THE CMU ROVER

Hans P. Moravec
Robotics Institute
Carnegie-Mellon University
Pittsburgh, PA 15213

Abstract

An advanced camera-equipped mobile robot is being built at the CMU Robotics Institute to support research in control, perception, planning and related issues. It will be used initially to continue and extend visual navigation work (the Cart Project) completed in 1981 at Stanford's AI Lab, but is designed with mechanical, sensor and controller flexibility which should permit it to support a wide variety of other efforts.

Introduction

Experience with the Stanford Cart [2], a minimal computer controlled mobile camera platform, suggested to me that, while maintaining such a complex piece of hardware was a demanding task, the effort could be worthwhile from the point of view of AI and vision research. A roving robot is a source of copious and varying visual and other sensory data which force the development of general techniques if the controlling programs are to be even minimally successful. By contrast the (also important) work on disembodied visual data often focuses on relatively small image sets, and is in some danger of overspecializing. Drawing an analogy with the natural world, I believe it is no mere co-incidence that in all cases imaging eyes and large brains evolved in animals that first developed high mobility.

Other groups have come to similar conclusions [3] or have developed sophisticated rovers with an eye to near term applications [1] [4]. The new Unimation rover [1] is very similar in size, shape and mechanical capabilities to the machine we are building. It achieves a full three degrees of freedom of floor-plane mobility by use of three novel "omnidirectional" wheels which, by virtue of rollers in place of tires, can freely move in a direction broadside to the wheel plane, as well as performing, under motor control, the usual wheel motion. We achieve the same effect with steerable wheel assemblies, described below.

Overview

The project so far has been focused on developing a smoothly functional and highly capable vehicle which will serve a wide variety of future research. A hint of this future work can be found in [2] and later in this paper.

The shape, size, steering arrangements and onboard and external processing capabilities of the rover system were chosen to maximize the flexibility of the system (naturally limited by present day techniques).

The robot is cylindrical, about a meter tall and 30 cm in diameter (Figure 1) and has three individually steerable wheel assemblies which give it a full three degrees of freedom of mobility in the plane (Figures 2, 3). Initially it will carry a TV camera on a pan/tilt/slide mount, several short range infrared and long range sonar proximity detectors, and contact switches. Our design calls for about a dozen onboard processors (about half of them powerful 16 bit MC68000s) for high speed local decision making, servo control and communication (Figure 4). Serious processing power, primarily for vision, is to be provided at the other end of a remote-control link by a combination of a host computer VAX 11/780 an FPS-100 array processor and a specially designed high performance analog data acquisition and generation device.

We hope eventually to provide a manipulator on the rover's topside, but there is no active work on this now. We chose the high steering flexibility of the current design partly to ease the requirements on a future arm. The weight and power needed can reduced by using the mobility of the rover to substitute for the shoulder joint of the arm. Such a strategy works best if the rover body is given a full three degrees of freedom (X, Y and angle) in the plane of the floor. Conventional steering arrangements as in cars give only two degrees at any instant.
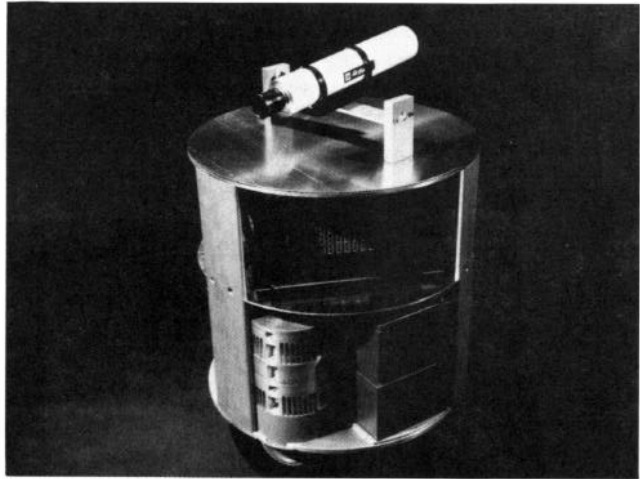

Figure 1:   The Rover

Details

Three degrees of freedom of mobility are achieved by mounting the chassis on three independently steerable wheel assemblies (Figure 2). The control algorithm for this arrangement at every instant orients the wheels so that lines through their axles meet at a common point. Properly orchestrated this design permits unconstrained motion in any (2D) direction, and simultaneous independent control of the robot's rotation about its own vertical axis. An unexpected benefit of this agility is the availability of a "reducing gear" effect. By turning about the vertical axis while moving forward the robot derives a mechanical advantage for its motors. For a given motor speed, the faster the rover spins, the slower it travels forward, and the steeper the slope it can climb. (Visualization of this effect is left as an exercise for the reader.)
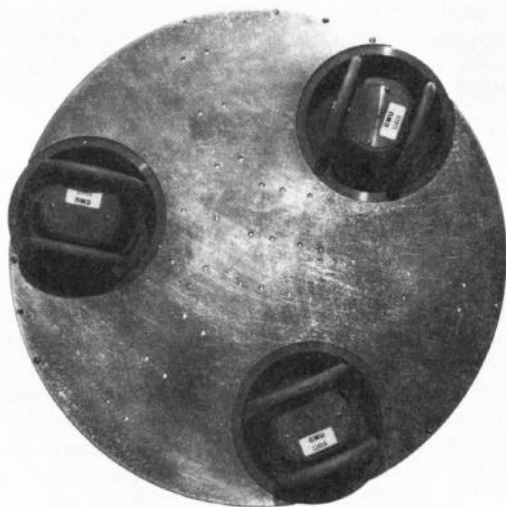
Figure 2:   The Wheelbase

Figure 3:   The Wheel Assembly

To permit low friction steering while the robot is stationary, each assembly has two parallel wheels connected by a differential gear. The drive shaft of the differential goes straight up into the body of the robot. A concentric hollow shaft around this one connects to housing of the differential (Figure 3). Turning the inner shaft causes the wheels to roll forwards or backwards, turning the outer one steers the assembly, with the two wheels rolling in a little circle. The assemblies were manufactured for us by Summit Gear Corp.

Each shaft is connected to a motor and a 4000 count/revolution optical shaft encoder (Datametrics K3). The two motors and two encoders are stacked pancake fashion on the wheel assembly, speared by the shafts. There are no gears except for the ones in the differential. (Figure 3 shows a schematic cross section of a complete motor/wheel-assembly structure, and a partially assembled stack in the flesh).

The motors are brushless with samarium-cobalt permanent magnet rotors and three-phase windings (Inland Motors BM-3201). With the high energy magnet material, this design has better performance when the coils are properly sequenced than a conventional rotating coil motor. The coils for each are energized by six power MOSFETs (Motorola MTP1224) mounted in the motor casing and switched by six opto-isolators (to protect the controlling computers from switching noise) whose LEDs are connected in bidirectional pairs in a delta configuration, and lit by three logic signals connected to the vertices of the delta.

The motor sequencing signals come directly from onboard microprocessors, one for each motor. These are CMOS (Motorola MC146805 with Hitachi HM6116 RAMs) to keep power consumption reasonable. Each processor compares the output of its shaft encoder to a desired motion (supplied by yet another processor as a time parameterized function) and energizes or de-energizes the appropriate motor phase to keep the motor on track with no wasted power. Since each motor has a potential power dissipation of 250 watts this is more than a nicety.

The shaft encoder outputs and the torques from all the motors, as estimated by the motor processors, are monitored by another processor, *the Simulator*, a Motorola MC68000 (with all CMOS support circuitry the power requirement for our 32K 68000 is under one watt. The new high performance 74HC series CMOS allows operation at full 10MHz speed.), which maintains a dead-reckoned model of the robot's position
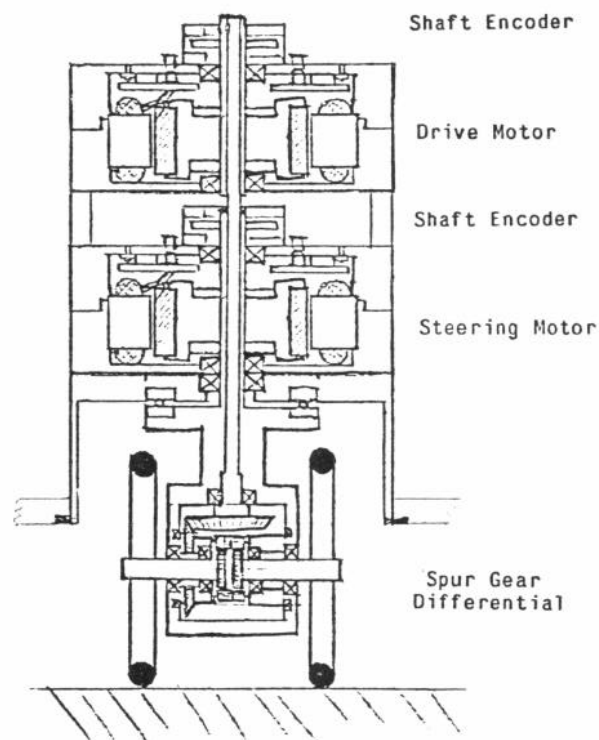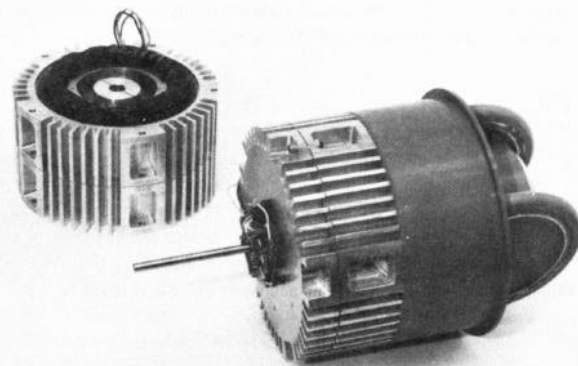
from instant to instant. The results of this simulation (which represents the robot's best position estimate) are compared with the desired position, produced by another processor (68000), *the Controller*, in yet another processor (68000), *the Conductor*, which orchestrates the individual motor processors. The conductor adjusts the rates and positions of the individual motors in an attempt to bring the Simulator in line with requests from the Controller, in what amounts to a highly non-linear feedback loop.

Other onboard processors are:

Communication A 68000 which maintains an error corrected and checked packet infrared link with a large controlling computer (a VAX 11/780 helped out by an FPS 100 array processor and a custom high speed digitizer) which will do the heavy thinking. Programs run in the Controller are obtained over this link.

Sonar A 6805 which controls a number of Polaroid sonar ranging devices around the body of the rover. These will be used to maintain a rough navigation and bump avoidance model. All measurements and control functions of this processor and the following ones are available (on request over a serial link) to the Controller.

Camera A 6805 which controls the pan, tilt and slide motors of the onboard TV camera. This compact camera (Edo Western 1631) broadcasts its image on a small UHF or microwave transmitter (made by 3DBM Inc?). The signal is received remotely and the video signal captured by a high bandwidth digitizer system and then read by the remote VAX. There are tentative plans for a minimal onboard vision system using a 68000 with about 256K of extra memory onboard the rover, for small vision tasks when the rover is out of communication with the base system.

Proximity A 6805 which monitors several short range modulated infrared proximity detectors which serve as a last line of defense against collision, and which sense any drop off in the floor, and contact switches.

Utility A 6805 which senses conditions such as battery voltage and motor temperature, and which controls the power to non-essential but power hungry systems like the TV camera and transmitter.
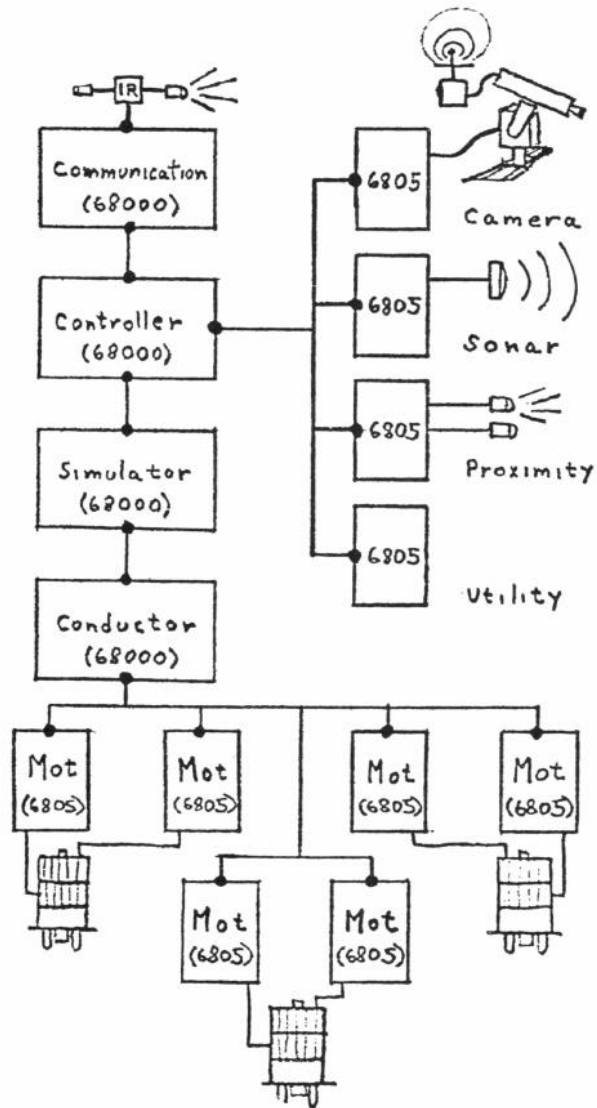
Communication between processors is serial, via Harris CMOS UARTs, at a maximum speed of 256 kilobaud. The Conductor talks with the motor processors on a shared serial line and the Controller communicates with the Sonar, Camera, Proximity, Utility and any other peripheral processors by a similar method.

The processors live in a rack on the second storey of the robot structure (Figure 1), between the motor and battery assembly (first floor) and the camera plane (penthouse). Figure 4 shows the rack alone, not fully populated.

The rover is powered by six sealed lead-acid batteries (Globe gel-cell 12230) with a total capacity of 60 amp hours at 24 volts. The motors are powered directly from these, the rest of the circuitry derives its power indirectly from them through switching DC/DC converters (Kepco RMD-24-A-24 and Semiconductor Circuits U717262). Each 6805 processor draws about one eighth of a watt, each 68000 board only one watt.

Physically the robot is a meter tall and 55 centimeters in diameter. It weighs 90 kilograms. The maximum acceleration is one quarter g, and the top speed is 10 kilometers/hour. With appropriate onboard programming the motions should be very smooth. The great steering flexibility will permit simulation of other steering systems such as those of cars, tanks and boats and other robots by changes in programming.

Figure 4: The Processors
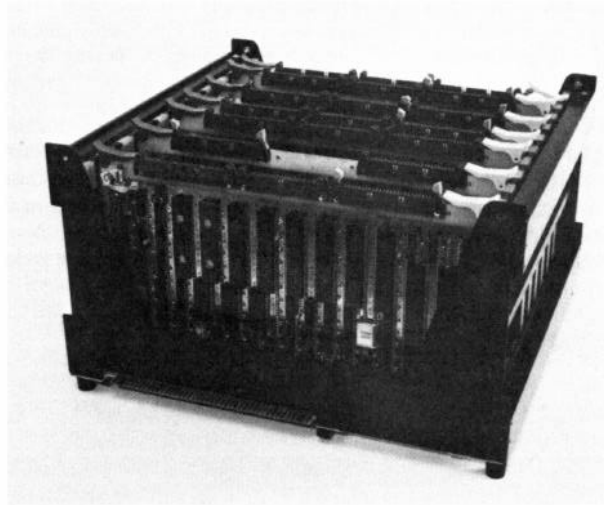
Progress

As of this writing (June 8, 1982) we are awaiting final fabrication of the remainder the robot's mechanical parts, and have bench prototypes of all the major electronics. Debugging is proceeding furiously. Three instances of each of the two processors are completely operational. Low level control software has been debugged on a rover simulation, but remains to be tested on hardware. We feel confident that a minimal demonstration of a running rover can be arranged for the mid August date of this conference.

Promises

The hope expressed in the introduction, that mobility somehow forces evolution in the direction of true intelligence, may already be being vindicated. Our efforts to devise a suitable programming language useful for writing Controller scripts for interesting robot tasks floundered until we realized clearly that controlling a rover is an essentially different task than controlling a robot arm. An arm, in the typical assembly line setting, executes a sequence of tasks, possibly with branching, that can be visualized and charted in advance. Exceptional conditions are an infrequent nuisance that can be handled by an error exit or a small number of interrupt activated routines that reset the situation to some standard state and return control to the mainstream code. This approach does not work for programming a rover in typical environments because *almost every condition encountered by a rover is exceptional*. Obstacles and opportunities cannot in general be anticipated in advance, and if we attempted to deal with them by interrupt coding, our stacks would soon overflow as exception interrupted exception.

We are now working on an approach in which the Controller is governed by a set of production rules, each rule having an activating condition involving the state of sensors and internal variables and consequences which affect internal variables and motor functions. Conceptually the potentially relevant rules are active at all times, and may take over control from each other as preconditions change. Conflicts are mediated by weights (or confidence levels) produced by activated rules. Sequences of steps are made possible in this arrangement by the introduction of sequence variables, which trigger rules and are then changed by those rules to trigger successive ones. To maintain some coherence of behavior there can also be variables which affect the preconditions of large classes of rules - for instance a variable that at a low value causes the robot to ignore minor signals from its proximity sensors while responding to them vigorously if high. A sequence of proximity signals would gradually raise the value of this variable, and the robot's behavior would change from boldly pursuing its main goal (whatever that may be) to cautiously feeling its way around. We are tempted to give such variables names like "fear" and "enthusiasm", but will refrain from doing so for the moment.

Our system must operate in real time, a need which distinguishes it from typical rule based expert systems. We currently imagine a scheme in which, for large jobs, the entire rule system is maintained in the remote VAX, with efficiently indexed rule subsets likely to be relevant being dynamically downloaded as conditions change. Even so it is clear that the load on the Controller will be heavy. We are preparing a 256 Kbyte memory expansion board for it, and are looking forward to a challenging job.

Other interesting preliminary thinking has resulted in a scheme by which a very simple arm with only three actuators will enable the robot, making heavy use of its great steering flexibility, to enter and leave through a closed standard office door. Discussions with the distributed sensor net crew here has raised the possibility of equipping the robot with ears, so it could passively localize sound, and thus perhaps respond correctly, both semantically and geometrically, to a spoken command like "Come here!". We are also toying with the idea of a phased array sonar with about 100 transducers operating at 50 KHz which, in conjunction with the high speed analog conversion device mentioned above and the array processor, would be able to produce a modest resolution depth map (and additional information) of a full hemisphere in about one second, by sending out a single spherical pulse, then digitally combining the returned echoes from the individual transducers with different delay patterns to synthesize narrow receiving beams.

Support

References

1. Carlisle, B. and B. Shimano. The Unimation mobile robot. personal communication.

2. Moravec, H. P. *Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover*. Ph.D. Th., Stanford University, September 1980.

3. Raphael, B.. *The Thinking Computer*. W. H. Freeman and Company, San Francisco, California, 1976.

4. Yakimovsky, Y. and R. Cunningham. "A System for Extracting Three-Dimensional Measurements from a Stereo Pair of TV Cameras." *Computer Graphics and Image Processing 7* (1978), 195-210.