MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

Working Paper 231 December 1982

Aspects of the Rover Problem

Richard J. Doyle

Abstract: The basic task of a rover is to move about automonously in an unknown environment. A working rover must have the following three subsystems which interact in various ways: 1) locomotion--the ability to move, 2) perception--the ability to determine the three-dimensional structure of the environment, and 3) navigation--the ability to negotiate the environment. This paper will elucidate the nature of the problems in these three areas and survey approaches to solving them while paying attention to real-world issues.

INTRODUCTION

Why Rovers?

A rover is a mobile vehicle which has to fend for itself. Research into rovers has been motivated by a variety of scenarios that identify the need for a mobile robot which can operate with a high degree of autonomy.

Perhaps the most exciting of these scenarios is planetary exploration. The successful Viking landings on Mars in 1976 returned the first images of that planet's surface and generated excitement both inside and outside the space community. Yet almost anyone who sees one of these images is struck by a yearning to know what lies behind that rock or beyond that hill or over the horizon. Despite all that was learned, the sampling of the Martian environment provided by these missions remains frustratingly small, and not guaranteed to be representative. The next step is, of course, a mobile lander. Such a vehicle on the surface of Mars cannot be remotely controlled from the earth however, because of the finite speed of light. One-way communication time between earth and Mars averages about ten minutes. An image showing the vehicle approaching a crevasse will not be of much use because the frantic command to stop issued from ground control will not be received by the vehicle for twenty minutes. Sometime before then the signal will disappear forever. The vehicle must be able to perceive and avoid obstacles on its own.

A more "down-to-earth" scenario is an aquatic robot. Both general exploration of the largely unknown undersea environment and mining of the ocean floor require a mobile submersible. Once again, remote control is impractical because water severely attenuates signals over long distances. An aquatic robot must also be autonomous.

Once the technology is available, rovers are likely to be used in applications where remote control is at least possible, but the use of an autonomous mobile robot is much more attractive. Removing the overhead of human operators and maintaining a constant communication link is desirable as long as the continued functionability and survivability of the remote vehicle can be assured. These rovers will be useful for performing tasks in environments that are hostile to humans. Some examples are geological and petrological investigations in deserts and the polar areas.

Rover technology will be extended in the more distant future to machines of greater sophistication which will be used for more everyday applications in environments of greater complexity. Household servants, robots to perform maintenance of nuclear power plants, security guard robots, and automatic cars are examples of distant benefits of rover research.

A Representative Sample

The following sampling of rover research efforts provides a brief history of the progress that has been made. They will be referenced often in the ensuing discussions.

One of the first successful long-running rover experiments was performed at the Stanford Research

Institute under the direction of Nils Nilsson and was known as SHAKEY [Nil69]. SHAKEY helped to define the problem areas more concisely and paved the way to further research.

Hans Moravec of Stanford University inherited a much-maligned mobile cart which had made the rounds from Mechanical Engineering through Electrical Engineering to Stanford's Artificial Intelligence Laboratory. Moravec decided to use this cart for his Ph.D. work and his bold effort at taking the rover problem by the horns provided valuable results and helped to define further the areas in which more research was indicated [Mor80]. Moravec is now at the CMU Robotics Institute and is continuing his research with a new rover [Mor82].

Georges Giralt, Raja Chatila, and others at the Laboratoire d'Automatique et d'Analyse des Systemes in Toulouse, France are working with a mobile robot known as HILARE (Heuristiques Integrees au Logiciel et aux Automatismes dans un Robot Evolutif) [Cha81]. This research effort is important because the problem of navigating in an unknown environment is being dealt with in depth.

Research towards realizing a rover for the exploration of Mars was performed at the Jet Propulsion Laboratory after the Viking landings [Tho77]. Because of the specific application, this rover research effort, more so than any other, was heavily motivated by real-world implementation issues[JPL77]--even though the "real world" was Mars!

The Rover Scenario

A rover is an autonomous vehicle. Clearly the need for a locomotion system is implicit in the idea of a rover. Furthermore, because the environment in which a rover will operate may be partially or completely unknown, a means of creating and updating a model of the environment is necessary. A vision system will provide this capability. Finally, a rover will use the information provided by this model to plan its movements to reach goals--to navigate. These three capabilities--locomotion, perception, and navigation--are the essential capabilities of a rover.

1. LOCOMOTION

Overview

There are several general issues involved in the design of any locomotion system. These include stability, robustness, traction, flexibility, and maneuverability. The following analyses of different locomotion system designs will be performed at a rather high level in the context of these five issues. The particulars of the drive systems and similar low-level issues will not be discussed in great detail.

Design Issues

Stability is the single most important attribute of any locomotion system. Allowing the locomotion effectors to become irrevocably decoupled from the environment is fatal. A convenient measure of stability is the location of the center of gravity of the rover. Ideally, it should be close to the physical center of the rover, low to the ground, and should not shift very much as the rover moves.

Robustness refers to the ability of the locomotion system to handle minor obstacles such as small localized surface undulations and small discrete objects. Closely related to robustness is *traction*--the ability of the locomotion system to handle moderate changes in grade and to gain purchase on loose or smooth surfaces. It is desirable for a rover not to have to worry about every gully or pebble it encounters and it should not have to skirt gentle hills, valleys, or craters. If the locomotion system can confidently handle these minor hazards then they can be effectively ignored as obstacles and the path-finding task of the navigation system is made that much easier. The choice of locomotion effectors is an important factor in building a locomotion system that is both robust and provides a high degree of traction.

Flexibility refers to the ability of the locomotion system to negotiate a cluttered environment. It represents an approach to handling minor obstacles which is different from that taken by a robust system. Rather than driving straight over the obstacles, a flexible system carefully picks its way through them. A highly flexible system provides greater ground clearance for the body of the rover and has smaller effectors whose positions relative to each other may not be fixed. The total contact area with the ground is smaller and the effectors are placed to avoid the obstacles. The smaller effectors may pose a problem on soft surfaces because the weight of the rover is not as well distributed. Both the robust and flexible approaches to handling minor obstacles raise the issue of stability. Driving over obstacles or shifting the effectors to avoid them both have a potential for upsetting the rover. Robustness and flexibility are dual attributes of a locomotion system--the existence of one generally implies the absence of the other.

Maneuverability refers to the ability of the locomotion system to change the direction of the rover's motion easily and quickly. Turning radius while in motion, and the abilities to turn in place and operate in reverse all contribute to the maneuverability of a locomotion system. This attribute is important when the rover is trying to struggle through a tight passage or worse yet, trying to back out of such a passage which proved to be impassable. The maneuverability provided by a locomotion system may be assessed by imagining trying to parallel-park a rover so equipped.

Wheels

Wheels were among the first artificial locomotion systems invented and remain today the simplest and most commonly implemented locomotion system for any type of vehicle. The drive mechanisms for wheels are generally less complicated than those required by other locomotion systems.

For a rover, both four and six-wheeled systems may be considered. (The Soviet Lunokhod, a remotely controlled vehicle used on the lunar surface (round-trip communication time to the moon is only about 2.5 seconds) used eight wheels, but it is not clear what advantage, if any, was accrued by having the extra two wheels). The basic four-wheel locomotion system performs excellently on the open road, but because it does not possess a great amount of either robustness or flexibility, its usefulness to a rover on jumbled terrain is limited. After running a wheel into a rock of moderate-size, the rover will most likely have no choice but to back off, change direction slightly, and try again.

Both robustness and traction may be boosted by using balloon tires and four-wheel drive. Independent suspension can be used to maintain stability while the rover drives over obstacles.

Whatever flexibility is inherent in this system is provided by ground clearance alone, for the positions of the wheels cannot be changed. By maneuvering carefully, the rover may pass over a small obstacle--all wheels passing on the side--instead of skirting it.

Maneuverability in a wheeled system can be achieved by driving the four wheels separately. In the case of a six-wheeled system, pivoting of the entire rover can be easily accomplished by driving only the two center wheels in opposite directions. Of course, having a seperate drive mechanism for each wheel is not only more expensive, but it also increases the control burden.

Mostly because of their simplicity, wheels are employed as the locomotion system for SHAKEY, Moravec's cart, HILARE, and the prototype JPL robot.

Treads

Treads indicate a commitment in the design of a locomotion system to robustness over flexibility as a means of handling minor obstacles. Obstacles whose size is a significant percentage of the rover's own size can be handled with ease by treads. Some thought must be given to a form of suspension which keeps the rest of the rover stable while a tread is crawling over an obstacle. The drive mechanisms for treads are more complex than for wheels--they must not only rotate the caterpillar tracks but also maintain high tension on them.

When an obstacle is encountered over which the treads cannot simply be driven, the treaded system's lack of maneuverability becomes obvious. Steering is achieved by driving the tracks in opposite directions and is imprecise. The overall motion is rather crude and quick starts and stops are difficult, making small maneuvers in tight spaces virtually impossible.

Treads provide the quintessence of robustness in a locomotion system. However, the treaded system sacrifices most everything else to achieve this robustness and the overall effectiveness of the resulting locomotion system is questionable.

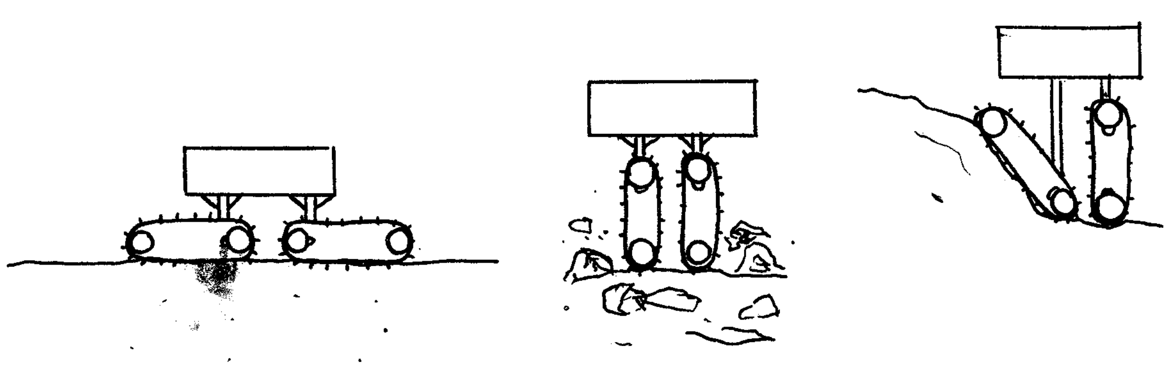
Hybrids and Extensions

Another way to approach the design of a locomotion system is to consider the different advantages provided by wheels and treads and try to combine them. One such system has been designed and a prototype built by Lockheed on contract to the Jet Propulsion Laboratory for consideration as a locomotion system to be used on a Martian rover. The system uses "loopwheels", which are an interesting cross between treads and wheels. Each loopwheel is essentially a seperately mounted minitread able to pivot about an axle. With the aid of independent suspension to maintain stability, loopwheels provide robustness while removing some of the unwieldiness of a fully treaded system. A loopwheel can climb up the side of an obstacle, flop over the top while pivoting on its axle, and continue down the other side, while hardly upsetting the rest of the rover at all (see **Fig. 1**). Loopwheels combine the robustness of a treaded system with the higher maneuverability of a wheeled system.

Another interesting design extends the idea of using loopwheels even further [Koh78]. Instead of attaching the axle at the center of the minitread, it is attached at the end nearest the center of the rover. Pivoting now allows the loopwheels to take on a whole range of different configurations. When they are completely drawn up, the system becomes essentially a wheeled system. Intermediate configurations allow inclines to be traversed while keeping the body of the rover level (see **Fig. 2**). The ability to draw the treads up off the ground allows this locomotion system to lower the total contact area with the surface, imparting a degree of flexibility which the basic loopwheel system does not share. This capability can be used to assist the rover in squeezing through tight places. This design is currently implemented in a vehicle which is remotely controlled. An on-board control system for this design would have to be fairly complex to be able to realize the full range of configurations. Some means of deciding what configuration is appropriate for a given situation is needed as well.

Fig. 1. Loopwheels.

Fig. 2. Another loopwheel design.



The CMU rover uses three independently steerable wheel assemblies to achieve a remarkable degree of maneuverability [Mor82]. Motion in any direction is possible and the control algorithm is sufficiently sophisticated to permit independent control of the rover's rotation about its own vertical axis. For a constant motor speed, the faster the rover rotates, the slower its forward motion. This unorthodox capability to "gear down" gives this locomotion system a variable degree of robustness to permit the rover to climb grades.

Legs

Legged systems are the most sophisticated, and also the most interesting of the locomotion systems considered in this discussion. Walking is really controlled falling. The legged system moves by falling forward and then reestablishing stability before overbalancing occurs by moving a leg (or legs) forward. This use of gravity to power locomotion is peculiar to legged systems and results in a a dynamic form of stability which actually works very well. Even the human bipedal system is remarkably stable, despite the rather high location of the center of gravity and the fact that only one foot is on the ground most of the time during walking [GHM74].

The human system uses both compliance and some remarkable reflexes for recovery if stability is challenged when a foot does not come down in the expected place. Systems with more legs can generally rely on compliance to permit settling should a misfall occur. Of course, a rover should know as accurately as possible the positions of the obstacles it is stepping around, to help prevent misfalls from occurring at all.

Robert McGhee, David Orin, and others at Ohio State University have been investigating legged locomotion with a six-legged walking machine known as the OSU Hexapod. The drive mechanisms needed in a legged system are radically different from those used in a wheeled or treaded system. The OSU Hexapod uses manipulator-like limbs. The control mechanism has to be quite complex to provide the capability for a variety of gaits [McS74]. Open, graded, and cluttered terrain all indicate different walking patterns. Besides these gaits, which may be precomputed in the control program of the locomotion system, there should be a capability to drive each of the legs seperately, to take full advantage of the flexibility and maneuverability that a legged system offers [McP74][McG77][Ori82].

Having legs which can operate independently allows the rover to pick its way through jumbled terrain, placing its footpads wherever obstacles are not present, within the limits of stability. A legged rover, supported by a perception system which accurately determines the locations of obstacles, can boldly negotiate terrains which would defy other locomotion systems [OMJ76][McI79].

Legs also provide a great degree of maneuverability. Tight turns, pivoting, and "shuffling" to squeeze through tight places are all feasible, as is backing up. Again, sophisticated control mechanisms are required to execute these maneuvers.

Deciding the number of legs to use involves considering some trade-offs. More legs means better overall stability and a larger variety of gaits, but it also means increased cost, weight, and control burden, finding more places to put footpads down, and slower speed.

Marc Raibert of CMU has actually built a one-legged hopping machine [Rai81]. At first glance, the problem of stability in a one-legged locomotion system seems too formidable, but Raibert has solved this problem impressively. First he constrained the hopper to motion in a vertical plane, so that the danger of instability was present only in the forward and backward directions. He has built a control mechanism that not only maintains stability but also resists perturbations (pushes) while the machine is hopping in place. Raibert is now dealing with the problem of stability for unconstrained, three-dimensional motion of the hopper. With the stability problem under control, a hopper could be a locomotion system with a high degree of flexibility. One could imagine it hurdling obstacles and hopping between small clear spaces. Raibert definitely is not proposing the hopper as a serious locomotion system, but rather as a laboratory for investigating the issue of dynamic stability.

Wrap-up

Implicit in the above discussion was the assumption that the rover being considered was operating on the ground, whether it be the surface of the planet Mars or the indoor environment of a laboratory. An antarctic rover may have to use skis, but the same design issues come into play. An aquatic rover presents a somewhat different scenario for there is no actual surface except when on the ocean floor. In the final analysis, the design of any locomotion system has to be weighed in the context of the environment in which the rover will operate.

Simple wheeled systems are likely to prove inadequate unless the rover spends its entire life indoors or on paved roads. However, wheeled systems are easy to construct and if speed is a consideration, they are the best choice. Treads or one of the loopwheel designs are the indicated choice for off-the-road terrain such as desert, or ice, or the ocean floor. But the most versatile locomotion system--which can handle any terrain the other designs can and some they cannot--is the legged system. Unfortunately, because they are the least understood and most difficult to build, there are no automatically controlled legged systems today in a very workable state. This is the area which needs the most serious research effort.

2. PERCEPTION

Overview

Before the rover can move, it must reconstruct the three-dimensional structure of its environment before it decides where it can go. The rover will accomplish this task through the use of a vision system. The problem of extracting three-dimensional information from the two-dimensional projection of a scene in an image and producing a description of the environment has been approached in many ways.

A complete description of an environment includes the identification of the constituent discrete objects, their locations in space and their volumetric extent, and the shapes of their surfaces. This information will be used by the rover to perform navigation.

One of the problems with which the rover must deal is the incompleteness of its model of the environment. It cannot see behind objects. As will be discussed in more detail later, the rover must assume that the backsides of objects are finite in extent and that paths behind them are possible. Otherwise, it will never get anywhere.

Stereo

One of the most useful representations of an environment is the depth map. In this representation, there is a distance associated with each point in the image plane.

One way to compute depth is through the use of stereo. The basic idea is quite simple. Two images are taken simultaneously through two cameras seperated by a known distance. The depth to any surface which appears in both images can be computed in a straightforward manner by noting the displacement (disparity) between the locations of the surface in the two images. The hard part is finding the same surface elements in both images. This is the correspondence problem.

Determining exactly which set of pixels corresponds to the same object in two different images is extremely difficult. Stereo algorithms approach the problem on a lower level instead and try to produce correspondences between image features rather than objects. What constitutes a feature and the method of establishing correspondence is different for different stereo systems.

Moravec used what he called an interest operator to select small regions of the image which exhibited enough peculiarity (i.e. were interesting enough) to be good candidates for re-identification in another image. The interest operator was designed to reject uniform areas and straight contours. Correspondence was established by searching smaller and smaller regions of the new image at successively higher resolutions for the best match with the regions in the original image of the same sizes and resolutions which contain the feature in question. The best match at each stage was the region with the highest correlation between brightness values--from coarsely averaged values down to the original pixels [Mor80].

The JPL robot used a stereo technique along the same lines. A feature is found by choosing a small window which exhibits a high variance among its brightness values. This window is then expanded until its correlation with itself shifted one pixel becomes sufficiently poor. Thus the window should match only when overlaid exactly. This procedure is designed to ensure that a non-homogeneous region contains enough information to be correctly identified in another image. Correspondence is then established by searching for the window of the same size in the second image with the highest correlation of brightness values with this chosen feature window from the first image [YaC78].

Features which pass the various tests for identifiability often are associated with abrupt changes of brightness in the image. Grimson's implementation [Gri81a] of a theory of human stereo vision due to Marr and Poggio [MaP79] explicitly finds these brightness changes by convolving the image with an operator which is a Laplacian of a Gaussian [MaH80]. Zero-crossings in the convolved image correspond to brightness changes in the original image. Features to be matched by the stereo algorithm are short segments of zero-crossing contours which are described by such attributes as slope (contrast in the original image) and orientation. Matching is accomplished by searching for a zero-crossing contour in the other image with the same attributes. Initial matching occurs on a coarsely filtered image. The crude estimates of disparity provided by this coarse matching can serve to bring successively finer filters into range of correspondence, whereby disparity values can be refined. This mechanism is thought to be what drives vergence movements of the eyes in human stereo fusion.

Motion--Discrete

Motion is another means whereby depth may be calculated. There are several ways in which motion can be used to accomplish this task. The problem is simplified by assuming that the rover's environment is static and that all perceived motion arises solely from the rover's own motion.

A different form of stereo which may be called temporal stereo uses images separated in time as well as space. One image is taken, the camera is moved, and another image is taken. If the motion of the camera is known then the distance between the points where the two images were taken can be calculated. This temporal stereo baseline can then be used to calculate depth in a manner completely analogous to binocular stereo.

A problem with this form of stereo is that the baseline is parallel with the direction of the rover's motion. Thus depth cannot be calculated for that part of the environment which lies straight ahead. As long as the rover remains on a straight-line path it will gather no new information about the area toward which it is heading. Not only is this the area which presumably holds the greatest interest for the rover, but moving blindly forward in this manner is likely to be dangerous.

This problem is probably not very severe, however. The existence of obstacles in the environment will force deviations from straight-line motion or better yet, the rover can cause and take advantage of the changing appearances of objects by deliberately tacking (zigzagging) toward its goal like a sailboat heading into the wind. The idea of using the rover's own motion to obtain structural information about the environment which will further enable that motion has a certain appeal.

The temporal stereo method outlined above does not specify how the motion of the camera is to be determined. Of more interest are methods which simultaneously solve for structure and motion. These methods also require that a correspondence be established between features in different images. Ullman showed that the identification of four features in each of three images provided sufficient information to compute both the structure (the relative distances of the four points) and the motion (to the same scaling factor) provided the four points were non-coplanar, the structure was rigid and the motion constant [Ull79]. The rover may then find the actual values for the depths and motion parameters by a single calibrating distance measurement of one of the features with a laser rangefinder. A recent result showed that the same can be accomplished for only two images by matching eight features and solving a system of linear equations [Lon81].

Once the motion has been computed, the depth of any additional matched features can be obtained easily. The disparity in the location of a feature in two images due to motion of the observer in a static environment is inversely related to the distance. The calculation progresses from correspondences through motion to depths.

Robustness

The above methods for computing depth via stereo or motion are not very robust as described. The depths obtained are very sensitive to inaccuracies in the positions of the image features which are matched. The methods can be made more robust simply by forming an overdetermined system. In the case of stereo, this means more images. In the case of motion, more features and/or images. The stereo system used by Moravec on his cart actually matched features from nine different images.

Incompleteness of the Depth Map

The depth map obtained by any of these methods is largely incomplete. Depths are calculated only at those points where features can be matched in different images. The relation between these features and real-world objects is not very obvious. Yet the perception system of the rover must produce a representation of the environment which describes objects, if it is to be at all useful.

Moravec's stereo system chose image features in a rather random fashion. Whether or not objects in the environment were represented by at least one feature was probabilistic. In one experiment, the cart drove blindly toward an obstacle without realizing its proximity because no feature had been chosen (for which a depth was calculated) which resided on the obstacle. Such are the dangers of an incomplete depth map.

Object Surfaces

Grimson has approached the problem of deriving surfaces of objects from an incomplete depth map [Gri81b]. The image features used in his implementation of a theory of human stereo vision are zero-crossing contours in the convolution of the image with an edge detector; these contours correspond to brightness changes in the original image. Brightness changes may arise from changes in surface reflectance, shadows, orientation or depth discontinuities. However, every depth discontinuity is assumed to be mirrored in a change of brightness, so surfaces may be confidently interpolated between the zero-crossing contours. (A pathological situation such as full frontal illumination on surfaces with uniform reflectance can be conceived which violates this assumption but there is no stereo system known, including the human one, which would not be misled in this case).

The actual surfaces which Grimson derived satisfied certain minimum energy, smoothness of curvature, and boundary tightness constraints. These surfaces serve to complete the depth map. Demetri Terzopolous has continued and extended considerably Grimson's work[Ter82]. This description of object surfaces, more of a two-dimensional layering than a full three-dimensional description of separate objects, is often called the $2\frac{1}{2}$-D sketch [Mar77].

Motion--Continuous

The continuous analogue of the discrete approach to determining motion is the optical flow. This term refers to the instantaneous velocity field of an image. Many researchers have worked on the problem of extracting useful information from the optical flow. In a static environment, the optical flow is completely determined by the structure of the environment and the motion of the observer. It is in principle possible to recover both the structure and the motion from the optical flow up to a constant scaling factor. The problem may be formulated as a huge minimization in which the motion parameters and the depth map are simultaneously solved for [Hor82]. Most likely an iterative solution method would be required.

This approach has certain advantages. It is robust in its very essence, using the entire image as input. Depths are calculated at all places in the image where variations in brightness occur, and uniform areas are naturally filled in from their boundaries by iterative methods resulting in a complete depth map (at least to the resolution of pixels). Only brightness gradients are needed to drive this method, not discrete changes in brightness (edges). At this time, it is not known whether the problem as formulated is tractable.

The problem mentioned earlier concerning a rover which uses motion to determine the structure of its environment having difficulty with the area which lies straight ahead is easily solved by the optical flow method. The camera simply can be panned back and forth while the rover moves, and rich information about the area directly ahead is obtained.

Segmentation

The final step in producing a full representation of the three-dimensional structure of an environment is the segmentation of an image into discrete objects. Simplistic procedures such as separating the image into regions of uniform brightness are not very good, for large variations in brightness can be expected to occur on single objects. Looking for places where the gradient of the depth map is large is more promising, but this approach is also subject to ambiguities. A sharp change in depth is not necessarily associated with an occluding contour seperating an object from the background. If a smoothness constraint is used to compute the depth map in the first place, the situation is worse, for discontinuities will be corrupted. Yet another approach is to analyze textures and colors in the image and look for discontinuities. How segmentation may be achieved without performing object recognition at a high level (which is even less well understood) is one of the current outstanding problems in vision research.

Wrap-up

Perception is the most difficult task of the rover. Besides the problems already mentioned, a rover's vision system has to deal with difficulties such as shadows, and variations in brightness of the same point in two images due to changes in surface orientation or illumination. For the methods discussed, environments with smoothly textured surfaces produce images with less information content. Despite all the difficulties, progress has been made in computer vision both through understanding the physics underlying image formation and through recognizing properties of the real world and how they constrain the possible three-dimensional interpretations of a two-dimensional image [Bra81].

Given the state of the art, a stereo system is a good choice for the vision system of a rover. Stereo is, of course, only one of several ways to obtain depth but the information it provides is perhaps the most exact and explicit. Among the other approaches mentioned, the idea of using the optical flow to compute a possibly complete, continuous depth map looks promising. This method would neatly avoid the need for interpolating surfaces between isolated points where depth can be constructed, which is a problem with a strict stereo system.

3. NAVIGATION

Overview

Once a model of the environment exists in which the layout of obstacles and their volumetric extent is at least partially available, the problem of navigation can be addressed. The task involves spatial reasoning--how to use the information in the 3-D model to find a path which avoids obstacles. As the rover actually moves about, its model of the environment will be updated, i.e. it will see the back sides of objects which it could not see before. If it turns out that an incorrect assumption has placed the rover in a cul-de-sac, it should be able to recover gracefully by retracing its steps.

Local path planning would be embedded in a global strategy towards reaching some desired goal. This process can be likened to that of a mountain climber who must scale cliffs, traverse ledges, cross passes, etc. in his quest for the summit. This multi-level processing strategy is a recurring theme in artificial intelligence.

Approaches to Obstacle Avoidance

The simplest obstacle avoidance algorithms involve proposing a simple straight-line path from start to goal, testing for collisions with known obstacles and proposing a new path. Determining whether a collision will occur involves solving for intersections between the volume swept out by the rover model along the path and the volumes of the obstacle models. Depending on the complexity of the models, this calculation can be quite difficult. A new path is proposed by selecting a vertex of the encountered obstacle as a new subgoal, and the search proceeds.

A severe problem with this approach is the total lack of any global information to initiate drastic path modification when necessary. Each new proposed path provides local information useful only for avoiding the immediate obstacle. This is a little like walking with one's head down. The search for a complete path from start to goal is likely to be inefficient and expensive.

Rather than looking for ways around obstacles only after bumping into them, a better idea is to look first. One of the early experiments with mobile robots--SRI's SHAKEY--introduced the notion of a visibility graph [Nil69]. If obstacles are modeled as polygonal projections of polyhedra in the plane of the rover's motion, then their vertices are the nodes of this graph and an arc exists between two nodes if and only if the two vertices can "see" each other--i.e. a straight-line path exists between the two points. SHAKEY structured its world by the use of a superimposed grid. Each cell within the grid was marked empty if no obstacle was present or full if any part of an obstacle was present. Thus the obstacles underwent a rectangular circumscription. The extent of the robot itself was represented by a circumscribed circle. Then to simplify the path-finding, the vertices of the obstacles were displaced by the radius of this circle and the model of the rover itself was simultaneously shrunk to a point. This technique has been called "growing" the obstacles. The point which represents the vehicle must stay outside the grown obstacles to avoid a collision. The optimal path was then found by applying a minimum-cost path determination algorithm [HNR68] to the visibility graph constructed from the vertices of the grown obstacles.

Moravec used a simpler way of modeling the obstacles at the cost of making the path-finding calculation more complex [Mor80]. The obstacles were also represented by circumscribed circles whose radii were augmented by the radius of the circumscribed circle representing the rover itself. At first glance, it seems a visibility graph can be constructed by considering the tangent lines connecting the circles. However, the departure and arrival points for a particular tangent line depends on the relative sizes and positions of the two circles it connects. There are an infinite number of possible vertices for each circle. Before a visibility graph can be constructed it is necessary to determine exactly which points on the circles will be used as nodes in this graph. Moravec used approximations to reduce the number of calculations his path finder actually had to make.

There is a collision avoidance algorithm due to Lozano-Perez and Wesley which generalizes the use of a visibility graph and grown obstacles to the case of a polyhedral object moving among known polyhedral obstacles [LoW79]. (As before, the problem for the rover can be simplified to two dimensions by considering the polygonal projections of the polyhedra in the plane of the rover's motion). The obstacles are grown so that they represent the locus of forbidden positions (those that would result in a collision) of an arbitrary reference point on the object (one of its vertices). This is the configuration space approach. Lozano-Perez has carefully examined the mathematics of configuration space [Loz80b][Loz81]. Because a polygonal object can take on arbitrary orientations, rotational motion has to be considered as well as simple translational motion. This turns out to be quite complicated. The grown obstacles have to be modeled in a higher-dimensional space. For rotation in the plane there is one degree of rotational freedom, and one extra dimension in configuration space. Unfortunately, the grown obstacles may now have curved surfaces, even if the original objects were strictly polygonal. Lozano-Perez solved this problem by dividing the rotation range between the starting and final orientations of the moving object into a fixed number of slices. Within each slice, the grown obstacles are approximated by bounding polyhedra. This approach is a refinement of an earlier approach by Udupa [Udu77]. A path is found by constructing a visibility graph from the grown obstacles and searching this graph in the usual way. The errors due to the approximations decrease, along with the chances of missing a possible path, as the number of slices taken increases. Of course, the computational burden increases as well.

Recently, Brooks and Lozano-Perez have developed a method which deals directly with the curved surfaces of grown obstcles [BrL82]. Predictably, this algorithm is very expensive, but it is guaranteed to find a path if one exists.

Brooks has quite a different approach to the obstacle avoidance problem [Bro82]. Instead of treating free space as the complement of the forbidden space occupied by grown obstacles in configuration space, he has developed an explicit representation of free space using generalized cones. A generalized cone is an exact description of the volume swept out by an object as it is translated and rotated through space. Free space can be represented as a union of generalized cones characterizing the natural "freeways" or "channels" between obstacles. Finding a safe path involves determining the sucessive orientations of an object as it sweeps out a volume during motion which result in generalized cones which are subsets (in the spatial sense) of the generalized cones representing sweepable volumes of free space.

This algorithm finds paths which tend to be equally far from all obstacles, unlike the visibility graph technique which finds paths which actually touch obstacles at vertices. The generalized cone method does not perform well on non-convex objects or in extremely cluttered environments.

Brooks and Lozano-Perez suggest using the generalized cone method in conjunction with the configuration space approach. The generalized cone method can be used in uncluttered environments and to make a first pass at cluttered environments. The configuration space approach can then be used to find paths through the difficult areas.

There is, of course, a motivation for modeling the obstacles and the rover itself as finely as possible. Coarse models may render invisible possible paths. In the worst case, the rover may embark on a search for an alternate path when it is not necessary. As usual, there is a tradeoff between accuracy and computational burden.

Generalizing to Three Dimensions

The obstacle avoidance algorithms discussed so far have treated only motion in the plane. Unfortunately, the problem for a rover is not that simple. Rovers will encounter ravines and overhangs and other dangers which can be made explicit only by employing a full three-dimensional model of the environment. Simple two-dimensional projections of three-dimensional obstacles are insufficient. In many cases, possible paths will be hidden from the rover. An example is an arch. If the arch is projected onto the ground then it becomes an obstacle which must be circumvented. The real solution--to go underneath (if the rover is short enough)--will be missed completely.

Unfortunately, the work in two dimensions doesn't generalize easily to three dimensions. Lozano-Perez has treated the problem of spatial planning in three dimensions in depth [Loz80b][Loz81]. First consider the case when rotations are disallowed. Obstacles are grown in a three-dimensional configuration space representing the three degrees of freedom in position. The visibility graph technique does not perform as well with three-dimensional obstacles because in general optimal paths do not traverse the vertices of grown obstacles and in some cases there are no paths which traverse vertices. One solution is to include additional nodes in the visibility graph which lie along edges of grown obstacles. An alternative approach uses a recursive decomposition of congifuration space [Loz80a].

The obstacles are first approximated crudely by bounding polyhedra. A solution is a continuous path from start to goal which passes through only free space. If no such path exists, then the bounding polyhedra are refined to represent the true obstacles more accurately, and the search continues. Since the search is for a path for a reference point, the recursive decomposition can continue to an arbitrary depth, limited only by inaccuracies in the known positions of objects.

If rotations in three dimensions are allowed, grown obstacles become complex curved objects in six-dimensional configuration space which are impossible to visualize. (Three additional degrees of freedom for rotation must be represented). Rather than deal with these objects, Lozano-Perez first divides the rotation ranges into a fixed number of subranges. For each subrange he defines a slice, representing those configurations for which the moving object intersects the obstacle for some orientation within the specified subrange. These slices are then approximated by bounding polyhedra and projected back into 3-space. The sequence of slice projections becomes an approximation for the original grown obstacle in the six-dimensional configuration space. The slice projections can be computed without ever actually computing the six-dimensional object. A path found by using the slice projections is guaranteed to be a solution to the original problem, but because of the loss of information, a path may be missed.

Consideration of the Locomotion System

A few words on representing the rover itself. As a first approximation, the model can be a polygon approximating the two-dimensional projection of the rover onto the ground. Any restriction on orientation implied by its motion should also be modeled. However, such a simple two-dimensional model may not reflect the advantages of the locomotion system employed by the rover. For example, a legged system which provides ground clearance for the body of the rover may be able to pick its way through cluttered terrain and may even be able to walk "sideways" if need be. The actual points of contact with the ground for this rover are much smaller than the projection of the rover body and there do not have to be continuous paths between contact points. These advantages must be reflected in the model which the navigation system uses to find (and reject) paths.

Dealing with an Unknown Environment

So far, none of the algorithms discussed have fully addressed the problem of navigation in a partially or completely unknown environment. A rover in this more realistic situation relies on perception to create and update its model of the environment. The robot HILARE is an example. Once again, obstacles are assumed to be polyhedra and the navigator operates on their polygonal projections. However, only the front of obstacles can be seen and their extent in depth remains unknown. This lack of a global view prevents a (global) visibility graph from being constructed a priori on which a path-finding algorithm may operate. Only local visibility graphs can be constructed. HILARE then structures its environment by creating convex polygonal cells, some of whose edges are object boundaries. The usefulness of these cells lies in the convexity, which guarantees there is a straight-line path between any two points in a cell. Regions which have not yet been acquired into the model are designated unknown. Thus the environment is structured locally and dynamically as the robot moves about [ChG80].

HILARE transforms the path-finding problem slightly by considering the connectivity graph of cells, rather than the visibility graph of object vertices. Two cells are connected if and only if they share an edge which the rover can cross. A path-finding algorithm is applied to this graph. Unfortunately, because Chatila does not grow the obstacles and shrink the rover to a point, this technique can run into problems because the rover may occupy more than one cell, which makes it difficult to find a path through those cells.

HILARE strikes out toward its goal using cost-driven heuristics which essentially select the shortest potential path for initial consideration. Thus in an unknown environment, a rover is often forced to operate "with its head down"--at least until it knows more. A potential path includes cells which are known to be traversible and unknown areas which are potentially traversible. A path initially goes through the midpoints of the common edges between cells and later can be smoothed and drawn up to obstacle vertices.

The JPL robot took a similar approach to the problem of operating in a partially unknown environment. The environment is segmented into three terrain types with polygonal boundaries--traversible, non-traversible, and unknown. However, the path planner of the JPL robot is the most complex of all those so far discussed. It uses an absolute minimum-distance cost metric. The visibility graph of obstacle vertices is constructed dynamically; nodes which may potentially be part of the optimal path (given the current information) are introduced in succession. Links to parent nodes are retraced to ensure optimality of the path and pruning techniques are used to keep the problem one of tree search rather than graph search [Tho77].

When a rover reaches an unknown area along a path it is exploring, it acquires new information through its vision system. New traversible areas are identified and the frontiers of the unknown are pushed back. A rover exploring an unknown environment may unwittingly enter a cul-de-sac. Both HILARE and the JPL robot know it is time to backtrack when there are no longer any further unknown (potentially traversible) areas to explore. Thus the importance of identifying unknown areas is two-fold. Potential paths are not ruled out and the rover will not blindly butt itself up against a dead-end.

Backtracking may also be triggered heuristically by deciding that the current path is causing the rover to stray too far from its goal. Backtracking is accomplished by keeping a stack of subgoals reached. When a decision is made concerning which potential path to explore next, the current position is placed on the stack. If the path turns out to be a bogus one, the position where the ill-advised decision was made can be regained by popping the stack. The area between is then retraversed.

Real-World Considerations

The approaches to rover navigation in an unknown environment discussed above are rather idealized. In general, the rover's model of the environment will contain inaccuracies due to noise and shortcomings of the vision system. Thus there will be errors in the locations of obstacles. The rover's own position will not be known precisely either.

Moravec's cart deduced its own motion by calculating the coordinate transformation between successive images from tracked features. Inaccuracies in the locations of those features were reflected in the cart's assessment of its own (new) position. Moravec found that these errors could propagate rapidly.

The JPL robot used an odometer and gyro-compass to update its position continuously by dead reckoning. The positions of obstacles are calculated with respect to this vehicle position, resulting in a propagation of error in the other direction. Some calculation which finds the best fit of the rover's position with respect to the previously known positions of obstacles (or any features) in the environment can help to keep all position inaccuracies down.

In any event, because these errors do exist, it is not reasonable to send the rover on paths which move directly from obstacle vertex to obstacle vertex. At the least, the rover is likely to take a lot of bruising! A "cushion" should be placed around obstacles (which can be incorporated at the time of obstacle growing) to compensate for these positional errors. Note that recursive subdivision of space naturally places these cushions around obstacles. Also note that Brooks' algorithm finds paths which really do avoid obstacles, rather than just barely avoiding them. Unfortunately, (or predictably) this algorithm bogs down in extremely cluttered environments.

Inevitably, the segmentation of the environment into traversible, non-traversible, and unknown areas will run awry, and some obstacle will not be identified as such in the terrain model. Both HILARE and the JPL robot have various sensors to aid in the detection of unexpected obstacles such as proximity and tilt sensors and laser rangefinders.

Also, the locomotion system employed by the rover will not be able to negotiate precisely the piecewise linear trajectory indicated by the path finder. A wheeled or treaded system can manage to turn in place but it is more desirable to make heading changes without stopping the vehicle. Because the steering rate is finite there will be a tracking error. A legged system will be able to follow a trajectory more accurately. For any type of locomotion system, it is necessary to model the actual rover motion it generates to ensure obstacle avoidance. This can also provide a sort of "kinesthetic" sense to aid in the reckoning of the rover's current position.

Wrap-up

All in all, the navigation problem is much more tractable than the perception problem. Effective roving in an unknown environment requires an updatable, structured representation of the environment--a model which is segmented into regions classified as traversible terrain (free space), obstacles (forbidden space), and unknown areas. The actual navigating needs a path planner which can find paths to a goal and enable retracing of paths when necessary. The only method for spatial planning and obstacle avoidance which will work in general three-dimensional environments, not just in restricted cases, is the configuration space approach. Unfortunately, no rover project has used this method.

FINAL WRAP-UP

The Changing World

One final aspect of most real-world environments which has been suppressed in the above discussion is change. Creating and updating a model of the environment is extremely difficult when objects are moving. A way of determining the motions of objects and predicting their changing locations is required. The rover, too, is moving so all perceived motion is relative motion, adding another complication. Path planning can become frustrated because paths may exist only temporarily. A rover can conceivably find itself trapped when it tries to retrace a path that no longer exists.

The problems that arise in dealing with a dynamic environment are very complex and they are only hinted at here. It would be premature to try to tackle these problems now. At this time research should concentrate on solving the rover problem for static environments.

The Rover Laboratory

The three capabilities that a rover must have--locomotion, perception, and navigation--will be realized in interacting subsystems of a working rover. The perception system creates and updates the model of the environment which the navigator uses to search for paths. A chosen path is passed along to the locomotion system in a form in which it may be executed. The interactions can be more complex. The characteristics of the locomotion system are embedded in the representation of the rover used by the navigation system. Both these characteristics and information provided by the perception system contribute to determining the rover's current position--also necessary for navigation.

These three necessary subsystems and any others such as a manipulator must be able to operate concurrently and should be organized in a distributed system under the coordination of an executive. There may also be capacities for error recovery and learning, as well as a general capacity for problem solving and decision making driven by an embedded representation of the rover's "raison d'etre," whether it be planetary exploration or the discovery of new oil deposits.

Working rovers are robots which are continuously moving, seeing, and thinking. As such they provide excellent laboratories for current research in artificial intelligence.

ACKNOWLEDGEMENTS

The Viking mission to Mars originally sparked my interest in rovers.

For their helpful comments on drafts of this paper, I would like to thank Tomas Lozano-Perez, Michael Brady, Rod Brooks, Bob Cunningham, Bruce Donald, Eric Grimson, Ellen Hildreth, and Berthold Horn.

REFERENCES

[Bra81] J. Michael Brady, *Computational Approaches to Image Understanding*, MIT Artificial Intelligence Lab Memo 653, October 1981.

[BrL82] Rodney A. Brooks and Tomas Lozano-Perez, in preparation, 1982.

[Bro82] Rodney A. Brooks, *Solving the Find-Path Problem by Representing Free Space as Generalized Cones*, MIT Artificial Intelligence Laboratory Memo 674, May 1982, also *IEEE Transactions on Systems, Man, and Cybernetics*, to appear.

[Cha81] Raja Chatila, *Systeme de Navigation pour un Robot Mobile Autonome: Modelisation et Processus Decisionnels* (These), Universite Paul Sabatier, Toulouse, France, July 1981.

[ChG80] Raja Chatila and Georges Giralt, "A Task-Oriented Navigation System for the Mobile Robot HILARE", *Proceedings of the 2nd International Meeting on Artificial Intelligence*, October 1980.

[GHM74] F. Gubina, H. Hemami, and Robert B. McGhee, "On the Dynamic Stability of Biped Locomotion", *IEEE Transactions on Biomedical Engineering* BME-21, No. 2, 102-108, March 1974.

[Gri81a] W. Eric L. Grimson, "A Computer Implementation of a Theory of Human Stereo Vision", *Philosophical Transactions of the Royal Society of London* B 292, 217-253, 1981.

[Gri81b] W. Eric L. Grimson, *From Images to Surfaces: A Computational Study of the Human Early Visual System*, MIT Press, Cambridge, MA, 1981.

[HNR68] P. E. Hart, Nils J. Nilsson, and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths", *IEEE Transactions on Systems Science and Cybernetics* SSC-4, No. 2, 100-107, July 1968.

[Hor82] Berthold K. P. Horn, in conversation, March 1982.

[JPL77] *Feasibility of a Mars Multi-Rover Mission*, Jet Propulsion Lab Internal Report 760-160, February 1977.

[Koh78] G. W. Kohler, "Status of Remote Systems Technology", *Proceedings of the 8th International Symposium on Industrial Robots*, 815-828, 1978.

[Lon81] H. C. Longuet-Higgins, "A Computer Algorithm for Reconstructing a Scene from Two Projections", *Nature* 293, 133-135, 10 September 1981.

[LoW79] Tomas Lozano-Perez and Michael A. Wesley, "An Algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles", *Communications of the Association for Computing Machinery* 22, No. 10, 560-570, October 1979.

[Loz80a] Tomas Lozano-Perez, *Spatial Planning with Polyhedral Models*, MIT Artificial Intelligence Laboratory Technical Report, September 1980.

[Loz80b] Tomas Lozano-Perez, *Spatial Planning: A Configuration Space Approach*, MIT Artificial Intelligence Laboratory Memo 605, December 1980, also *IEEE Transactions on Computers*, to appear.

[Loz81] Tomas Lozano-Perez, "Automatic Planning of Manipulator Transfer Movements", *IEEE Transactions on Systems, Man, and Cybernetics* SMC-11, No. 10, 681-698, October 1981.

[MaH80] David C. Marr and Ellen C. Hildreth, "Theory of Edge Detection", *Proceedings of the Royal Society of London* B 207, 187-217, 1980.

[MaP79] David C. Marr and Tomaso Poggio, "A Computational Theory of Human Stereo Vision", *Proceedings of the Royal Society of London* B 204, 301-328, 1979.

[Mar77] David C. Marr, *Representing Visual Information*, MIT Artificial Intelligence Lab Memo 415, May 1977.

[McG77] Robert B. McGhee, "Control of Legged Locomotion Systems", *Proceedings of the 18th Joint Automatic Control Conference*, 205-215, 1977.

[McI79] Robert B. McGhee and G. I. Iswandhi, "Adaptive Locomotion of a Multilegged Robot over Rough Terrain", *IEEE Transactions on Systems, Man, and Cybernetics* SMC-9, No. 4, 176-182, 1979.

[McP74] Robert B. McGhee and A. L. Pai, "An Approach to Computer Control for Legged Vehicles", *Journal of Terramechanics* 11, No. 1, 9-27, 1974.

[McS74] Robert B. McGhee and S. S. Sun, "On the Problem of Selecting a Gait for a Legged Vehicle", *Proceedings of the 6th IFAC Symposium on Automatic Control in Space*, 1974.

[Mor80] Hans P. Moravec, *Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover*, CMU Robotics Institute Technical Report 3 (also Ph.D. thesis, Stanford), September 1980.

[Mor82] Hans P. Moravec, "The CMU Rover", *Proceedings of the American Association for Artificial Intelligence Conference*, August 1982.

[Nil69] Nils J. Nilsson, "A Mobile Automaton: An Application of Artificial Intelligence Techniques", *Proceedings of the International Joint Conference on Artificial Intelligence*, 509-520, 1969.

[OMJ76] David E. Orin, Robert B. McGhee, and V. C. Jaswa, "Interactive Computer Control of a Six-Legged Robot Vehicle with Optimization of Stability, Terrain Adaptability, and Energy", *Proceedings of the 1976 IEEE Conference on Decision and Control*, 1976.

[Ori82] David E. Orin, "Supervisory Control of a Multilegged Robot", *The International Journal of Robotics Research* 1, No. 1, 79-91, 1982.

[Rai81] Marc H. Raibert, "Dynamic Stability and Resonance in a One-Legged Hopping Machine", *Proceedings on Theory and Practice, Robots and Manipulators*, 419-429, 1981.

[Ter82] Demetri Terzopolous, *Multi-Level Reconstruction of Visual Surfaces: Variational Principles and Finite Element Representations*, MIT Artificial Intelligence Lab Memo 671, April 1982.

[Tho77] Alan M. Thompson, "The Navigation System of the JPL Robot", *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, 749-757, 1977.

[Udu77] S. Udupa, "Collision Detection and Avoidance in Computer Manipulators", *Proceedings of the 5th International Joint Conference on Artificial Intelligence*, 1977.

[Ull79] Shimon Ullman, *The Interpretation of Visual Motion*, MIT Press, Cambridge, MA, 1979.

[YaC78] Y. Yakimovsky and Robert Cunningham, "A System for Extracting Three-Dimensional Measurements from a Stereo Pair of TV Cameras", *Computer Graphics and Image Processing* 7, 195-210, 1978.