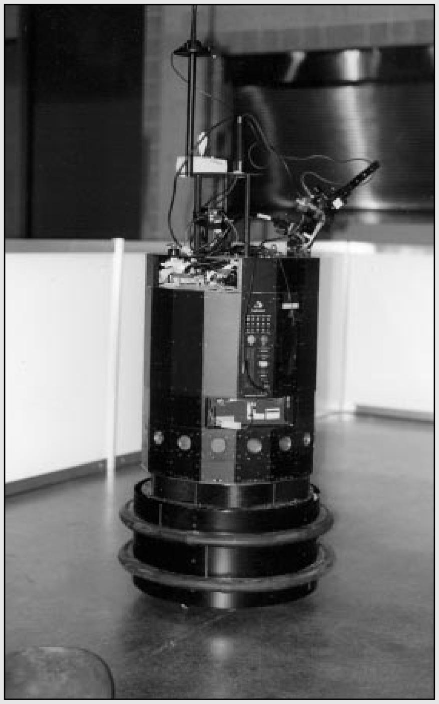
AI Magazine Volume 19 Number 3 (1998) (© AAAI)

Profile of a Winner: McGill University

Much of the work at the McGill Mobile Robotics Lab concerns computational problems related to use of sensors: vision, laser, and sonar. As a result, the McGill team entered the nonmanipulator category. The team was made up of four students: Francois Belair, Eric Bourque, Deeptiman Jugessur, and Robert Sim, with myself acting as faculty mentor. The robot they used was a NOMAD 200, a chest-height cylindrical robot with a three-wheeled synchrodrive, a standard ring of 16 sonar sensors, and a single-color camera mounted on a simple pan-and-tilt unit (see figure). Although planning for the competition started early, most of the key software for the competition was custom designed in the 6 to 10 weeks before the conference. Early in the design process, several existing research tools and subsystems were used, but as the system developed, most preexisting code was either heavily modified or replaced. These revisions took place to maximize efficiency and reliability as well as to reduce dependence of subsystems not fully under the control of team members (in the last weeks, the team members worked on an almost 24-hour schedule and were unable to tolerate delays that would have been incurred by having to wait for other people to modify their subsystems). Even at the time, it was apparent that the use of special-purpose modules would limit the reusability of the code and incur other disadvantages, but it became an inevitable necessity.

The design of the McGill entry was very loosely based around the McGill mobile robotics architecture, a software interconnection methodology. The key software modules, instantiated in the form of UNIX, were for planning and collision avoidance, scheduling and error recovery, and user interface and diagnostics. The operation of some these modules is as follows:



McGill Entry in the Find-Life-on-Mars Event.

The basic visual-perception module dealt with using color to classify and segment the image. It used a multidimensional lookup table to map pixels to specific objects (such as rock, floor, or target). In its initial form, it was able to compute the color distribution of groups of pixels, but this approach proved to be excessive for the types of object that were in the actual environment. The classifier could be trained by showing it samples of objects of interest along with their correct labeling. Once pixels were classified, the vision module proceeded to group them into blobs that could efficiently be described by polygons of limited complexity. The three-dimensional position of these obstacles was computed from knowledge of the camera geometry and a flag ground-plane assumption. These labeled polygons were then transmitted to the mapping and planning module.

The mapping and planning module dealt with the maintenance of a long-term map of the environment, collision avoidance, path planning, and target acquisition. The map was composed of obstacles observed from either sonar or vision. Older objects were gradually removed to account for uncertainty growth as a result of dead-reckoning errors. Path planning was accomplished by computing an obstacle-free convex region about the robot for simple short-range planning, combined with long-range planning that directed the robot within this "safe polygon." Although several long-range path-planning modules were developed, random motion driven by directly observed targets proved sufficient in the final competition.

The object-recognition module that was used matched already-classified blobs to images of known objects using subspace projection. To improve the performance of the recognizer, the planning module attempted to approach targets to get them into a standard viewing position. This type of technique can be used to recognize complex objects, but it can have difficulty with illumination variations, which proved to be a challenge until the last minute.

One of the greatest tests of the competition was coping with issues of system integration and robustness. In fact, a last-minute communications problem led to the robot occasionally becoming blind, with potentially disastrous consequences. In the last days before the finals, the students worked a straight 40-hour debugging stretch, a frighteningly common behavior at the AAAI competitions. A more subtle issue is the standard question of how to reconcile conflicting input in a complex AI or robotic system. For example, during the preliminaries, the vision module managed to detect a spectator's colored clothing through a semitransparent partition, producing an internal inconsistency that incapacitated the robot. As Rob Sim put it, the robot "chose to go after an audience member's shoes and barreled full speed into the wall, while the unfortunate target leapt back in unmasked terror."

It is fair to say the competition served to illustrate the importance of some underappreciated issues in our lab as well as extensively test some software and hardware modules (some of the work has also played a role in ongoing thesis research). The drive and resourcefulness of my students, and the others in the competition, is a refreshing and wonderful aspect of the event for those who have a direct involvement. I believe the participants, by virtue of their increased appreciation of a range of issues, are much enriched for the experience.

– Gregory Dudek