VISUAL AIDS FOR LUNAR ROVER TELE-OPERATION

David P. Miller[1] **and Kyle Machulis**[2]

[1]University of Oklahoma, 865 Asp Ave, Rm 212, Norman OK 73019, USA; dpmiller@ou.edu
[2]KISS Institute for Practical Robotics, 1818 W Lindsey, Bld D-100, Norman OK 73069, USA; kyle@kipr.org

ABSTRACT

Future Lunar missions will require significant and speedy traverses of the terrain by robotic vehicles. Because of the Moons proximity to the Earth it is very tempting to tele-operate robotic vehicles from the Earth, rather than have them operate autonomously. We have found that operators quickly develop the ability to predict the effect of their commands but have difficulty synchronizing their commands with the feedback they are getting, often causing the robot to be over controlled. To help mitigate these effects we have created an operator workstation that graphically shows the operator where their commands are with relation to the feedback loop. In this way the operator can integrate the visual feedback from the robot with the commands they are about to send and those they already have issued. This interface has been used in a rover field test, and a testing simulator has also been created.

Key words: Tele-operation, Lunar rover.

1. INTRODUCTION

[4] outlines NASA's new vision for Lunar exploration. The plan outlines numerous roles for mobile robots on the Moon. These roles include both exploration and survey tasks which run the gamut from pure science to pure logistical operations support.

For NASA to carry out their Lunar objectives they will need to be able to traverse the Lunar surface with robots through all sorts of terrain and conditions. As important as mobility is the ability to remotely control these robots through that hazardous terrain at useful speeds. One of the key factors is the speed and distance that can be covered by a tele-operated robot that has to work with time-delay is operator confidence that the commands they issue will not cause the robot to be damaged or immobilized.

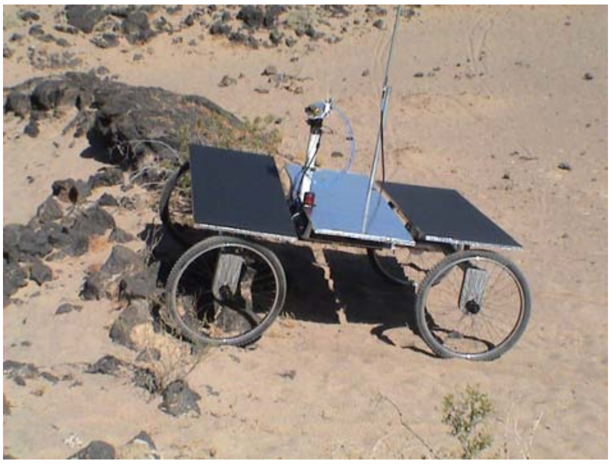


Figure 1. The Copernicus rover at a small volcanic crater rim

Previous lunar rovers, the Lunokhods, had traverse speeds far below a kilometer a day [1]. Additionally, the driver fatigue for the Lunokhods was very high, sometime incapacitating the driver after a relatively short shift of less than two hours.

We believe these low traverse speeds and high operator fatigue and stress are due to two main factors:

1. Poor tele-operation interfaces, and
2. Lack of confidence (perhaps well justified) in the robustness of the mobility system of the rover.

The first of these factors is the subject of this paper. The second is covered in [**?**].

2. TELE-OPERATION INTERFACE

The goal of the tele-operation interface is to allow the operator to easily see the critical information needed for

Proc. of 'The 8th International Symposium on Artifical Intelligence, Robotics and Automation in Space - iSAIRAS', Munich, Germany. 5-8 September 2005, (ESA SP-603, August 2005)

driving under conditions of time-delay. Our basic approach is to have all of the critical data on a single heads-up immersive display. The operator's input comes almost exclusively from that single display, and the operator's commands are all issued through a joystick device. Supplemental data can be brought up on the main display or on additional displays, but that data was seldom actually used in practice, and usually only at the very beginning or end on an operator session.

During the field tests, the rover was controlled from an operator's station located inside of a motor home parked adjacent to the test site. The station communicated to the rover over a wireless link using a directional antenna ($\approx 20^o$ beam width) mounted on a 5m mast on the side of the motor home. The rover used a COTS base station with an amplified omnidirectional antenna.

Figure 2. The operator's panel with the camera pointed at the left front tire

The operator's station is a Powermac Dual G4 workstation with a stereo display formed by two XGA video projectors displaying images through polarized filters onto an aluminized (non-depolarizing) screen and complementary polarized glasses being worn by the operator. A CRT display was used in the rare occasions when the mono wheel-cam image from the USB camera was requested by the operator.

A typical image from one of the main cameras is shown in Figure 2. The icons along the top right of the image show the status of various aspects of the system. The rightmost icon tells the operator that their joystick is in active control of the rover. The next icon to the left indicates whether the GPS[1] is currently active and sending updates. To the left of that are two icons which show whether or not the rover cameras are sending data. And the left most icon in that group indicates the status of the wireless connection to the rover.

In the upper right hand corner of the image is a combined heading, roll and pitch indicator. In the lower right is a transparent overlay of an arial photo (7.5 × 7.5km in this case) with a trail of the robot's movements, as measured by the GPS (or in an actual mission, INS) system, marked in red.

Some previous tele-operated systems (e.g., [6, 2]) have used predictive displays to show the robot's anticipated state (the expected state of the robot at the time when the commands is expected to reach the robot) to assist the operator as new commands are being entered. Predictive displays of that sort have been used for manipulators or free flying spacecraft. We do not feel such displays are appropriate for rovers. An attempt was made to use a predictive display for a simulated Lunar rover [5]. However, while that experiment showed an increase in stability and performance of control over no predictor, it made certain assumptions that are not generally valid. In particular, it assumed a rigid single body rover (and the implicit simplification of rover terrain interaction); it assumed that the upcoming terrain was predictable; and it assumed that mono video image would be updated only once every 8 seconds. Given the much improved display and refresh rates, combined with unknown terrain and therefore unknowable vehicle-terrain interation, we believe a predictive display of that sort would not be particularly beneficial.

Instead of predicting where the rover will be, we remind the operator of what they have told the rover to do by providing them with a streaming command indicator. The lower left graphic in Figure 2 shows the commands being sent to the rover as they work their way through the time delay. In a zero light-time delay situation, this graphic is a square, with the cross hairs indicating the position of the joystick, and hence the command being sent to and instantly received by the rover. For situations with non-zero delays the X and Y axis are extended so that the operator can see what commands are effectively already queued up to be executed by the rover. Figure 2 shows a light time delay of four seconds. The X-axis shows a string of forward motion commands (movement of the joystick forward). Along the Y-axis, the queued commands represent an increasing right hand turn, which after a couple seconds will start to straighten out. The far ends of these command streams are the commands being received and executed by the rover at the time the video from the camera is produced.

In an actual mission, the commands would reach the rover and be executed when they were at about the mid point along the length of the axis, and the feedback reflecting those commands would reach the operator when the commands reached the end of their respective axis. During our terrestrial trials, the commands are buffered in the operator's workstation for the full duration of the delay. They are then sent to the rover and the response from the rover is immediately reflected in the operator's screen. The effect on the rover and the operator in our trials are indistinguishable from an actual two-way delay.

The input device for the driver is a Logitech Attack 3 joy-

---

[1] Of course GPS would not be available on the Moon. However, here we use GPS to simulate the tracking capability that could be performed by a Sun tracker and an inertial navigation system.

stick. In addition to a 2D joystick, this device contains a wealth of readable buttons and scroll devices. The buttons were used to control the pan/tilt head, update and turn the wheelcam, toggle numerical data (e.g., battery voltages, odometry, etc) on and off the main display and for additional engineering functions.

3. TESTING THE INTERFACE IN SIMULATION

3.1. The Simulator

The simulator interface uses the same front end as the Copernicus Rover control software. However, instead of navigating a rover through real terrain, a random 3d height-map is used. The terrain is generated using a Perlin Noise algorithm for smooth slopes and grades, with height settings to match a portion of the Lunar Surface. All movement and camera functions that were available on the Copernicus rover are also available in the simulator. Drive and camera control are provided through the keyboard or the Attack 3 joystick.

On the actual Copernicus rover the cameras are 1.5m off the ground. In the simulator, the camera height was arbitrarily set at 12 units. To keep things scaled in the simulator, the vehicle has a maximum speed of 4.8 units per second. This speed corresponds to maximum speed of 60cm/second – which approximates the no load speed on the Copernicus rover.

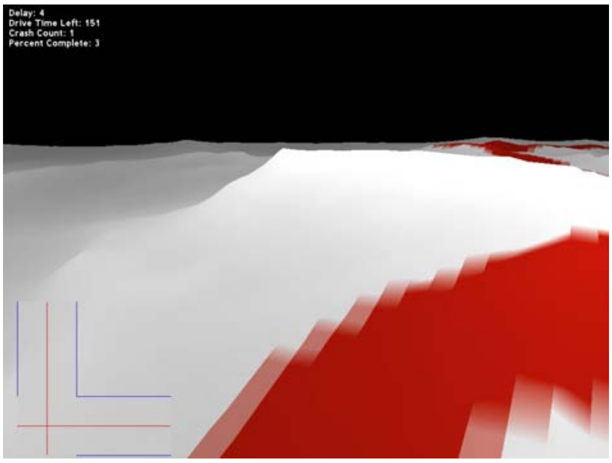

Delay: 4
Drive Time Left: 151
Crash Count: 1
Percent Complete: 3

Figure 3. The simulator interface with the rover stopped on the path (shown in red).

During simulation, users are given a set path to follow (Figure 3). The path is roughly 1m wide and if the center point of the rover ever leaves the path, then robot is considered to be off the path. With a maximum velocity of 0.6m/s, this means it can take considerably less than the Lunar time delay for the user to veer off the course. The path is generated using the same basic navigation goals as the Copernicus Rover, following valleys in the terrain and avoiding large dropoffs. Users are also required to navigate through very tight turns that require complex movement such as backing up and recentering in order to stay on the path. In order to inspect the upcoming path, the user can move the camera, thought this movement is bound to the same time delays as driving.

The path is bounded on either side, so that if the user drives off the path, it will be counted as a ”crash”. This is used as a metric for the effectiveness of the driving tools. After a crash, the user’s position is reset to the middle of the path, with their trajectory set to follow the path again. All prior commands are cleared from the system. After being given a small amount of time to prepare to drive again, the user can resume driving along the path until time runs out.

After the driving time has elapsed, a log of the session is saved in comma seperated value format for analysis. This log contains information on position and velocity versus time, as well as what path and terrain were used, collision points during the session, and percentage of the path completed before time ran out.

3.2. Testing Protocols

As of this writing, only a very preliminary set of tests has been performed with the simulator, though almost a hundred hours of drive time with the actual rover have been performed.

For the simulator tests, the subjects were divided into two groups, half following protocol A and the others protocol B.

Protocol A:

1. 1 minute practice path 1, 0 time delay
2. 1 minute practice path 1, 4 sec time delay, no aid
3. 1 minute practice path 1, 4 sec time delay + aid

4. 3 minute path 2, 4 sec time delay + aid
5. 3 minute path 3, 4 sec time delay, no aid
6. 3 minute path 2, 4 sec time delay, no aid
7. 3 minute path 3, 4 sec time delay + aid

Protocol B:

1. 1 minute practice path 1, 0 time delay
2. 1 minute practice path 1, 4 sec time delay + aid
3. 1 minute practice path 1 4 sec time delay, no aid

4. 3 minute path 2, 4 sec time delay no aid

5. 3 minute path 3, 4 sec time delay + aid
6. 3 minute path 2, 4 sec time delay + aid
7. 3 minute path 3, 4 sec time delay no aid

Before the experiment begins, users input their name, a communication delay time, and a time limit. They also choose whether or not to use the driving helper. Time limit, delay time, score, and percentage of course completion are shown to the user at all times. For the experiment, communication delay was always set to 4 seconds, with the users running over a total of 3 different courses. During these runs, the command visualizer aid is turned on and off in order to see what effect it has on the quality of navigation.

3.3. Simulation Results and Analysis

The data set for the simulator, collected at the time of this writing, is shown in Figure 4. Looking at the numbers alone it is not at all clear that the time delay visualization aid actually helps in performance at all. However this data may be misleading for several reasons.

When asked after their run whether the visualization aid was helpful, made no difference or a distraction, every user answered that it was a tremendous help. They all indicated that it made things much easier. They all said that it allowed them to move continuously rather than in a stop and go fashion (e.g., Figure 5). All were surprised to find out that it had not radically improved their performance on the test.

Additionally, the test administrator said that when driving with the aid the subjects did indeed drive more continuously, and acted with much more confidence.

Half the subjects volunteered that the test was way too short, and that a half hour or more should be done for each run. One subject offered that it was very unpleasant to drive with time delay and without the aid.

Another factor may be that the task of following a line is not a sufficiently realistic driving task. Perhaps the simulator needs to be retooled to reflect obstacle avoidance on the way to a goal to provide more meaningful data.

Based on this feedback we are planning a much more extensive set of tests

4. FIELD TESTS

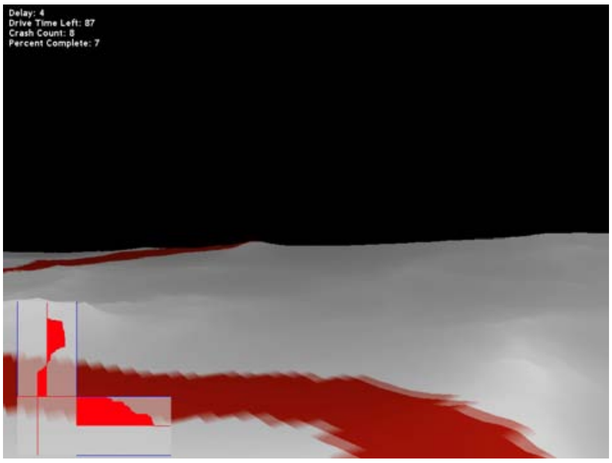
In May of 2004 the Copernicus rover was loaded into a motor home that also housed the control station, and driven west to Amboy, California. Amboy is a 70km^2 primarily pahoehoe lava field (see Figure 6) with a prominent 75m high cinder cone near the Northern edge [7].

Subject	Path	Aid	Time	DT	#Cr	%
A	1	On	60	0	3	18
A	1	Off	60	4	3	9
A	1	On	60	4	4	9
A	2	On	180	4	7	14
A	3	Off	180	4	4	13
A	2	Off	180	4	1	11
A	3	On	180	4	3	13
B	1	On	60	0	3	25
B	1	Off	60	4	4	10
B	1	On	60	4	3	12
B	2	Off	180	4	7	23
B	3	On	180	4	8	18
B	2	On	180	4	6	18
B	3	Off	180	4	4	10
C	1	On	60	0	3	13
C	1	Off	60	4	2	7
C	1	On	60	4	2	5
C	2	On	180	4	5	11
C	3	Off	180	4	4	12
C	2	Off	180	4	2	11
C	3	On	180	4	6	20
C	2	Off	180	4	3	12
C	3	On	180	4	2	18
C	2	On	180	4	3	12
C	3	Off	180	4	2	19
D	1	On	60	0	2	7
D	1	On	60	4	2	3
D	1	Off	60	4	1	5
D	2	Off	180	4	3	10
D	3	On	180	4	4	13
D	2	On	180	4	3	13
D	3	Off	180	4	3	18
D	2	On	180	4	7	22
D	3	Off	180	4	3	13
D	2	Off	180	4	2	12
D	3	On	180	4	4	18

Figure 4. The data set for simulator tests. Time is the amount of time the subject was allowed, DT is the time-delay, #Cr is the number of times the rover strayed off the path and % is the percent of the path that was completed.

The terrain is littered with boulders, broken sheets of rock, occasional ah ah lava flows, ridges and small (10m in diameter and a couple meters deep) vent-hole craters.

Several days of mobility and driver testing were performed. During this period the rover was run over a half kilometer course a number of times using different image resolutions, frame rates and time delays. The drivers got a feel for what the rover was and was not capable of, and also learned how to deal with the issues of time delay. After the second day the communications parameters were frozen at a four second time delay (lunar time delay plus 1.3 seconds to account for latency in the communications network) and a 100kbit/sec data rate from the rover. This is a data rate easily achievable from a rover on the Moon equipped with a hemispherical antenna, broadcasting to the 10M dish of the DSN. The data rate limit meant that



Delay: 4
Drive Time Left: 87
Crash Count: 8
Percent Complete: 7

Figure 5. The subject sets up a four second command string, to go forward right and then forward left, as shown in the display at the bottom left of the screen.

Figure 6. Copernicus in the middle of the Amboy lava field

the JPEG image quality level was set rather low, and the frame rate would bounce between 1.5 and 2 FPS depending on the particular compressibility of the images. An example image from the rover camera is shown in Figure 7.

On the sixth day of testing the rover was driven to the cinder cone and back – a round trip of almost four kilometers. During the trip the rover encountered and traversed slopes of up to 34^o. The average speed (while moving) on the way out was 0.355m/sec^2. The trip to the cone (1.9km) required 170 minutes. This means that including stops for panoramas, rest breaks for the driver and other delays, the rover still averaged 0.186m/sec; over 16km per day.

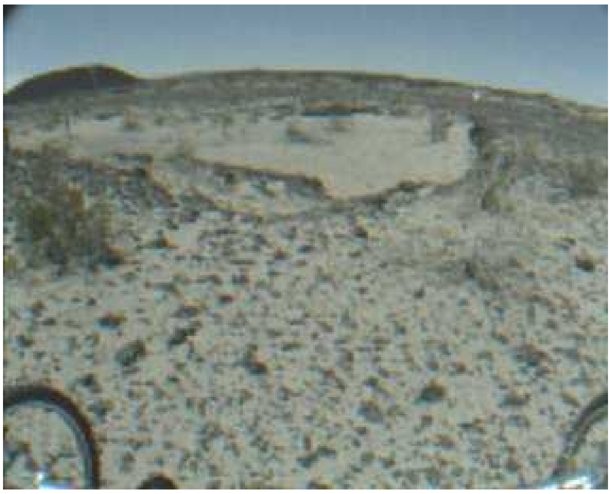


Figure 7. Image of the Amboy cinder cone taken from the rover camera at field test resolution

Interestingly, by the end of the field test the physical rover was being treated by the drivers similarly to the way the simulated rover was treated by the test subjects. The rover did not need to be coddled. The approach the drivers took between the first day of testing and the last differed greatly. The stereo images and slightly lower than normal height view point of the rover cameras tended to exaggerate the roughness of the terrain to a driver's eyes. The view from the rover as it went over an embankment into a crater was often quite frightening. But experience showed that the rover could do it and that the speed that the rover went over a ridge or a rock, or down a slope, was in most cases immaterial. By this last day of testing, the rover was usually being driven at close to full speed for a significant portion of the time. It was still necessary to pause on occasion to get a panorama, so as to plot out the next phase of driving, but when the rover was moving it tended to be moving at full speed – and it was moving more than half the time.

5. CONCLUSIONS

We have created a tele-operation interface that allows a capable rover to be driven with time-delay and low-bandwidth communications. The operators report low fatigue levels. This performance is similar or better than that reported in systems such as [3]. However, in that system there was a sizable amount of intelligence onboard (sometimes simulated by a human) the robot – performing obstacle recognition and command vetoing without any time-delay.

---

[2] Schedule constraints on the return trip required the time-delay to be eliminated for a portion of the return, so the speed during the return trip is not considered valid.

Our simulation results are not definitive as to the value of our time-delay visualization tool, however statements by our subjects indicate that it very beneficial – at least in lowering the operator's stress and raising the operator's confidence. More extensive simulation tests are being planned so as to determine the aid's value to actual driving performance.

ACKNOWLEDGMENT

The authors wish to thank the Copernicus rover team. This work was supported in part by grants from MSSS Inc. and KIPR.

REFERENCES

[1] The vsm: Lunokhod-2 25-year anniversary. `http://vsm.host.ru/e_lunhod.htm`, 1998.

[2] Jeffery B. Ellis. An investigation of predictive and adaptive model-based methods for direct ground-to-space teleoperation with time delay. Master's thesis, Wright State University, 1998.

[3] Eric Krotkov, Reid Simmons, Fabio Cozman, and S. Koenig. Safeguarded teleoperation for lunar rovers: From human factors to field trials. In *IEEE Planetary Rover Technology and Systems Workshop*, April 1996.

[4] NASA. The vision for space exploration. `http://www.nasa.gov/pdf/55584main_vision_space_exploration-hi-res.pdf`, February 2004.

[5] T. B. Sheridan and W. L. Verplank. Human and computer control of undersea teleoperators. Technical report, MIT Man-Machine Systems Lab, 1978.

[6] Thomas B. Sheridan. Space teleoperation through time delay: Review and prognosis. *IEEE Transactions on Robotics and Automation*, 94:592–606, October 1993.

[7] C.A. Wood and J. Kienle. *Volcanoes of North America: United States and Canada*. Cambridge University Press, 1990.