Ambler

An Autonomous Rover for Planetary Exploration

John Bares, Martial Hebert, Takeo Kanade, Eric Krotkov, Tom Mitchell, Reid Simmons, and William Whittaker

Carnegie Mellon University

For centuries people have been fascinated by Earth's planetary neighbors. There has been much speculation, in science and science fiction, about what lies on and under their surfaces. Despite considerable study, our knowledge remains very limited. Orbiting vehicles cannot examine internal features, and stationary vehicles, like the three Soviet and two US landers on Mars, miss what is over the horizon, atop mountains, and in ravines.

Active exploration of other planets could answer many questions about the nature and origins of our solar system. Sending astronauts or remotely controlled vehicles is a possibility, but a manned expedition is highly unlikely in the near future, and conventional teleoperation is impractical because of the long signal times (for example, up to 45 minutes for a round trip to Mars at the speed of light). A more promising approach is to launch an unmanned prospector and a vehicle to return collected samples to Earth—for instance, NASA's proposed Mars Rover and Sample Return mission.[1] The broad objectives of such a mission would be to observe and gather materials representative of the planet's geophysical, meteorological, and biological conditions and to return a varied selection of samples. Since the payload of the return vehicle is limited, the mission requires a sophisticated on-site system that can explore, assay, and select.

Extremely self-reliant, this six-legged robot will prioritize its goals and decide on its course of action as it explores the rugged terrain of a place like Mars.

Recently we initiated a research program that addresses the central robotics challenges of designing a roving explorer capable of operating with minimal external guidance. The purposes of this research are to confront issues not faced by laboratory robots, to identify and formulate the difficult problems in autonomous exploration, and to generate insights, principles, and techniques for their solutions. We are not attempting to satisfy all the requirements of a system that would be flown to another planet (for example, space-qualified processors). Instead, we are building a prototype legged rover, called the Ambler (loosely an acronym for Autonomous MoBiLe Exploration Robot), and testing it on full-scale, rugged terrain of the sort that might be encountered on the Martian suface.

To undertake an extraterrestrial prospecting mission, we must extend existing robotic technology. Because the rover will be beyond the reach of timely aid from Earth, it must exhibit extreme self-reliance. It must be able to navigate, explore, and sample, and to know, moreover, what tasks do and do not lie within its capabilities. Particular issues critical to autonomous planetary exploration include locomotion, rough-terrain navigation, sample acquisition, perception, self-awareness, task autonomy, safeguarding, and system integration. Although semiautonomous and remotely assisted systems may be practical for some tasks,[2] our

0018-9162/89/0600-0018$01.00 ©1989 IEEE

research strategy is to strive for full autonomy wherever possible, with the rover deciding when to ask for missing information.

In this article we present an overview of our research program, focusing on locomotion, perception, planning, and control. We summarize some of the most important goals and requirements of a rover design and describe how locomotion, perception, and planning systems can satisfy these requirements. Since the program is relatively young (one year old at the time of writing), this article aims to identify issues and approaches and to describe work in progress rather than to report results. Although our discussion concentrates on a Mars mission, we expect many of the technologies developed in our work to be applicable to other planetary bodies and to terrestrial concerns such as hazardous waste assessment and remediation, ocean floor exploration, and mining.

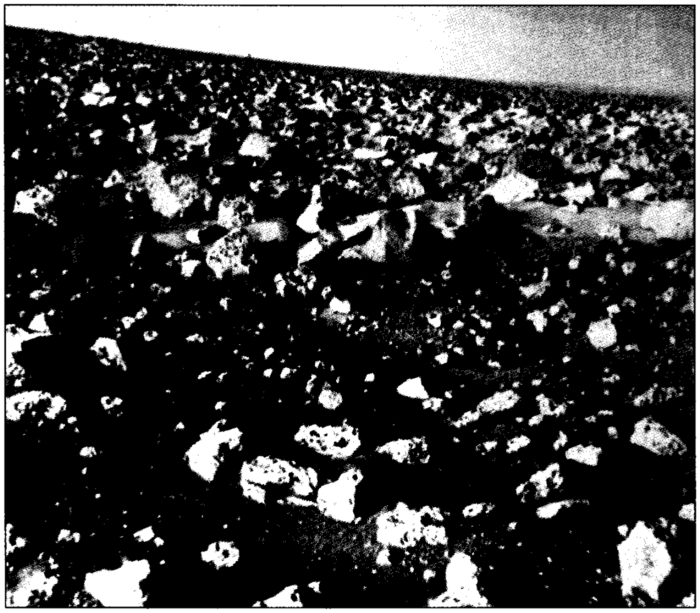

Figure 1. Martian terrain viewed by the Viking 2 lander.

Locomotion

Our data on the Martian landscape indicates that an explorer would encounter a wide variety of terrain features, including a canyon 4,800 kilometers long by 7 kilometers deep, a mountain 27 kilometers high, and numerous sand dunes, rock fields, and craters. Figure 1 illustrates the barren, rugged terrain viewed by the Viking 2 lander.

Since the locomotion system must safely transport the vehicle over vast expanses of irregular terrain, perhaps the most important design criterion for the locomotion system is traversability: it must be able to navigate over extremely rugged terrain. Specifically, the rover should be capable of traversing a one-meter step, negotiating a 60 percent slope, and maintaining an average velocity of approximately one kilometer per day (these specifications, although somewhat arbitrary, reflect plausible assumptions[1] about the scale of objects on the Martian surface and about potential missions). Energy efficiency poses an additional design constraint because total on-board power generation is expected to be less than one kilowatt. As the dominant energy consumer, the locomotion mechanism must be extremely efficient. Another design consideration is that the locomotion mechanism must provide a stable platform for sensors and sample acquisition tools.

These design criteria admit a wide variety of possible locomotion candidates, including mechanisms that roll, walk, combine rolling and walking, or perform so-called hybrid locomotion.[3] Rolling machines have wheels or tracks in continuous contact with the terrain; they propel themselves by generating traction parallel to the terrain surface. Walkers suspend themselves over the terrain on discrete contact points and maintain principally vertical contact forces throughout propulsion; this allows more tractable models of terrain interaction than are possible for wheels. In addition, walking mechanisms isolate the robot's body from the underlying terrain and propel the body along a smooth trajectory independent of surface irregularities.

After comparing these candidates (see article by Bares and Whittaker[4] for a trade-off analysis of locomotion mechanisms with respect to these constraints), we selected legged locomotion because of its superior rough-terrain traversability characteristics, its theoretical efficiency, and its ability to keep sensors and sampling equipment steady and stable.

Our initial Ambler configuration consists of six legs stacked coaxially at their shoulder joints (Figure 2). Each leg is mounted at a different elevation on the central axis of the body and can rotate fully around the body. Each leg (Figure 3) consists of two revolute joints (shoulder and elbow) that move in a horizontal plane to position the leg, and a prismatic joint at the end of the elbow link that effects a vertical telescoping motion to extend or retract the foot. Thus, the locomotor has 18 degrees of freedom. The planar reach (combined length of shoulder and elbow links) of a leg is 2.5 meters and the vertical stroke (telescopic distance) is 1 to 2 meters, depending on the position of the leg on the stack. The average overall height of the Ambler is approximately 3.5 meters, and its nominal width is approximately 3 meters. With these dimensions, the Ambler can step over obstacles 1 meter high while maintaining a level body trajectory.

The Ambler body, a cylinder one meter in diameter situated below the leg stack, will contain equipment for power generation, computing, sample analysis, and scientific instrumentation. Sample acquisition tools can be mounted on legs or on the underside of the body. Communication equipment can be mounted either

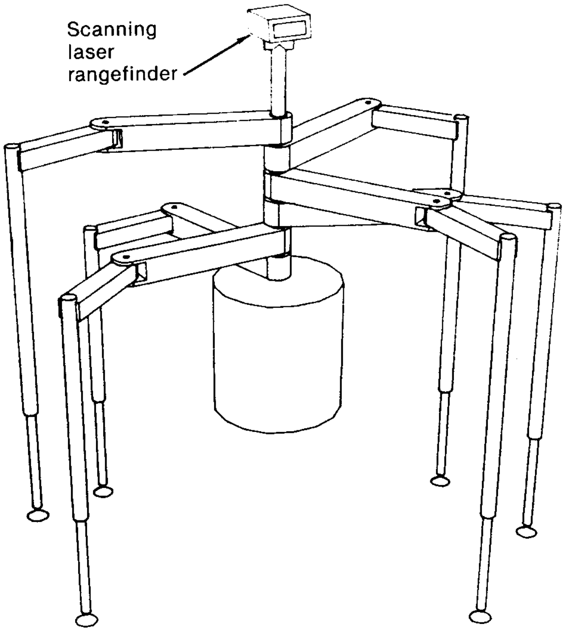

Figure 2. Sketch of the Ambler.

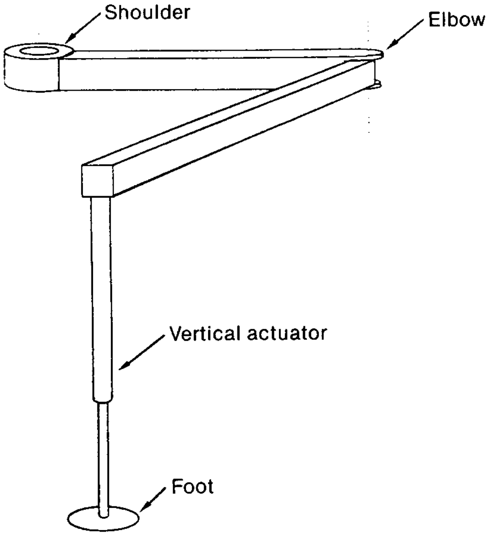

Figure 3. Side view of an Ambler leg.

above the leg stack or in the body. Perception sensors will be mounted above the leg stack, where they will have large fields of view; other high-resolution sensors can be placed under the body or directly on legs.

This configuration has a number of benefits. First, decoupling the vertical and horizontal joints simplifies walk planning and motion control by reducing complex, six-dimensional problems to smaller ones. Second, the sampling tools under the body have a clear view of and close proximity to the terrain that they must access; body movements position and orient the tools, reducing the number of degrees of freedom they require. Third, during locomotion the legs isolate the body and sensors from terrain roughness.

In operation the Ambler will walk over rugged terrain much as one poles a raft floating over a rough lake bottom. The six vertical actuators in the Ambler's legs level the body over terrain, while the planar joints propel the body. As the body advances, one leg at a time moves ahead of the walker, much as the pole is placed ahead of the raft. A unique result of the stacked leg configuration is that an overlapping gait is possible—that is, a gait in which a rear leg moves ahead, or "recovers," past forward supporting legs. Figure 4 illustrates this gait. An overlapping gait requires fewer foot placements, saving energy (energy is expended whenever a foot interacts with terrain) and reducing demands on perception and planning. While one leg is recovering, the five other legs support the body. The stability of the stance can be maximized by maintaining the center of gravity inside a conservative support polygon. Inside this region, the vehicle remains stable even if one (and possibly more) of the legs ceases to support it, either due to failure or slippage.

Experience with existing walking mechanisms (see article by McGhee[5] or Raibert[6] for an overview) suggests that they are difficult to coordinate due to their complexity, suffer large energy losses due to actuator conflict, and can be unreliable upon failure of one or more legs. We designed the Ambler to overcome these problems.

Unlike those of other walkers, the Ambler's actuator groups for body support and propulsion are orthogonal; a subset of the planar joints propels the body, while the vertical actuators support and level the body. The Ambler can level itself without propelling and propel without leveling, having no power coupling between the two motions. This should make it both easier to control and more efficient than other walking mechanisms.

The Ambler locomotor configuration is a dramatic improvement in reliability over conventional walking mechanisms. Because the legs are stacked above the body and can rotate by 2π about their shoulder joints, any leg can operate in any body sector. Thus, any functional leg can reposition itself to substitute for any failed leg, and three legs would have to fail to cause immobilization.

Perception

The Ambler needs timely and detailed perception to plan effective locomotive and sampling strategies and to monitor their execution. The perception system's task is to build and maintain represen-

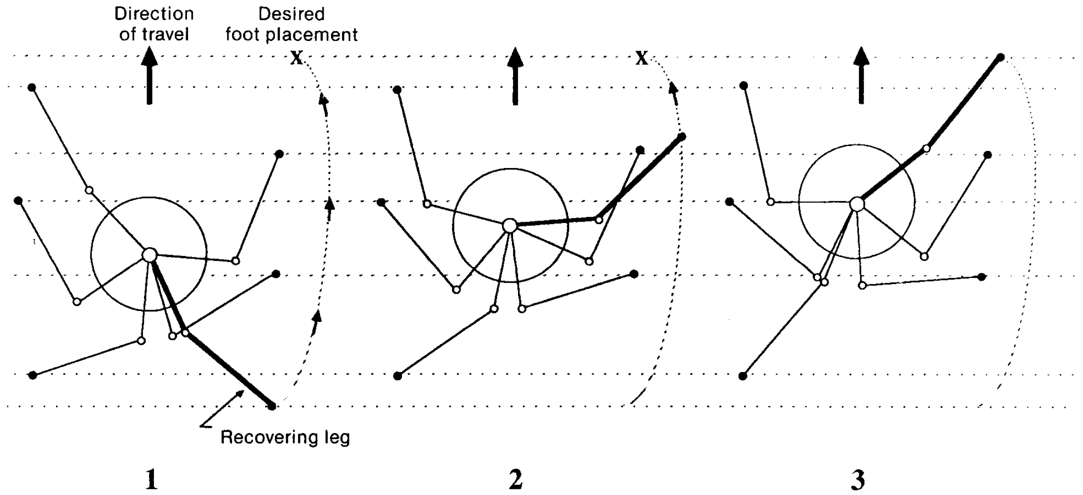

Figure 4. Overlapping gait. To advance the walker, the recovering leg (drawn thicker) overlaps the two right supporting legs. Depending on a leg's location on the central stack, some weaving around supporting legs may be necessary for it to move past forward legs.

tations—which we call terrain maps—of terrain (geometry, soil type) and discrete objects (size, shape).

Perceiving and mapping rugged, outdoor terrain are significant challenges. Current machine perception techniques can be applied with some success to man-made, structured, indoor scenes. Unlike industrial systems, however, the rover will have little need to recognize or describe regular geometric shapes and will not be able to capitalize on the powerful features (such as symmetry, smoothness, constant illumination) utilized to perceive worlds consisting of blocks. We must develop new techniques for constructing maps of natural, unstructured, outdoor environments.

The problem of building and maintaining those maps raises several issues: (1) representation of data at different levels of resolution, (2) construction of maps and descriptions from different sensors, and (3) effective use of the maps.

Representation. The perception system must provide an environmental representation that is appropriate for a wide variety of tasks, each with different requirements. For example, locomotion and sampling require detailed, local representations, while navigation and mission planning demand broad, global descriptions. To uniformly accommodate these diverse needs, we have selected a hierarchical representation scheme that describes terrain and objects at varying levels of resolution. At each level of resolution we describe the environment in two ways: on a geometric grid and as object descriptions. Together these comprise a terrain map at one scale.

We define an elevation map on a regular grid. Each grid square records information about the terrain in that area—for instance, its elevation above a ground plane. Other terrain attributes include the following: the uncertainty of the estimated elevation; roughness; slope; labels indicating whether the terrain is unknown (has never been observed) or occluded (currently not observed because it lies in a shadow cast by another object); mineralogical composition; and a measure of traversability derived from slope, roughness, and other properties.

Object descriptions include the size, shape, and location of particular objects such as a boulder; symbolic terrain descriptions, such as hill, valley, saddle, and ridge, which may be useful for identifying promising routes or sample sites; paths the vehicle has followed; locations that have been sampled; and viewpoints from which observations have been made.

Constructing terrain maps. Constructing terrain maps requires sensing and interpretation, ranging from low-level data collection to high-level scene modeling. This section first focuses on the lowest level of abstraction (sensors) and then describes an intermediate level of abstraction (local surface geometry). It does not address the highest level of object identification and semantic interpretation.

We will equip the Ambler with a battery of different sensors to collect multispectral data. Our primary sensor is a scanning laser rangefinder that measures both reflectance and range.[7] The scanner directly recovers the environment's three-dimensional structure, operating more rapidly and reliably than other vision tech-

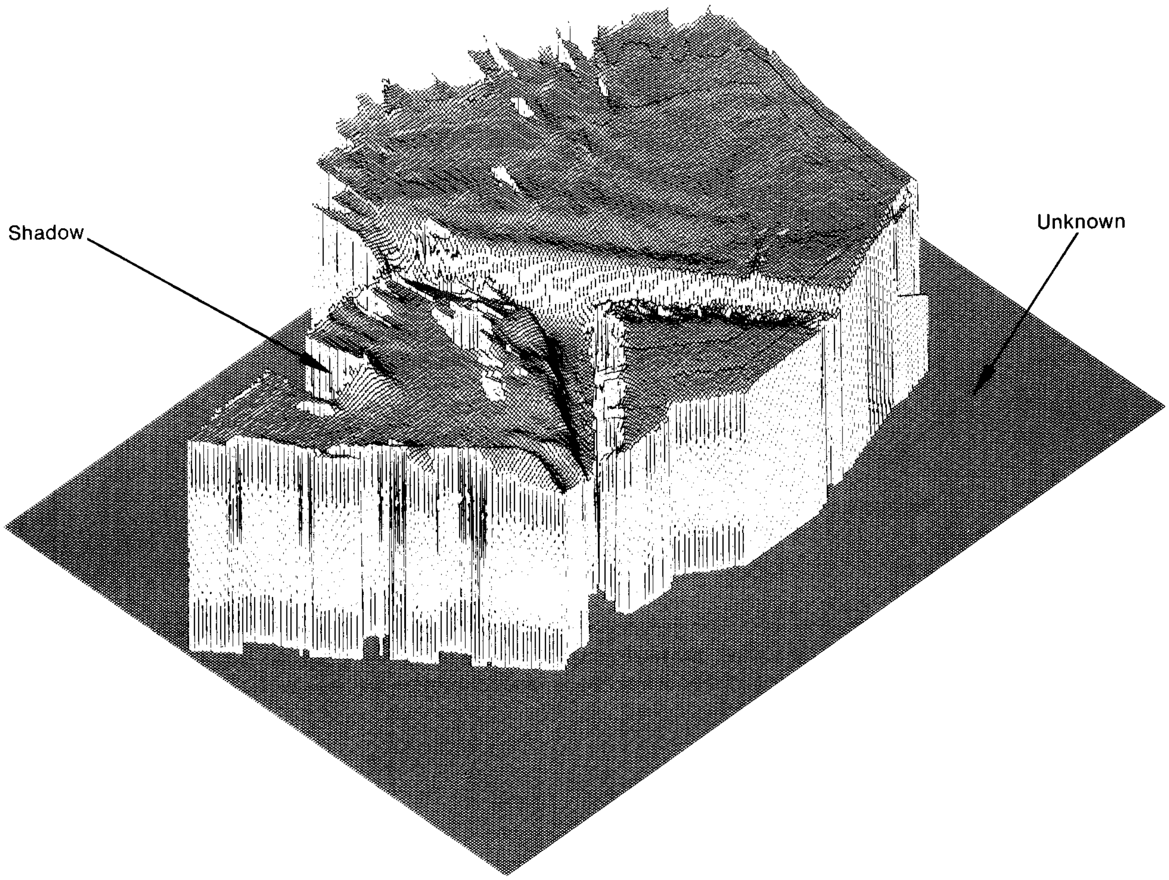

Figure 5. A composite elevation map constructed by merging four rangefinder views of rugged terrain at a construction site. The grid size is 10 centimeters.

niques such as binocular stereo and motion. In the near future we will also use a pair of color cameras for determining material properties from color and texture, for long-range viewing, and for stereo viewing to back up the rangefinder. We also plan to incorporate an inertial reference sensor, inclinometers, tactile sensors on sampling tools, and other imaging devices. However, the perception system need not be limited to passively interpreting data; it can actively use the Ambler vehicle itself as a sensor to determine soil cohesion and friction parameters either directly, by measuring leg joint torques while walking, or indirectly, by comparing the soil in its footprints to nearby soil.

As an example of how the perception system operates, we will consider the interpretation of rangefinder signals in terms of local surface geometry (interested readers can find details in a technical report by Hebert, Kanade, and Kweon[7]). This involves creating an elevation map from a range image, matching it to another elevation map, and merging the two maps to form a composite map.

To construct elevation maps from range images, we have developed an algorithm that operates at arbitrary resolutions. It computes the locations where rays emitted by the sensor strike the terrain, and then it refers the intersection points and an estimate of their uncertainty to a reference grid, thus creating an elevation map.

To merge elevation maps from successive viewpoints, we have developed a two-stage algorithm to determine the correspondence between two elevation maps. The first stage matches a sparse set of geometric features extracted from the two maps, whose output is the estimated rigid transformation T relating the two sets of features. The second stage takes T as an initial estimate and refines it by gradient descent, iteratively minimizing an error functional defined over all the data points in the two maps. Once we know T, we can apply it to merge the maps; Figure 5 illustrates a composite elevation map constructed by merging four rangefinder views of the rugged terrain at a construction site.

Using terrain maps. Once maps and descriptions of the environment have been constructed, the perception system must support and facilitate their use for a variety of tasks. For locomotion the Ambler will access elevation maps to select footfall locations that can accommodate its feet and support its mass. For navigation the Ambler will use the elevation maps to plan paths and to localize itself from landmarks. For sample acquisition the Ambler will use both elevation maps and object descriptions: the former to identify promising sampling sites based on topographic features; the latter to identify objects to be sampled, determine approach directions, and select control regimes (such as force and position).

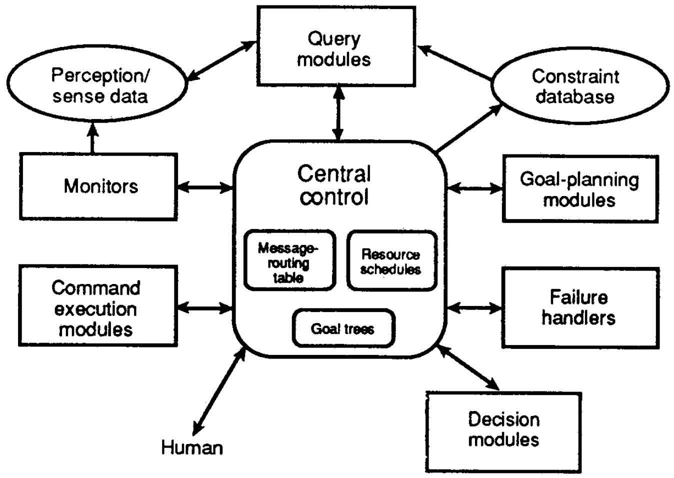

Figure 6. Task control architecture.

Planning and control

We are designing and implementing a general robot architecture that addresses three important planning and control issues: (1) coordination of multiple modules and multiple goals, (2) flexibility in handling plan failures and contingencies, and (3) self-awareness of the robot's own capabilities and limitations.

A basic coordination problem is the integration of different modules. For practical reasons of efficiency and ease of implementation, the various components will use different algorithms, representations, and even languages. For example, the route planner (the software that plans routes) may use geometric algorithms, while the mission planner might use symbolic techniques. The architecture must facilitate the exchange of data between modules and flexible yet efficient control flow management.

Another important coordination issue is the handling of multiple goals. The Ambler will have multiple, often conflicting, goals, such as navigation, science, and health goals. The control architecture must prioritize and schedule goal achievement on the basis of a cost-benefit analysis that accounts for factors such as the environment, the robot's capabilities, and its past actions. For example, if the Ambler detects a potential sedimentary rock while traveling, it might decide to detour to obtain a sample if the rock is close, if the robot has no pressing deadlines, and if the robot has not already acquired sufficient sedimentary samples.

The issue of flexibility concerns how well the robot reacts to changes in its uncertain, dynamic environment. The robot must notice indications that its plans are failing, are no longer applicable, that there are unexpected contingencies, or even unexpected opportunities. The issue is complicated because, practically, the robot cannot perceive everything. Instead it must focus on those aspects it deems most important, deploying monitors to check on specified conditions.

The robot also needs flexibility in deciding how to handle problems indicated by the changes it detects. Instead of merely passing an error up the goal tree, for instance, the robot might try re-achieving the goal (for example, chipping a rock again), adding a new subgoal (detouring around an obstacle), or even attending to a different goal altogether. Once the robot decides how to handle a problem, however, the recovery processes can utilize the same algorithms that created the robot's initial plans. For example, once deciding to add a detour, the robot can use its path planner, treating the detour as if it had been planned from the start.

The issue of self-awareness is particularly important for planetary exploration robots, since they are remote from human assistance and must be largely responsible for monitoring their environment and choosing acceptable actions. The robot needs knowledge of its resources, capabilities, and limitations to make intelligent decisions about its course of action. It needs to schedule limited resources and to prioritize conflicting goals according to their relative costs and benefits. In addition, the robot should be aware of deadlines for its goals, the expected reliability of its planners, and the time necessary to execute its plans.[8]

Although our robot control architecture is being designed to provide many tools for constructing a planetary explorer, it will be open enough for easy experimentation and extension. The architecture is a distributed system with centralized control. It supports modules running on separate machines in different languages (currently, C and Lisp), communicating via coarse-grained message passing. Building on ideas from the Navlab project,[9] communication and data transfer are totally transparent to the individual modules.

Whereas planning, sensing, and actuation are distributed, control of when and how to attend to goals is centralized (Figure 6). The central control receives messages from other modules and routes them to the appropriate module to be handled. The control module also schedules the available computational and physical resources and maintains goal trees to help in error recovery and decision making.

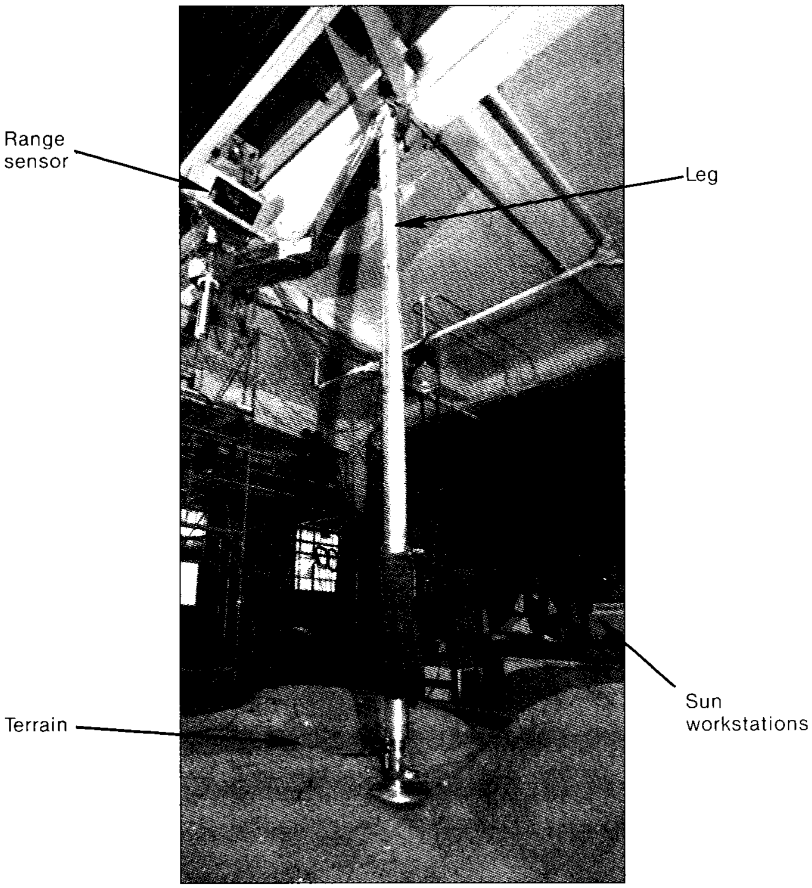

Figure 7. Single-leg testbed.

Although centralized control has the potential of being a bottleneck, we believe that global control is crucial for robots that have to contend with multiple goals and limited resources. At the same time, we are examining how to combine centralized control architectures with architectures that provide fast, reflexive actions (see Brooks,[10] for example).

The architecture directly supports different classes of messages for constructing robot systems. Query messages, which are used to access perceptual and internal sense data, return information and control to the requesting module. Goal messages, which plan actions by issuing other goal or command messages, are nonblocking, so that control returns before the subgoals are actually achieved. This nonblocking feature facilitates error recovery and the coordination of multiple tasks, since the central control can decide when to suspend or achieve goals.

Other message classes include (1) command messages, nonblocking messages used to control actuators, (2) constraint messages, used to add information to a database, (3) monitor messages, which check on the status of the robot or its environment, (4) decision messages, used to prioritize goals, and (5) failure messages, which indicate that either a plan-time or execution-time error has been detected.

Although the Ambler is designed to be largely autonomous, it is important that humans can intervene and operate it remotely when the situation warrants. The architecture meets this need by allowing humans to send or receive messages, at any level of abstraction, just as any other module can. This feature helps in development of the Ambler because it lets us substitute human input for as-yet-unwritten modules.

Experimental testbeds

Although simulations are often useful abstractions of the problems an autonomous robot will face, they are never as revealing as operating the actual mechanism. Our philosophy is to embed our ideas in working mechanisms that operate in natural environments. While we build the six-legged vehicle described here, we are testing our locomotion, perception, planning, and control ideas on two operational testbeds.

The first testbed is a one-legged version of the Ambler (Figure 7). It enables us to begin integrating the component technologies into a complete but simplified system that can demonstrate single-leg walking, using a few frames of range data, simple walk planning, and simple error recovery. A full-scale leg has been built and mounted on a carriage that travels along rails on the ceiling to simulate body motion. A scanning laser rangefinder mounted above the leg provides data for building terrain maps. A large sandbox under the testbed contains different soil types and obstacles. A rudimentary version of the planning and control architecture, running on Sun workstations, enables the leg to lift, recover, and land at a chosen position. We are also using this testbed to study foot-terrain interactions such as foot slippage and sinkage, and power consumption during footlift and footfall.

The other testbed is a Heathkit Hero 2000, a commercially available robot, which we are using to explore ideas for combining navigation and sample collection in an indoor environment. The Hero mechanism is wheeled and has an arm and gripper. For perception it has a base sonar and a rotating head sonar, and we have added a sonar on its wrist. In addition, a camera mounted in the ceiling of our lab gives the Hero a global overhead view. Using its vision system, the Hero plans paths to objects, and, once in the vicinity of an object, uses its sonars to locate and grasp the object. At present the Hero picks up and deposits plastic cups and cans, and soon we expect it to be able to retrieve printer output and to schedule the achievement of multiple, conflicting goals.

Constraints inherent in the task of autonomously exploring another planet have driven our design of a robotic rover. In particular, the task demands a system that efficiently and reliably navigates over rough terrain, reliably perceives rugged terrain and irregularly shaped objects, and exhibits extreme self-reliance in performing a multitude of tasks. We have incorporated these constraints into our design for the Ambler. By pursuing the issues of locomotion, perception, planning, and control in the design of a robot that operates on another planet, we hope to find solutions for the problems involved in sending intelligent machines to Mars and beyond. □

Acknowledgments

We would like to acknowledge the contributions of the entire Ambler research team at Carnegie Mellon University, whose collective efforts have produced the ideas reported here.

This research was sponsored by NASA under contract NAGW-1175. The views and conclusions contained in this article are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of NASA or the United States government.

References

1. J.R. Randolph, ed., *Mars Rover 1996 Mission Concept*, Technical Report D-3922, Jet Propulsion Laboratory, Pasadena, Calif., 1986.
2. B. Wilcox and D. Gennery, "A Mars Rover for the 1990's," *J. British Interplanetary Society*, Vol. 40, 1987, pp. 484-488.
3. D.D. Wright and R.E. Watson, "Comparison of Mobility System Concepts for a Mars Rover," *Proc. SPIE Conf. Mobile Robots II*, Vol. 852, Society of Photo-Optical Instrumentation Engineers, Cambridge, Mass., Nov. 1987, pp. 180-187.
4. J. Bares and W. Whittaker, "Configuration of an Autonomous Robot for Mars Exploration," *Proc. World Robotics Conf.*, Society of Mechanical Engineers, Gaithersburg, Md., May 1989.
5. R. McGhee, "Vehicular Legged Locomotion," in *Advances in Automation and Robotics*, G.N. Saridis, ed., Jai Press, Greenwich, Conn., 1985.
6. M. Raibert, Introduction to Special Issue on Legged Locomotion, *Int'l J. Robotics Research*, Vol. 3, No. 2, Summer 1984, pp. 2-3.
7. M. Hebert, T. Kanade, and I. Kweon, *3-D Vision Techniques for Autonomous Vehicles*, Technical Report CMU-RI-TR-88-12, Robotics Institute, Carnegie Mellon University, 1988.
8. T. Dean and M. Boddy, "An Analysis of Time-Dependent Planning," *Proc. Seventh Nat'l Conf. Artificial Intelligence*, St. Paul, Minn., 1988, pp. 49-54.
9. C. Thorpe et al., "Vision and Navigation for the Carnegie Mellon NAVLAB," *Annual Review of Computer Science*, Vol. 2, 1987, pp. 521-556.
10. R.A. Brooks, "A Robust Layered Control System for a Mobile Robot," *IEEE J. Robotics and Automation*, Vol. RA-2, No. 1, Mar. 1986, pp. 14-23.

John Bares is pursuing a PhD degree at Carnegie Mellon, researching configuration techniques for autonomous, rugged-terrain walking machines, including the Ambler. His research interests include configuration, simulation, design, and implementation of teleoperated and autonomous field robots.

Bares received the BS and MS degrees in civil engineering from Carnegie Mellon University in 1985 and 1987, respectively.

Martial Hebert is a research scientist at the Robotics Institute at Carnegie Mellon University. His research focuses on the range data in computer vision for object recognition and autonomous vehicle navigation. He is involved in the development of a 3-D vision system for the Navlab, an autonomous vehicle for road and cross-country navigation, as well as in the Ambler project.

Hebert received the doctorate in computer science from the University of Orsay, France, in 1984. He is a member of the IEEE Computer Society.

Takeo Kanade is a professor of computer science and codirector of the Robotics Institute at Carnegie Mellon University. Before joining CMU in 1980, he was an associate professor of information science at Kyoto University, Japan.

Kanade has worked on various problems in vision, sensors, manipulators, and mobile robots. He has written and edited three books and written more than 70 papers in these areas. He is the principal investigator in three robotics research programs at CMU. He also chairs CMU's newly established robotics PhD program.

Kanade received his PhD in information science from Kyoto University in 1974. He is a member of the IEEE Computer Society. He served as general chair of the 1983 IEEE International Conference on Computer Vision and Pattern Recognition and vice chair of the 1986 IEEE International Conference on Robotics and Automation. He is the editor of the *International Journal of Computer Vision*.

Eric Krotkov is a research scientist studying machine perception and mobile robots at the Robotics Institute of Carnegie Mellon University. Before joining CMU in 1988, he worked as a postdoctoral fellow in the robotics group of the Laboratorie d'Automatique et d'Analyse des Systemes, Toulouse, France. Krotkov has worked in several areas of robotics and artificial intelligence: three-dimensional computer vision, mobile robot systems, and philosophy of perception.

Krotkov received his BA in philosophy from Haverford College in 1982 and his PhD in computer and information science from the University of Pennsylvania in 1987. He is a member of the ACM and the IEEE Computer Society.

Tom Mitchell is a professor of computer science at Carnegie Mellon University and an affiliated faculty member of the Robotics Institute. He taught in the Computer Science Department at Rutgers University from 1978 until moving to Carnegie Mellon in 1986. His current research focuses on robots that learn and general architectures for problem solving and learning.

Mitchell earned his BS degree in 1973 from MIT and his MS and PhD degrees from Stanford University in 1975 and 1978, respectively. In 1983 he received the IJCAI Computers and Thought Award in recognition of his research in machine learning, and in 1984 an NSF Presidential Young Investigator Award.

Reid Simmons is a research associate in the School of Computer Science at Carnegie Mellon University. His doctoral thesis developed and analyzed techniques for combining associational and causal reasoning for planning and interpretation tasks. His current research is in robot control architectures that can handle multiple, conflicting tasks in uncertain and changing environments.

Simmons earned his BA degree in 1979 from SUNY at Buffalo and his MS and PhD degrees in artificial intelligence from MIT in 1983 and 1988, respectively. He is a member of ACM, AAAI, and Sigma Xi.

William Whittaker is a senior research scientist with the Robotics Institute at Carnegie Mellon and a senior lecturer in the university's Department of Civil Engineering. He is also director of the Field Robotics Center. His research interests center on mobile robots in unpredictable environments and include computer architectures to control mobile robots, modeling and planning for nonrepetitive tasks, complex problems of objective sensing in random or dynamic environments, and integrations of complete field robot systems.

Whittaker received his BS from Princeton in 1973 and his MS and PhD from Carnegie Mellon in 1975 and 1979, respectively. He received Carnegie Mellon's Teare Award for Teaching Excellence. *Science Digest* named him one of the top 100 US innovators in 1985 for his work in robotics.

Readers may write the authors at the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213.

When it comes to choosing compilers, MetaWare is the right choice!

Developing the next generation of software products is serious business. You need the best tools to produce the best code. You need **MetaWare**.

Superior Compilers

A compiler that handles large programs with ease, while producing the expected results, is the key to developing the highest quality applications. Many of **MetaWare's** customers say that **High C** ™ and **Professional Pascal** ™ are the highest quality compilers in the industry. They are reliable, and well documented, and their superior diagnostic messages help to produce better products more quickly. **There are no surprises with MetaWare compilers.**

Compiler Features

ANSI Standard with extensions • Generates small, fast executables • Support of 80x87, Weitek 1167/3167, 68881, and Am29027 math co-processors • Global common subexpression elimination • Live / dead code analysis • Constant propagation, copy propagation • Tail merging (cross jumping) • And many more!

Multiple Platform Support

MetaWare uses its own **Translator Writing System** to create all of its compiler products. Common components and library functions are shared across the product line. Improvements to compilers are quickly realized on all platforms.

MetaWare, High C and Professional Pascal are trademarks of MetaWare Incorporated; UNIX is a trademark of AT&T; Other products mentioned are trademarks of the respective companies indicated.

Supported Platforms (re: Dhrystones)

• Sun-3	— >50% > resident compiler.
• Sun 386*i*	— >50% > resident compiler.
• Sun-4	— >25% > resident compiler.
• PC: DOS, OS/2—	3-10% > Microsoft C; 30%>MS Pascal, Lattice C.
• 386 32-bit DOS—	No real competition.
• 286, 386 UNIX—	66% > than pcc on 386.
• VAX VMS	— ≈ DEC's excellent C and Pascal; Host for cross compilers and TWS, not Native.
• VAX Ultrix	— 19% > pcc on Dhrystone; much > Berkeley Pascal.
• RT PC	— 93% > 4.3bsd port of pcc.
• AIX/370	— Much better than any 370 C and VS Pascal.
• AMD 29K	— >40,000 Dhrystones.
• Intel i860	— >70,000 Dhrystones at 33 MHz.

So when it comes to selecting a compiler company, **you need the best!**

Can you really afford anything less?

MetaWare™
INCORPORATED

The Compiler Products for
Professional Software Developers

2161 Delaware Avenue • Santa Cruz, CA 95060-5706
Phone: (408) 429-6382 • FAX: (408) 429-9273

Reader Service Number 4