Future Directions in Multi-Robot Autonomy and Planetary Surface Construction

Trey Smith, Reid Simmons, Sanjiv Singh, and David Hershberger
Carnegie Mellon University Robotics Institute
Pittsburgh, PA 15213
{trey, reids, ssingh, hersh}@ri.cmu.edu

10 May 2001

1 Introduction

We identify the contributions of a general software architecture for intelligent robotics under development at our lab: first, by placing it in the context of a taxonomy of intelligent robotic functionalities, and second, by relating it to the challenges posed by future planetary surface construction scenarios.

The architecture we describe is the key research focus of the Distributed Robotic Architectures (DIRA) Project. In this paper, we only give an overview of its goals and implementation. For more detail, consult [Simmons et al., 2000].

2 Overview of the Architecture

Some complex and demanding tasks have requirements which make them difficult for a single robot to perform:

- Widely separated actions must be performed simultaneously
- A great deal of work must be performed in a short time
- The system must be very reliable (but a single robot can fail)

In practice, these tasks require a team of multiple robots. To date, most research efforts in the area of robot teams have used many copies of the same type of robot, focusing on preventing harmful interactions between the robots (such as collisions), so that the speed of execution of the task increases with the number of robots. This type of team can be used successfully for exploration, or demining [Mataric, 1992].

However, we have identified a more difficult class of tasks which require

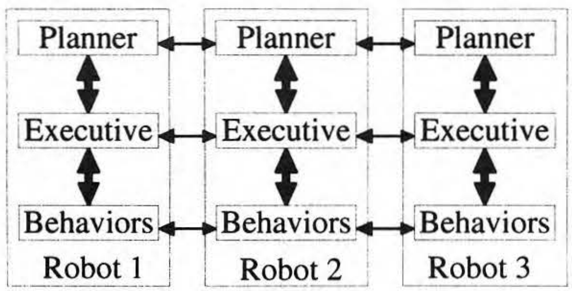

Figure 1: Three layer architecture

- Heterogeneous robots. In a complex task with many different sub-tasks, a single type of robot may not be able to fulfill them all. Even if you can design a general-purpose robot, a team of generalists will often be outperformed by a team of specialists [Whittaker et al., 2000].
- Explicit coordination. In some cases, members of a robot team cannot make any progress on the task without working together. A single robot may not be strong enough, or may not be able to sense the effects of its own actions. In these tasks, teamwork means more than just staying out of each other's way [Donald, 1995].

Our software architecture supports this more complex team structure. Our research is balanced between concept development applicable to many domains, and implementation on our multi-robot test bed.

2.1 Layered Structure

Each agent in our architecture is built from three layers (figure 1). As we move from top to bottom through the layers, they become progressively less

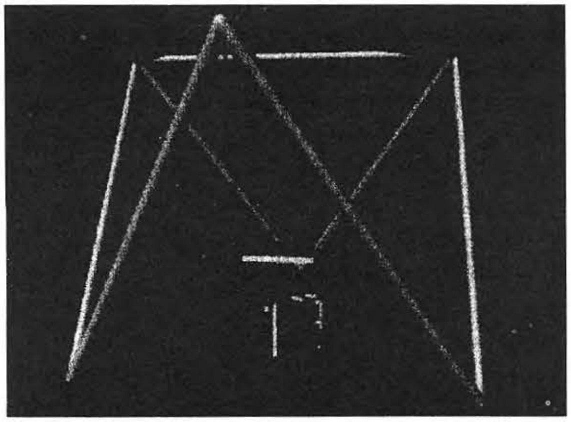

Figure 2: Cooperating robots in the DIRA simulator

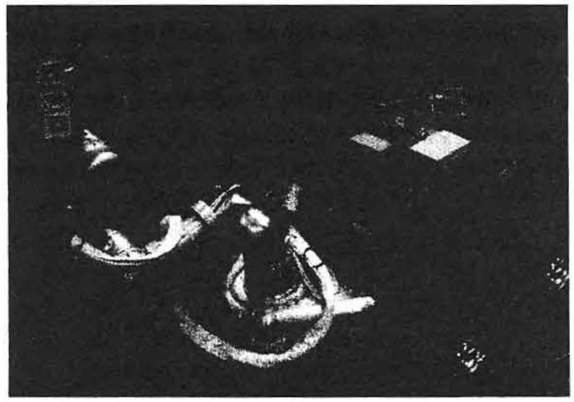

Figure 3: The mobile arm

abstract and more tightly tied to the physical sensors and actions of the robot. The layers within a single robot communicate with each other, and each layer can communicate with the corresponding layer in other robots of the team, so that they each bear part of the responsibility of coordinating the team's actions:

- Planner: Reasons about goals, breaking them down into sub-goals and generating flexible hierarchical plans. The planning layers of different robots communicate about forming teams, and about commitments to perform actions. This layer is least tied to real-time constraints.
- Executive: Runs the plans generated by the planner, and tracks the state of the execution. Executive layers communicate with each other in order to synchronize tasks. The executive interacts in real time.
- Behaviors: This layer involves stateless, reactive control. Behavioral layers of different robots coordinate tight physical interactions between the robots. Behaviors have the fastest real-time cycles in the system.

2.2 Test bed

The test bed demonstrates a construction task involving assembly of a structure from beams. The task was inspired by watching the construction of a real steel-frame building. The test bed uses three robots of different types:

- Robocrane: The gross manipulator, which allows rough control of heavy objects (hundreds of kilos). The end effector is a triangle suspended from six cables, and the crane exerts full six degree of freedom control on the triangle by extending or retracting the cables. A beam can be suspended from the end effector. The Robocrane is on loan from the National Institute for Standards and Technology (NIST), where it was developed [Albus et al., 1992].
- Mobile arm: The fine manipulator, which allows tighter control than the robocrane, but exerts less force (it can lift tens of kilos). The arm is roughly the size of a human arm, and mounted on a rolling base which allows it to move into position. The arm was developed at Metrica Corporation, under contract to Johnson Space Center.
- Roving eye: Acts as a mobile observer to direct the two manipulator robots. The roving eye carries a stereo pair of cameras which allow it to sense the position of a beam relative to the stanchion it will be mated with.

In order to add a beam to the structure, the crane must move the beam roughly into position, and then allow the mobile arm to grasp the beam and guide it to its precise final mating. Both manipulators are guided by position information from the roving eye.

This has been a brief overview of our software architecture and test bed. For more detail, see [Simmons et al., 2000].

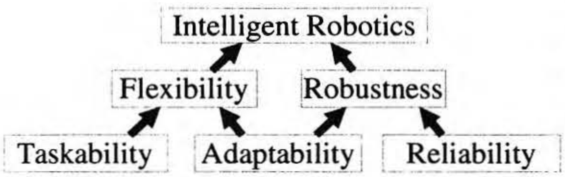

Figure 4: Graphical view of the taxonomy

3 Relating the Architecture to a Taxonomy of Intelligent Robotic Functionalities

We have attempted to break down intelligent robotics along the lines of what we see as some of its key challenges. The taxonomy builds from technologies to functionalities (figure 1). The bottom row of the figure shows three technologies:

- Taskability: A taskable system has components that can be recombined in different ways for different tasks.
- Adaptability: An adaptable system can learn about its environment and reconfigure itself to improve performance.
- Reliability: A reliable system can tolerate and accomodate component failures.

Together, these three technologies build two key functionalities, flexibility and robustness. A *flexible* system is general: it is able to perform a wide variety of tasks (taskability) in a wide variety of environments (adaptability). A *robust* system always gets the job done. It works in the face of a dynamic environment (adaptability) and component failures (reliability).

Our architecture primarily helps to support taskability and reliability. Implementing a robotic system as a heterogeneous team can lead to greater flexibility. A large group can divide itself into many different sub-teams, each with the particular robots appropriate for its task. Trying to combine all of the system's functionality into a single robot would make the design problem much harder. For example, in an exploration scenario, each sensor added to a rover makes it that much less manouverable. By using multiple rovers with different sensors, we can flexibly select which sensors should cover each area.

Another area we are researching is increasing reliability through distributed monitoring. Robots can monitor the performance of other robots, and notify them when there is a problem. The hierarchical structure of plans helps the executive to handle failures at the right level of detail in the plan. Through distributed monitoring of its actions, a robot gains access to additional sensors. Also, the team as a whole can compensate when a robot catastrophically fails and can't communicate its own status.

To sum up, we have tried to place the contributions of software architecture in the context of a general taxonomy for intelligent robotic technologies and functionalities.

4 Multi-Robot Systems in Future Space Missions

As missions in space become more ambitious, there will be an increasing need for multi-robot autonomy. Several proposed astronomical systems include multiple free-flying interferometers. The ST3 Starlight mission, planned to launch in 2005, will use two interferometers, and the Terrestrial Planet Finder (TPF), nominally launching in 2012, will use five. Both systems will have very tight constraints on the formation of the array. Interesting problems from the coordination perspective may include deciding how to distribute the relative position sensors and how to propagate their information (distributed monitoring). The system may also need to dynamically select which interferometers should move in response to an error in the formation.

The construction of massive space structures, such as space solar power stations, is an obvious candidate for multi-robot systems. For a detailed look at robotic research in this area, and an explanation for why we can expect robots to outperform humans in this kind of construction, refer to Sarjoun Skaff's paper in this volume.

There are also many planetary surface applications for robot teams. We will look closer at some proposed Mars scenarios. Deep drilling on Mars is currently an area of great interest. However, we can imagine that sending a single immobile drilling robot will have limited utility, as geologists can understand their data much better given greater context. A team of support robots which can deploy an array of seismographic sensors and study the surface features in the surrounding area could aid greatly in selecting drilling sites and in interpreting the resulting data.

The NASA reference mission for the human exploration of Mars [Hoffman and Kaplan, 1998] includes launching precursor missions to the surface years before the first human arrival, because long periods are needed to generate rocket propellant from in situ materials. Among the tasks that must be performed by robots before human arrival are:

- Scouting out good locations for habitat modules, ISRU propellant production plants, and nuclear reactors
- Moving the large modules into position
- Laying power cables
- Erecting a communications antenna
- Deploying greenhouse modules

All of these tasks will need to be performed with a high degree of autonomy, given light-speed round-trip communication delays of tens of minutes between Earth and Mars.

We can chart out several different functionalities required by the precursor robots, including:

- High mobility: needed to to initially locate routes to the base sites, and to perform initial geological studies before humans arrive.
- Gross manipulation: needed for unstowing, lifting, and moving large components like the communications tower.
- Fine manipulation: necessary to inspect and maintain modules, and to perform precision mating, latching, and unlatching.

Because of the difficulty involved in satisfying all of these functionalities with a single type of robot, there is good reason to believe that heterogeneous multi-robot systems will need to be employed for such a precursor mission.

We can conclude that heterogeneous multi-robot systems and autonomy in general have clear uses in future space missions.

5 Acknowledgments

We would like to thank the other members of the DIRA project team (Josue Ramos, Sule Yildirim, and Greg Armstrong), for their ideas and assistance in implementing the architecture. This research was sponsored in part by a grant from NASA (NAG9-1226).

References

[Albus et al., 1992] Albus, J., Bostelman, R., and Dagalakis, N. (1992). The NIST ROBOCRANE. *Journal of Robotic Systems*, 10(5).

[Bonasso et al., 1997] Bonasso, P., Kortenkamp, D., Miller, D., and Slack, M. (1997). Experiences with an architecture for intelligent, reactive agents. *Journal of Artificial Intelligence Research*, 9:1.

[Donald, 1995] Donald, B. (1995). Distributed robotic manipulation: experiments in minimalism. In *Proc. International Symposium on Experimental Robotics (ISER)*, pages 11–25. Springer-Verlag.

[Hoffman and Kaplan, 1998] Hoffman, S. J. and Kaplan, D. L., editors (1997 (updated 1998)). *Human Exploration of Mars: The Reference Mission of the NASA Mars Exploration Study Team.* Johnson Space Center, Houston, Texas.

[Jennings, 1993] Jennings, N. (1993). Specification and implementation of a Belief-Desire-Joint-Intention architecture for collaborative problem solving. *International Journal of Intelligent and Cooperative Information Systems*, 2(3):289–318.

[Mataric, 1992] Mataric, M. (1992). Distributed approaches to behavior control. In *Proc. SPIE Sensor Fusion V*, pages 373–382.

[Muscettola et al., 1998] Muscettola, N., Nayak, P. P., Pell, B., and Williams, B. (1998). Remote agent: To boldly go where no ai system has gone before. *Artificial Intelligence*, 103(1-2):5–48.

[Parker, 1998] Parker, L. (1998). Alliance: An architecture for fault tolerant multirobot coopera-tion. *IEEE Transactions on Robotics and Automation*, 14(2):220–240.

[Simmons et al.,] Simmons, R., Goodwin, R., Haigh, K., Koenig, S., and Sullivan, J. O. A layered archi-tecture for office delivery robots. In *Proc. First International Conference on Autonomous Agents*, Marina del Rey, CA.

[Simmons et al., 2000] Simmons, R., Singh, S., Hershberger, D., Ramos, J., and Smith, T. (2000). Coordination of heterogeneous robots for large-scale assembly. In *Proceedings of the 7th International Symposium on Experimental Robotics (ISER)*.

[Whittaker et al., 2000] Whittaker, W. L., Urmson, C., Staritz, P., Kennedy, B., and Ambrose, R. (2000). Robotics for assembly, inspection, and maintenance of space macrofacilities. In *AIAA Space 2000 Conference and Exposition*. AIAA.