# Behavior-Based Control for Autonomous Mobile Robots

Terry Huntsberger[a] and John Rose[b]

**Abstract**

The harsh nature of planetary surfaces introduces many new constraints into the types of control systems suitable for use in such environments. These include low power requirements, operation within wide temperature extremes, relatively low computing capabilities and onboard memory, and multiple terrain types ranging from featureless flat plains to sheer cliffs. This paper presents the results of ongoing work at JPL and USC in autonomous rover control, which has concentrated on the development of a behavior-based control system called BISMARC. BISMARC includes full sensor and mobility models, geometric constraints, and environmental feedback based on the SRR/FIDO rovers at JPL. It is based on a biologically motivated navigation system derived from a study of human path planning in complicated exterior environments. The system has had over 800 simulation runs of a multiple cooperating rover, multiple sample return mission scenario with an overall success rate of 98.9%, and is currently being tested on the SRR rover in the Planetary Robotics Lab at JPL.

Keywords: Behavior-based control, neural networks, free flow hierarchy

## Introduction

Current NASA plans call for extended yearlong, multikilometer treks for the 2003 and 2005 Mars missions. A much greater amount of rover autonomy is required compared to the recent Sojourner mission, where the rover stayed within a 50-meter radius of the Pathfinder lander. Two mission prototype systems include the

[a] Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Pasadena, CA 91109, terry@telerobotics.jpl.nasa.gov

[b] Department of Computer Science, University of South Carolina
Columbia, SC 29208, rose@cs.sc.edu

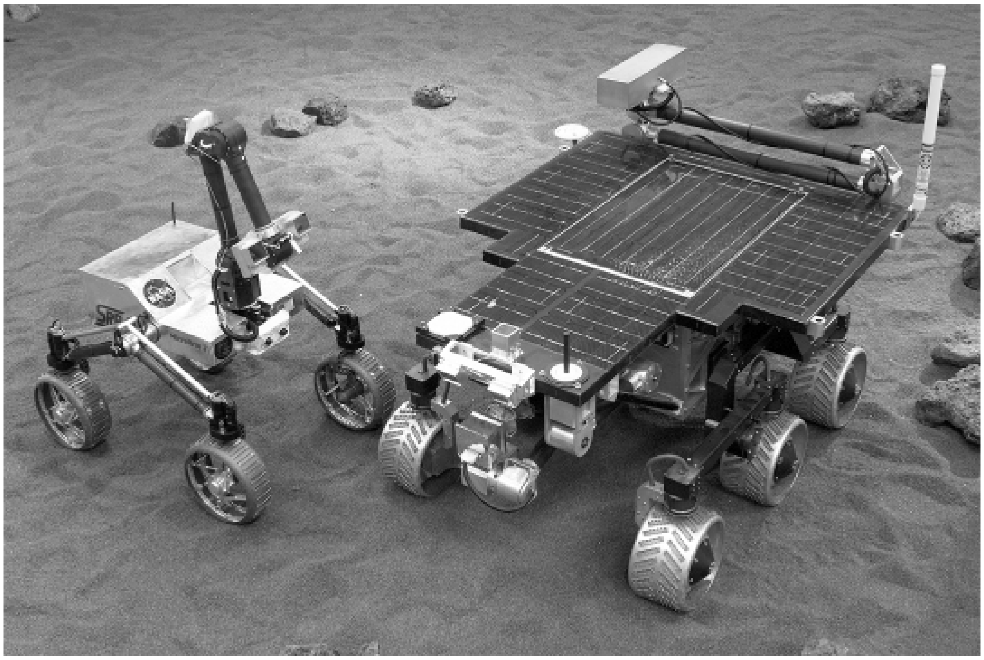

**Figure 1:** SRR and FIDO Mars rover prototypes in the Planetary Robotics Lab at the Jet Propulsion Laboratory in Pasadena, CA.

Sample Return Rover (SRR) and the Field Integrated Design and Operations (FIDO) rover, both shown in Figure 1, which are currently being field tested at the Jet Propulsion Laboratory in Pasadena, CA, USA. SRR is built for low mass, high speed and mobility, while FIDO includes a full science suite equivalent to the 2003 Athena system. Future missions currently under study include the deployment of multiple cooperating robots in a robotic colony.

Behavior-based systems approach the autonomy question from the standpoint of collections of behaviors. These run the gamut from the purely subsumptive, reactive single robots detailed by Brooks [Brooks (1986)] to cooperative multiple robot systems [Robot Colonies (1997), Arkin (1998)]. The wide range of possible behaviors that are needed for a planetary rover obviates the need for an action selection mechanism (ASM) to provide the correct behavior for any given situation. Comprehensive reviews of behavior coordination (or action selection) mechanisms can be found in Arkin (1998) and Pirjanian (1998). Recent work of Pirjanian and Mataric [Pirjanian (1998), Pirjanian & Mataric (1999)] using Multiple Objective Decision Making (MODM) provides formal tools for generating strategies that can guarantee an appropriate trade-off between the optimal solutions, which are not feasible for these types of tasks in a planetary surface environment, and *Pareto-optimal* and *Satisficing* solutions.

BISMARC (**B**iologically **I**nspired **S**ystem for **M**ap-based **A**utonomous **R**over **C**ontrol) is a hybrid wavelet/neural network based system that is capable of the required autonomy for such ambitious missions [Huntsberger & Rose (1998)]. The BISMARC architecture is shown in Figure 2. Previous simulations demonstrated that the system is capable of control for multiple rovers for multiple cache recovery [Huntsberger (1997)] or manned habitat site preparation [Huntsberger, Mataric & Pirjanian (1999)]. A subsequent study extended BISMARC to include fault tolerance in the sensing and mechanical rover subsystems [Huntsberger (1998)]. The vision subsystem in the original BISMARC implementation relied on the generalization capabilities of a fuzzy self-organizing feature map (FSOFM) neural network [Huntsberger & Ajjimarangsee (1990)]. A better vision subsystem based on camera models combined with tilt sensors that is fully integrated into the BISMARC framework was recently developed [Huntsberger, Kubota & Rose (1998)]. The next section gives a general discussion of the architecture of BISMARC, followed by some experimental studies and a concluding section.

## BISMARC Organization

The three level BISMARC system (shown in Figure 2) uses a hybrid mix of neural networks and behavior-based approaches. The first level performs a wavelet transform on the rover's stereo image pair, the second level inputs these processed

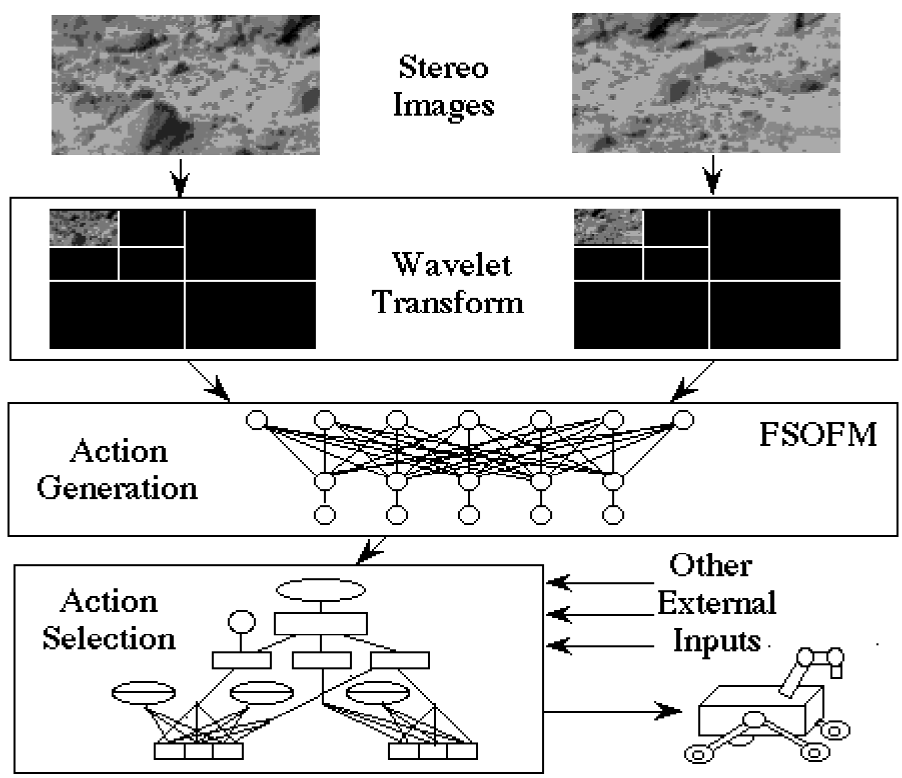


**Figure 2:** Three level BISMARC architecture with stereo processing, action generation, and action selection subsystems.

images into an action generation navigation network, and then to a third level action selection mechanism (ASM) network modeled after that of Rosenblatt and Payton [Rosenblatt & Payton (1989)]. BISMARC is only encoding the raw stereo visual information without any attempt to label individual features or objects beyond the desired action associated with the input pattern.

The action generation level of BISMARC is a FSOFM trained with a representative mix of wavelet processed stereo images for the six movement actions of forward, left turn, right turn, back up, go either way, and goal. BISMARC is only encoding the raw stereo visual information without any attempt to label individual features or objects beyond the desired action associated with the input pattern. In the operational mode, the FSOFM generates membership values to the classes of visual input that the system has previously seen. When coupled with onboard rover components such as accelerometers and dead reckoning inputs, an egocentric map of the environment is built using the FSOFM response as an index. An advantage of using the FSOFM for the action generation level lies in the membership values that are generated at the output nodes. The sum of these values is normalized to one, and the relative size of the membership values gives a ranking of the actions.

The BISMARC ASM for the multiple cache recovery task is shown in Figure 3. The collection of behaviors used by BISMARC can be broadly broken into two categories: survival (i.e. Avoid Dangerous Places), and task specific (i.e. Get Cache). Most tasks will share the same survival behaviors, which allow the rover to carry a set of task behaviors and switch between them if necessary. The survival behaviors include mobility as well as temperature and battery level preservation measures. Sensor feeds are only done at the appropriate level where needed, which eliminates the potential bottlenecks seen in traditional hierarchical ASMs.

Weights on the links between behaviors perform a type of priority weighting, which will ultimately favor selection of the correct action at the bottom level of the hierarchy. For example, the Sleep at Night behavior is the most heavily weighted since absolutely no motion is allowed at night due to the lack of night vision. In the event that sensors such as LIDAR are available, this weighting can be relaxed to allow movement at night. Combination of the weighted links is done in three ways: additive, multiplicative, or through a weighted summation process suggested by Tyrrell [Tyrrell (1993)]. At the bottom level of the ASM hierarchy, are the actions that are available to the rover. These include movement, surveillance, survival, and task specific actions. The movement and surveillance actions are direction specific, while the survival ones tend not to be. Once again, as was the case with the high level behaviors, the rover can carry a set of task specific actions, and select the appropriate one when needed.

Fault detection is built into the ASM using the following form of sensor activation function:

$$A_s = P_d * (1.0 - dist) * (1.0 - P_u), \qquad (1)$$

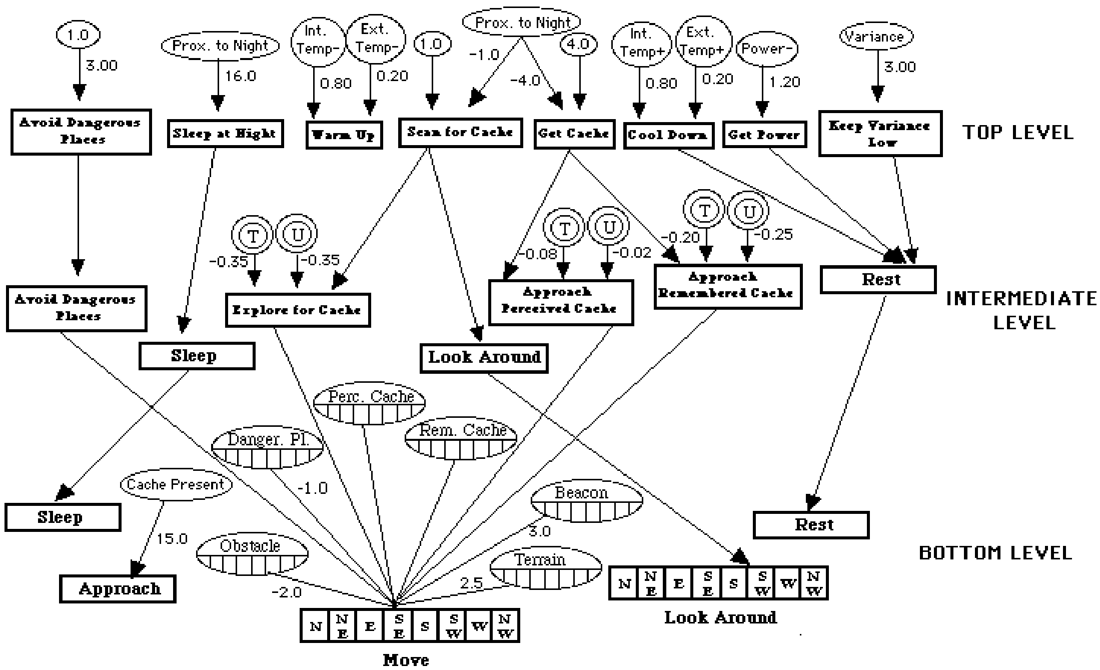


**Figure 3:** BISMARC action selection mechanism for cache retrieval operation.

where $A_S$ is the activation level for sensor S, $P_d$ is the normalized sensor input, dist is the normalized distance to the perceived objects, and $P_u$ is the perception uncertainty. The perception uncertainty is given by:

$$P_u = ABS\ [P_d(t+1)-P_d(t)], \tag{2}$$

where $P_d()$ refers to the time separated normalized sensor samples. This expression for $P_u$ experiences a maximum when the sensor input undergoes a full range swing. The perception uncertainty is used for fault detection (high values indicate a possible fault). Sensors with a high uncertainty will have little effect on subsequent nodes. These sensors are flagged, and are allowed to come back on-line if and when the uncertainty stabilizes.

## Experimental Studies

We ran 1000 trials using a random heightfield based on statistical information returned from the Mars Pathfinder mission. The area encompassed about 1 km by 1 km with a grid decomposition resolution of 5 cm at the detailed map level. Each trial had different starting positions and the placement of 4 cache containers was randomized within the area. Three rovers were deployed for each simulated mission:

a scout and two retrieval rovers. The bandwidth of the communication channel between the rovers is one Megabit/second, which is the same as the modem installed in the current SRR prototype at the Jet Propulsion Laboratory. The top speed on the rovers was set at 10 cm/sec, which is consistent with the SRR. In order to simulate wheel slippage, we set a 15% loss of traction when climbing over a rock or traversing rocky terrain. The battery lifetime was set at one week on all of the rovers and the time step size for the simulations was fixed at 0.5 sec. All of the rovers were forced to sleep during the night hours of the simulations, since there were no infrared sensors on any of the rovers, and navigation at night would be dangerous.

We included a set of possible faults based on a statistical analysis of 200 simulation runs [Huntsberger (1998)]. These faults included loss of one or both stereo cameras in front and back, loss of mobility in one or more wheel sub-assemblies, loss of power regeneration capabilities, loss of one or more wheel encoders, loss of one or more degrees of tilt sensing, and loss of internal temperature sensing capabilities. In the absence of faults the success rate for cache retrieval was 98.9%. Faults caused this rate to drop to 16% without fault tolerance, with an increase to 46% with the fault tolerant weight adjustment discussed above.

## Conclusions

This paper has presented a behavior-based system called BISMARC that is being evaluated for autonomous control of rovers on planetary surfaces. The system has shown itself in 800 simulations to be capable of successfully completing complicated multirover missions. Fault tolerant adaptation of the weights in between the behaviors has extended the system for long duration capabilities such as the 4-year Mars outpost mission being considered for a launch in 2007. This flexibility indicates that BISMARC is also well suited for terrestrial applications such as urban reconnaissance. We are currently porting the algorithm to SRR at JPL.

## Acknowledgments

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

## References

**Arkin (1998):** Behavior-Based Robotics, MIT Press, Cambridge, MA.

**Brooks (1986):** "A robust layered control system for a mobile robot", IEEE Journal of Robotics and Automation, 2(1), 14-23.

**Huntsberger (1997):** "Autonomous multi-rover system for complex planetary surface retrieval operations", Proc. Sensor Fusion and Decentralized Control in Autonomous Robotic Systems, SPIE Vol. 3209, Boston, MA, 220-229.

**Huntsberger (1998):** "Fault-tolerant action selection for planetary rover control", Proc. Sensor Fusion and Decentralized Control in Robotic Systems, SPIE Vol. 3523, Boston, MA, 150-156.

**Huntsberger & Ajjimarangsee (1990):** "Parallel self-organizing feature maps for unsupervised pattern recognition", International Journal of General Systems, 16, 357-372.

**Huntsberger, Kubota & Rose (1998):** "Integrated vision/control system for autonomous planetary rovers", Proc. IAPR Workshop on Machine Vision in Applications, Chiba, Japan, Nov 1998.

**Huntsberger, Mataric & Pirjanian (1999):** "Action selection within the context of a robotic colony", to appear in Proc. Sensor Fusion and Decentralized Control in Robotic Systems II, SPIE Vol. 3839, Boston, MA.

**Huntsberger & Rose (1998):** "BISMARC", Neural Networks, 11(7/8): 1497-1510.

**Mataric (1997):** ‘‘Reinforcement Learning in the Multi-Robot Domain,’’ Autonomous Robots, 4(1), 73-83.

**Pirjanian (1998):** Multiple Objective Action Selection & Behavior Fusion Using Voting, Ph.D. Dissertation, Department of Medical Informatics and Image Analysis, Aalborg Univ., Denmark, 1998.

**Pirjanian & Mataric (1999):** "Multiple Objective vs. Fuzzy Behavior Coordination", to appear in Lecture Notes in Computer Science, Fuzzy Logic in Mobile Robot Navigation.

**Rosenblatt & Payton (1989):** "A fine-grained alternative to the subsumption robot control", Proc. IEEE/INNS Joint Conf. on Neural Networks, Washington, DC, 317-324.

**Robot Colonies (1997):** Robot Colonies, (Ed. R. C. Arkin & G. A. Bekey), Kluwer Academic Publishers, Norwell, MA.

**Tyrrell (1993):** "The use of hierarchies for action selection", Journal of Adaptive Behavior, 1(4).