Classifying Autonomy for Mobile Space Exploration Robots

J. Schwendner* and S. Joyeux*

*DFKI Robotics Innovation Center, Bremen, Germany
e-mail: jakob.schwendner@dfki.de, sylvain.joyeux@dfki.de

Abstract

Robotic exploration missions to celestial bodies in our solar system provide us with the necessary knowledge to prepare for human presence on the moon or other planets, and could have the ability to give vital information about our own origins. Direct control over the mobile robot systems is only possible in particular cases, and the ability of the systems to take decisions autonomously increased in importance with more complex mission scenarios and targets further away. The ability to classify and benchmark autonomy is important for the design, evaluation and comparison of such systems. One of the problems is that existing schemes to classify autonomy are either too generic, or don't fit the requirements for mobile space exploration systems. The existing schemes for space engineering and other domains are discussed. Further, existing and planned robotic exploration missions are analysed towards their autonomy content. A new scheme for the classification of autonomy in the given domain is drafted and the existing systems classified. The evaluation of performance levels for autonomy is discussed, and initial ideas on how this could be feasible given. The paper is considered a starting point for further work in the area of classification and benchmarking of autonomy for mobile space exploration robots.

1 Introduction

The exploration of unknown places has always been an important part of our history. Traditionally this was the domain of brave man and woman often risking their lives to venture into the unknown. Today, we have the possibility to send machines in our stead, taking the physical risk for us.

For an exploration mission to be successful, we have to acquire new knowledge from the places we visit, and relate it to the things that are familiar to us. Judging which actions to take in order to gain the most insight is often a hard task. Up until now, human operators were responsible to handle that part of the mission. An alternative solution is to let the robotic system decide for itself what courses of action to take, in order to extend the information known about the environment it is in. In such a scenario only information that constitutes new knowledge is

transmitted back to the operator. As exploration missions go further away from Earth, this alternative solution might become more important to pursue.

There are usually two parts in any exploration scenario. On the one hand, one wants to verify scientific assumptions, i.e. find something particular that is expected by the scientists. On the other hand, it is very important that anything that is *not* expected gets investigated, as it may represent very valuable information. Autonomous information systems have a hard time here, since they are commonly programmed to cope with foreseen situations. The capability to be truly flexible is one of the prime examples where humans will be far superior to robots for a long time to come.

In order to drive the development of autonomous systems in the context of space missions, it is necessary to specify the level of autonomy which is required for the task and feasible to implement with the given resources. Although classification of autonomy levels are given for certain domains (e.g. ECSS Standard for European Space Autonomy), no definition has been provided for the domain of robotic exploration systems, yet.

In this paper we seek to provide a starting point for such a taxonomy of autonomy levels and give information on how to potentially benchmark and test them. Further, we use our scheme to classify robotic exploration systems for the space domain, and compare it to the existing classification schemes. We show that the existing schemes are not sufficient to represent the various aspects of exploration robot autonomy and that the proposed scheme is more suited for this. Finally, the possibility to test and/or benchmark such a scheme is also investigated.

1.1 Defining Autonomy

Numerous papers tried to define what autonomy *means*. What this section will do is first define a few terms, that are critical to explain the rationale underlying the definition we picked in this work.

Task, Environment and Scenario

task an abstract action which represents what the system should do. This definition will most often be general, i.e. can be reused across a wide range of systems and

situations.(e.g. Find evidence of water, map geological features, or explore cave system)

environment a description of the *context* in which the robot will operate and achieve the assigned task. In contrast to tasks, environment definitions are very specific to a mission (e.g., crater rims in moon's south pole for the NEXT-LL mission).

scenario a scenario is an association between one or multiple tasks and an environment.

What underlies the work presented in this paper is the belief that autonomy ratings should always be put in context, i.e. should always be considered with respect to a scenario. The rationale behind that is that a system that is well adapted to a particular environment (for instance lighting conditions on the Moon's south pole) might have poor autonomy on earth and high autonomy during its mission. In an identical manner, a system that is well suited for some tasks might not be usable at all for different ones. Take for instance exploration and infrastructure building tasks. The former requires the system to be able to handle the *unknown* (i.e. gather information and form a "big picture" of what is available), while the latter will probably provide an already good picture of the environment, probably in the form of a map. Some systems will probably be specialized in the first task, while others will be in the second task. They will both need specific sets of skills, some of them being common but some of them different.

Moreover, the definition of autonomy (and autonomy levels) should be general enough not to request a particular system architecture and/or solution. In other words, how autonomy and autonomy levels are defined should not constrain the space of solutions. It should only provide a standardized way to describe these solutions.

We therefore reduce the definition of autonomy to a minimum:

Autonomy is the ability for a system to achieve certain tasks in a specific environment by itself

It should be noted, that our definition of autonomy does not include the performance of the system. A completely autonomous system is not guaranteed to perform well under any given performance metric, and might perform much worse than a system with a lower level of autonomy.

1.2 Why do we need autonomy levels ?

From a practical point of view, a classification of autonomy has two main uses. First, the ability to describe autonomous capabilities to "upper management", i.e. in the context of development programs, in order to plan technological developments. Second, the ability to compare systems with each other, and to specify requirements in the frame of specific missions.

From a general point of view, it allows to classify systems through their capabilities.

2 Review

2.1 Existing schemes

In all existing schemes that we know of, it is acknowledged that autonomy cannot be rated in one global scale. This is because there are different aspects to autonomy, each aspect being more or less independent of each other. It is therefore possible to improve an aspect independently of the other ones. To represent this, the schemes we reviewed split the autonomy classification they propose along various axis that are then split into fine-grained levels.

In this section, we will focus on three different schemes that we believe are representative of autonomy classification schemes. First, the ECSS standard ECSS-E-ST-70-11C for space segment operability, which is the European standard for autonomy levels, provides a very simple qualitative assessment of autonomy. Second, the ALFUS framework, which provides a generic *quantitative* scheme for measuring autonomy. Finally, a scheme that has been developed by the US Air Force, targeted at UAV autonomy.

The European space standard body, ECSS, splits the autonomy levels into two aspects: "Mission execution" and "Fault Detection, Isolation and Recovery" (FDIR). The mission execution aspect is very classical, from direct ground control in E1 to "goal-oriented mission operations on-board" in E4 [5]. The FDIR aspect has only two levels: the F1 level in which the system is able to put itself in a safe condition upon failures, and the F2 level in which it can reestablish normal operations by himself. We unfortunately did not find the NASA equivalent of the ECSS standard for space operations.

The ALFUS framework [9], in contrast, aims at providing metrics that can provide a quantitative assessment of autonomy, mainly in the domain of ground systems. They do so along three axis. First, the *Mission Complexity* aspect measures how complex the mission is from the point of view of the system's decision-making component. It includes measures such as the number of interdependent choices that the system must do to achieve its goal. Second, the *Human Independence* aspect quantifies how much the system has to interact with its operators to achieve its mission. Finally, the *Environmental Complexity* aspect measures how difficult the environment is for a particular system.

Finally, the scheme in [8] splits into four axis: Perception/Situation Awareness, Analysis/Coordination, De-

cision Making and Communication. One very important point that the authors make is that, while the general axis can be reused, the definition of each levels in the axis will – and should – be interpreted differently in different scenarios/domains. This is illustrated by the fact that general names are used for each of the levels (e.g. "Fault/Event Adaptive Vehicle" for level 5), while the detailed description is very specific to the author's domain of interest (UAVs).

In our opinion, none of these schemes apply to the domain of space exploration systems, the domain we focus on.

First and foremost, we believe that a more detailed scheme than what ECSS proposes must be available for future space missions. Indeed, autonomy is a critical technology for deep space exploration, and a detailed scheme will provide decision-makers with the basic understanding of what an autonomous exploration system is made of in the context of exploration missions.

Second, quantitative metrics should be limited to very specific solutions. One trait of the ALFUS scheme is that the metrics are applicable only if one assumes that the evaluated solutions are based on some specific architecture. For instance, most of the decision-making metrics assume that planning and task decomposition schemes are at the root of the autonomous system. We, in contrast, believe that an autonomy description scheme should not constrain the solutions it can be applied on. Moreover, the ALFUS scheme advocates that the various metrics can be combined into a weighted sum to provide an overall autonomy value. We do not believe that such a quantitative combination is meaningful in the evaluation of autonomy.

Finally, we actually agree with most of the analysis included in the UAV classification scheme: that, to be useful in operational context, an autonomy classification needs to focus – at least partially – on the actual domain. As we will see later, we did need to take into account some aspects that are relevant for space exploration. Namely, the fact that most of the system's environment is unknown (contrary to having fine-grained and reliable maps and localization systems as it is for instance the case on urban environments on Earth).

2.2 Existing Systems

The evaluation of existing systems is limited to lunar or planetary exploration missions that include a mobile robotic element. So far only two celestial bodies in the solar system (excluding of course earth) have been visited by mobile robots: the moon and mars.

Lunokhod The Russian Lunokhod I (1970) and II (1973) rovers [7] where the first mobile robotic systems to operate on a celestial body. Both rovers where fully teleoperated. Video data was relayed back to the operators which directly controlled the movements of the vehicle. Lunokhod II managed to travel a distance of 37 km over in a time period about 4 months.

Sojourner was the first mobile exploration robot to successfully operate on another planet. In 1997 the rover operated for 83 days on the Martian surface and examined rock samples in a radius of 10 m around the lander. Because of the large time delay, the operators where only able to communicate with the rover once per sol (24.6 h). Because of this, sojourner already included autonomous traits, like simple waypoint navigation, including a hazard avoidance behaviour [11]. Hazard detection was performed using an array of laser stripes for proximity sensing, and an accelerometer for detection of hazardous slopes and a bumper sensor as the last resort.

MER The Mars Exploration Rovers (MER) Spirit and Opportunity are continuing to explore the Martian surface since 2004. The rovers include software autonomous capabilities in the area of navigation, science instrument placement, resource management and science data gathering [10]. Like the Sojourner, the MER operators only had a single communication window per sol. Autonomous navigation capabilities and hazard avoidance allowed the operators to give movement commands that go beyond the range at which obstacles can safely be identified and thus would have to rely on the on-board autonomy to perform this feature. Currently Opportunity is still operational after 6 years of mission time, and managed to cover a distance of more than 20 km in that time. The autonomy levels of the MER were increased through software update in the course of the mission time, which added a global path planner, target tracking and positioning as well as science event detection capabilities.

ExoMars and MSL Exomars and Mars Science Laboratory are two missions currently in the planning by ESA and NASA. Although not finally fixed, the autonomous capabilities of the two systems are expected to be higher compared to the MER rovers. Both navigation as well as science target selection abilities should be improved compared to the MER rovers. Work has also been performed on improved adaptive capabilities of the systems, like for example the ability to predict wheel slip based on past experience [6].

MSR The next step after MSL and ExoMars is considered to be a Mars Sample Return Mission (MSR) which aims at collecting rock samples to be delivered to an ascent vehicle. Due to limitations in temperature cycling of the fuel, the ascent vehicle would have to lift off again within a years time, and thus

limits the overall mission time. The mobile robot has to collect as many samples from different locations as possible within the given the time frame. This can only be achieved with a high level of autonomy, so that the rover does not have to rely on commands from the operators for travelling [6].

NEXT-LL NEXT Lunar Lander is an activity currently pursued at ESA which aims at a precursor mission to human presence on the moon, and might contain a mobile robotic element for exploration. The requirements for autonomy on this mission are limited, since the communication latency is within range for direct teleoperation. Nevertheless, for technology demonstration purposes, and to improve reliability, the system could potentially have some autonomous abilities available, especially in the area of diagnosis and fault recovery.

3 Method

3.1 Designing and Interpreting Autonomy Levels

One very important aspect, when assessing the autonomy of a system, is that measuring the capabilities of single subsystems does in no way allow to infer the autonomy of the whole system. This is so for two reasons:

- autonomy can be created by having the components interact with each other (i.e. by having a system). This is seen in cognitive architectures, where learning algorithms can end up having a deep situation awareness while components – in themselves – have little understanding of the overall goal and mission

- conversely, bad interactions between components can *degrade* the overall system autonomy.

The proposed scheme gives a qualitative assessment of a system autonomy in a given scenario. In no way do we aim at providing a quantitative measure of autonomy.

The proposed scheme does not have any assumption on the underlying architecture, which is a problem we identified in the ALFUS scheme.

Finally, one issue have arisen when we designed this scheme. We first tried to represent these levels on linear scales, conveying that to get a level 5 on one aspect, one would have first to first obtain levels 1 to 4. However, it ended up being difficult to obtain for the decision-making aspect.

To solve this problem, instead of adding new aspects – which would have added complexity to the description – we kept the representation of each aspect linear. The levels therefore might not represent dependencies between more or less advanced capabilities, but instead a level of

importance: we felt that, for instance, the ability to reason about time and resources (level 3 in decision-making) is more important than the ability to adapt to changing performance (level 6), and would therefore need to be implemented before. However, it should be stressed that it is our own opinion and that in some scenarios adaptation might be more important than complex reasoning capabilities.

3.2 Proposed Aspects

We propose to separate the description of autonomy into four different aspects (or axis):

Information Interpretation this is obviously critical in exploration scenarios. It describes the reasoning capabilities the system has with respect to the information that it has, and the information that is present in its environment.
The proposed levels are on Table 1

Decision-Making the mechanisms that allows the system to choose its course of action in order to achieve its goal. *Decision* in an autonomous system is a difficult term to define. What we propose is to limit it to the ability of the system to choose. I.e., given a scenario, if there are multiple ways to achieve the task, how is the system able to pick one way which is viable.
The proposed levels are on Table 2

Diagnostics assessment of the status of its different subsystems (software and hardware), and of how these subsystems interact together.
The proposed levels are on Table 3

Fault Recovery "Fault" is a bit misused here. What this aspect covers are all the mechanisms that allow the system to cope with the changes to its parts (including malfunctions).
The proposed levels are on Table 4

4 Classification of Existing Systems

In order to verify the proposed scheme, the previously mentioned systems have been analysed towards their autonomy aspects, and classified into the categories given. Figure 1 shows the result of this classification and compares the individual results. It does not come at a surprise, that the lunar systems rate much lower in the autonomy categories compared to the mars rovers. It is also notable that there is a low variance on the fault recovery axis.

5 Testing Autonomy

Benchmarking is the process of comparing one's business processes and performance metrics to industry bests and/or best practices (Wikipedia)

Level	Capability
1	Simple sensor processing without interpretation. Example: video camera with preprocessing, which simply stores the images and sends them to Earth.
2	Information assessment: can evaluate how important its sensor data is. An example is the event-based data gathering on the MERs, where data is stored only if the sensor information matches specific triggers
3	Is able to classify expected from unexpected. Most classifiers can tell how well certain samples fit into the classes they know. This capability extends on that by having the classifiers be able to announce that a data sample probably does not fit into the classifier's models
4	Model parameter correction: the classifiers can modify the number of classes and/or the importance of each class
5	Model structure correction: adapt the way classification itself is done, for instance by changing the type of classification itself.

Table 1. Proposed levels for the **Information Interpretation** aspect

Level	Capability
1	No decision-making capability. Executes prepared sequences of actions.
2	The system is able to decide its course of actions, but only reasons on what needs to be done (i.e. no reasoning on resource and/or time constraints)
3	Takes resources and/or time constraints into consideration
4	Can reason on how information affects the system. The system can represent the impact its decisions has on the performance of its subsystems (localization, map-building, . . .), and use that representation to make its choices.
5	Can evaluate the exactness of its own decisions. Given a certain information (and the assessment on this information), it is able to evaluate its decisions and provide a metric – qualitative or quantitative – that represents how sure it is that its decision is "the right one"
6	Adapts the reward / cost representation used for its decisions based on prior experience
7	Adapts the models used for reasoning to either fix it (corrections) or improve it. Example: optimizing dependencies between tasks in planning

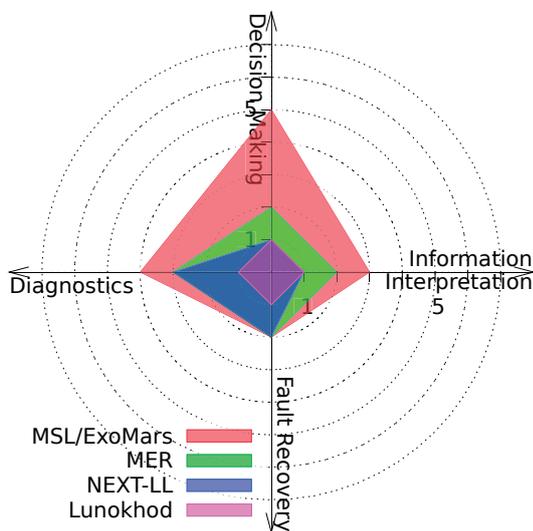**Table 2.** Proposed levels for the **Decision-Making** aspect

Figure 1. Comparing the autonomy levels of different exploration robots based on the proposed classification scheme

Level	Capability
1	No diagnostics
2	Single component failure detection
3	Subsystem failure detection
4	Detection of performance degradation
5	Detection of undesired behaviour. Example: detects temporal inconsistency in the actions: detects that the robot's plan execution system is stuck in a loop.
6	Adaptation to changing performance. During the system's lifetime, some components/hardware will probably evolve. The diagnostics routines that evaluate these components should therefore also be modified to take into account this evolution.

Table 3. Proposed levels for the Diagnostics aspect

Level	Capability
1	Fail-safe capability
2	Single-component redundancy
3	System reconfiguration in case a fault is detected. Example: the system is able to take into account that some sensor or subsystem is not available anymore, and still be functional in a degraded way.
4	Adaptation to changing performance. During the system's lifetime, some components/hardware will probably evolve. This obviously has an effect on the ability of the system to act. This capability will take these changes into account to both act as well as possible given the changed performance, and so that decision-making aspect can still operate.

Table 4. Proposed levels for the **Fault Recovery** aspect

What is then commonly accepted as benchmarks are standardized quantitative metrics that allow to quantify how well a particular solution behaves with respect to existing best practices. As we argued already, we believe that this kind of methodology does not apply to autonomy capabilities, as *autonomy capabilities* are dependent from the particular scenarios the system will be confronted with.

These last years, challenges have started to appear (DARPA's Grand Challenge and Urban Grand Challenge, micro-UAV competition, ELROB [1], ESA's LRC [2], Sauc-E [4]). Instead of trying to benchmark systems against a standardized metric, they evaluate systems by making them compete against each other in the same tests.

One *very* important aspect of these challenges is that, while the competing teams get task and environment descriptions beforehand, the specifics of the scenarios (details of the environment, trajectories, . . .) are not available. It is central to compare each system's robustness, i.e. how the autonomy part can robustly perform regardless of the actual instance of a scenario that it has to confront.

However, while they provide a forum in which systems can be compared to each other, these challenges are far from being *tests*. I.e. they do not provide a way to verify that a given system *does conform* to the autonomy levels it is supposed to.

We will now investigate what kind of tests could be used for each of the autonomy aspects we identified, and outline some ideas of how these could be implemented.

Environment Representation and Information Gathering test environments should include environment that are both benign ("expected") and extreme ("unexpected").

Adding totally unexpected elements could also be used as a way to see how the information assessment system can handle unknowns-unknowns.

Decision making, Diagnostics and fault recovery Low autonomy levels on these aspects can be unit-tested extensively. Evaluating the higher levels, however, is more tricky, as it would probably involve tight cooperation between the three aspects.

In general, an interesting way of testing high levels of autonomy would be to see the system as playing a game (in the "game theory" sense of the term) against the rest of the world. From that point of view, a test would be interactive: the system to be tested would "play" against computers and/or humans that would decide what goes wrong in realtime. Their ability to impact the software of the system to be tested being, for instance:

- simulating hardware failures

- simulating sensing degradation, e.g. changing the sensor error models.

- injecting wrong information into the system's information database – thus simulating potential problems on the information assessment aspect.

- injecting wrong decisions superseding the actual decisions taken by the system – thus simulating potential problems on the decision-making aspect.

In difference from a unit-testing, all these changes would be implemented in real time in a real environment. *When* a change is performed is indeed critical: it will be the role of the "test player" to choose to implement the change at what seems to be the worst situation possible for the system that is being tested.

Importantly, such a testing "game" would need to stay within the boundaries which the system is supposed to handle. It would indeed make no sense to simulate critical sensing degradation on a system that claims a level of two on diagnostics and error recovery.

Testing the hardware capabilities for recovery, however, would be a lot harder. It would indeed require to modify the environment at will, so as to have a broad range of situations the system should be confronted with, which is not practical.

In general, to allow extensive testing, one would require to build:

- a standardized software interface that would be provide the interface necessary to build "game-based" tests, regardless of the actual system implementation. Such a scheme is already applied on a completely different scheme in the domain of automated planning [3].

- a network of testing facilities. Over the years, numerous testing facilities have been created by different institutions. Making these facilities available would be a great step towards the general goal. Nonetheless, it would still be important to conduct tests and/or challenges in outdoor environments, where the actual test setup is not known in advance to the tested system.

6 Conclusion

The aspect of autonomy for robotic space exploration is no doubt a very important factor in the mission and system design. Current methods of describing autonomy for these domains are either too limited or too generic. We have given an overview of existing and planned robotic systems for the exploration of celestial bodies, and assessed their autonomous capabilities. A scheme was generated that constitutes of four independent scales, which we believe to be fitting to describe the autonomous capabilities of the system for the given scenario. The scheme can give qualitative information about the level of autonomy. Benchmarking a system is an entirely different matter, and can not be performed so easily. We propose to extend the recent trend in robotics to hold competitions, as they are able to provide at least a comparative information on the autonomy of the systems. Further, we propose that a standardised software interface for for fault injection could lead to a better evaluation of the fault tolerance of a system by letting third parties create real unexpected events, rather than responding to expected faults. The work presented is meant to be taken as a starting point for discussions, and further investigation into the subject is required towards the usefulness of the proposed scheme, and the nature of the competitions and measures for evaluating autonomy.

7 Acknowledgements

References

[1] ELROB - The European Robot Trial. Available from: http://www.elrob.org/.

[2] ESA's Lunar Robotic Challenge. Available from: http://www.esa.int/esaCP/SEM4GKRTKMF_index_0.html.

[3] ICAPS Competitions. Available from: http://idm-lab.org/wiki/icaps/index.php/Main/Competitions.

[4] Student Autonomous Underwater Challenges (Sauc-E). Available from: http://www.dstl.gov.uk/news_events/competitions/sauce/index.php.

[5] Space engineering - Space segment operability, 2008.

[6] M. Bajracharya, M.W. Maimone, and D. Helmick. Autonomy for Mars rovers: Past, present, and future. *Computer*, 41(12):4450, December 2008.

[7] Harvey Brian. *Soviet and Russian Lunar Exploration*. Springer Praxis Books. Springer Praxis Books, New York, NY, 2007.

[8] BT Clough. Metrics, schmetrics! How the heck do you determine a UAV's autonomy anyway? *NIST Special Publication*, (990):313319, 2002. Available from: http://www.csa.com/partners/viewrecord.php?requester=gs\&collection=TRD\&recid=200307110896MT.

[9] Hui-Min Huang, Elena Messina, and James Albus. AUTONOMY LEVELS FOR UNMANNED SYSTEMS (ALFUS) Volume II : Framework Models. *Framework*, II(December), 2007.

[10] C. Maimone, M.; Biesiadecki, J.; Tunstel, E.; Cheng, Y.; Leger. *Surface navigation and mobility intelligence on the Mars Exploration Rovers*, chapter 3, pages 45—-69. TSI Press, San Antonio, TX, USA, 2006. Available from: http://www-robotics.jpl.nasa.gov/publications/Mark_Maimone/05_Chapter3_final.pdf.

[11] J. Matijevic. THE PATHFINDER MISSION TO MARS: Autonomous Navigation and the Sojourner Microrover. *Science*, 280(5362):454–455, April 1998.

Autonomous exploration of an unknown environment by a mobile robot can be beneficial as robots can navigate in unknown environments without maps being supplied. Also it provides the possibility to produce maps without human interaction. Frontier based exploration is used here as the method for exploration. Some simulation and real world results with a Pioneer 3-AT are presented and discussed. Both simulation and real world experiments use ROS. Robots are machines that have inbuilt artificial intelligence. They are used in all kinds of situations, especially where they can alleviate strenuous tasks or complete missions that are too dangerous for a human to undertake. Space technology and space applications produce a huge amount of data, including spacecraft telemetry and product data â€" the useful scientific data that a spacecraft gathers, for example information about Earth from an Earth observation satellite. Another application of Machine Learning is to analyse all this data. For deep-space exploration NASA has also looked into designing more autonomous spacecraft and landers, so that decisions can be taken on-site, removing the delay resulting from communication relay times. @inproceedings{Schenker2001RoboticAF, title={Robotic autonomy for space: cooperative and reconfigurable mobile surface systems}, author={Paul S. Schenker and Terrance L. Huntsberger and Paolo Pirjanian and Gerard T. McKee}, year={2001} }. Paul S. Schenker, Terrance L. Huntsberger, +1 author Gerard T. McKee. Published 2001. Computer Science. We overview some of our recent work on highly autonomous mobile robots for planetary exploration. The work and related system developments address two themes. The first is multi-robot cooperation to perform planetary surface tasks beyond the scope of single