softwares's onboard collision volumes.

4.5 Localization

Localization is the task of generating an improved estimate of the rover's position by using ground based processing of various kinds. HyperDrive provides for localization by manual alignment of two textured terrain meshes. A detailed analysis has not yet been performed comparing the accuracy of the different localization methods but operational experience indicates an estimation error of order 10 cm could be achieved using this method.

4.6 Terrain Analysis

The last, and most general, task the rover planners were called on to perform was analysis of the terrain's traversability and morphology. A plane regression tool is provided to allow for slope calculations and obstacle height measurements and a point measurement tool supports querying the terrain for position and normal information. An additional, and unexpected source of analytic capability came as a result of the flexibility of the flight software that HyperDrive interfaces with. Figure 8 was generated by configuring the flight software to generate a slope contour analysis of the terrain and compositing multiple contours.

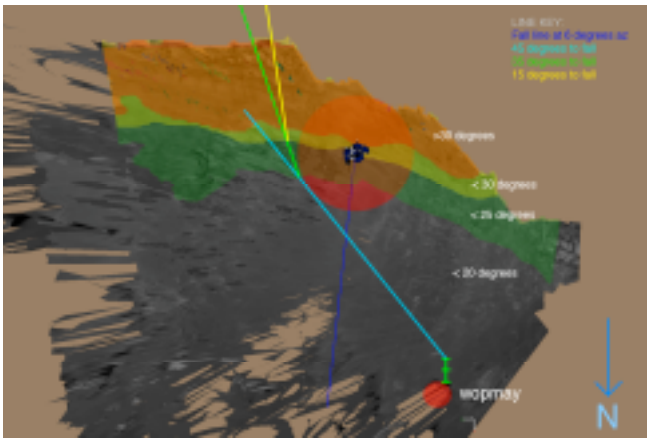

Figure 8. A presentation graphic prepared using RSVP and Photoshop for the Burns Cliff traversability discussion.

5 Results during surface operations

As of this date RSVP has been used to generate all sequences for a combined 780 sols on the surface of Mars. 538 of these sols included arm or mobility sequences generated by the rover planners using HyperDrive. The total traverse distance sequenced for Spirit was 4128.103 meters and for Opportunity 2559.88 meters. A total of 4,373 Microscopic Imager frames have been acquired, each requiring commanding of the arm to a distinct position. 1205 sequences have been produced and uploaded to the rovers containing a total of 68,071 commands. An important fact to remember in contemplating these numbers is the fact that these sequences were prepared in time frames measured in hours not days or weeks as is common for remote sensing missions. The science results are discussed elsewhere but we believe that the RSVP tool was essential in allowing the very high rate of science return by its role in generating accurate approaches to science targets and accurate placements of the arm instruments. Additionally, the HyperDrive generated visualizations were an important part of briefing the entire operations team during the upload process and allowing them to gain an intuitive visual grasp of a given sol's activities.

6 Conclusions

Visualization and data fusion techniques have been shown to have fruitful application during the surface operation of planetary rovers. The generation, on a daily basis, of multiple arm and mobility sequences would have been either extremely difficult or impossible using traditional text or image based commanding systems. Fusing all the relevant spacecraft data into a single three-dimensional database and providing interfaces to the vehicles' own flight software allowed a degree of accuracy and of flexibility that was invaluable in coping with the many unexpected challenges as the rovers traversed into unknown terrain on the surface of Mars. We plan to build on the many hours of experience and lessons learned during Mars surface operations to further enhance RSVP's capabilities for future in-situ missions.

Acknowledgments: The work described in this paper was performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. The Mars Exploration Rover drivers who used the RSVP tool and provided these screenshots are: Brian Cooper, Eric Baumgartner, Jeff Biesedecki, Bob Bonitz, Chris Leger, Scott Maxwell, John Wright, Ashitey Trebi-Ollennu, Mark Maimone, Khaled Ali, Jeng Yen, and myself. The interface to the rover autononav system was provided by Mark Maimone, rover autonomy lead programmer. Special thanks to Rudolf Bucus P.E.

References

[1] B. Cooper, “Driving on the Surface of Mars Using the Rover Control Workstation,” Proceedings of SpaceOps ’98, Tokyo, Japan, 1998.

[2] B. Green et al, “Processing and Analysis of the Mars Pathfinder Science Data,” Journal of Geophysical Research, Vol. 104, pp. 8831-8852, April, 1999.

[3] J. Wright, F. R. Hartman, B. Cooper, “Immersive Visualization for Mission Operations: Beyond Mars Pathfinder,” Proceedings of SpaceOps ’98, Tokyo, Japan, 1998.

[4] J. Wright, F. R. Hartman, B. Cooper, “Immersive Environment Technologies for Mars Exploration,” Concepts and Approaches for Mars Exploration, Lunar and Planetary Institute, Houston, Texas, 1999.

[5] S. Maxwell, B. Cooper, F. Hartman, J. Wright, J. Yen, “The Design and Architecture of the Rover Sequencing and Visualization Program,” Proceedings of SpaceOps 2004, Montreal, Canada, 2004.

[6] J. Wright, F. R. Hartman, B. Cooper, S. Maxwell, J. Morrison, “Terrain Modeling for Immersive Visualization for the Mars Exploration Rovers,” Proceedings of SpaceOps 2004, Montreal, Canada, 2004.

[7] J. Yen, B. Cooper, F. Hartman, S. Maxwell, J. Wright, “Sequence Rehearsal and Validation on Surface Operations of the Mars Exploration Rovers,” Proceedings of SpaceOps 2004, Montreal, Canada, 2004.

[8] J. N. Maki, J. F. Bell, K. E. Herkenhoff, S. W. Squyres, A. Kiely, M. Klimesh, M. Schwochert, T. Litwin, R. Willson, A. Johnson, M. Maimone, E. Baumgartner, A. Collins, M. Wadsworth, S. T. Elliot, A. Dingizian, D. Brown, E. C. Hagerott, L. Scherr, R. Deen, D. Alexander, J. Lorre, "The Mars Exploration Rover Engineering Cameras," J. Geophys. Res., 108(E12), 8071, December 2003.

[9] Norris, Jeffrey S., Powell, Mark W., Vona, Marsette A., Backes, Paul G., Wick, Justin V.. "Mars Exploration Rover Operations with the Science Activity Planner". To appear in Proceedings, IEEE Conference on Robotics and Automation, Barcelona, Spain, April 2005.

[10] S. Goldberg, M. Maimone, L. Matthies, “Stereo Vision and Rover Navigation Software for Planetary Exploration,” Proceedings of the 2002 IEEE Aerospace Conference, volume 5, Big Sky, Montana, March 2002.