Multiagent Learning for Autonomous Spacecraft Constellations

Peter Stone
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213
pstone@cs.cmu.edu
http://www.cs.cmu.edu/~pstone

Abstract

Achieving spacecraft autonomy, the goal of the NASA New Millennium Project, promises to increase the range of possible space missions while drastically reducing costs. However, *individual* spacecraft autonomy with *fixed* control algorithms is insufficient for meeting New Millennium goals. This paper presents the *layered learning* technique for flexible control in complex, real-time, multiagent environments. Layered learning has been successfully implemented in one such domain. Here, potential space-related applications are presented.

Introduction

Achieving spacecraft autonomy, the goal of the NASA New Millennium Project, promises to increase the range of possible space missions while drastically reducing costs. However, *individual* spacecraft autonomy with *fixed* control algorithms is insufficient for meeting New Millennium goals.

This paper presents multiagent learning techniques and flexible teamwork structures which are being developed in noisy, real-time environments. These techniques are applicable towards the realization of adaptivity, autonomy, and multiagent plan refinement in *constellations* of spacecraft. This new approach, multiagent *layered learning*, is a hierarchical, bottom-up method that makes use of expert domain decompositions to seamlessly integrate multiple learning modules.

A preliminary investigation and development of this strategy in a complex, noisy domain that requires real-time interleaving of planning and execution has shown promising empirical results in flexibly acquiring and using the learned knowledge at different levels of individual and team behaviors (Stone and Veloso 1998).

A specific potential space-related application is the control of constellations of interferometer-bearing spacecraft on long-range mapping missions. As well as increasing the range of possible missions and the size of possible fleets, spacecraft autonomy could reduce mission operations costs, exclusive of data analysis, by up to 60% (Ridenoure 1995), with even larger savings likely for multi-craft autonomy.

Significant multiagent learning challenges to be met in this endeavor include developing communication protocols among the spacecraft; developing decision-making methods that facilitate real-time reactions to unexpected opportunities and/or problems; coordination of multiple spacecraft in pursuit of a common goal; feedback monitoring systems to evaluate performance; and learning methods to adapt behavior over certain control variables. Since constellations may include different types of spacecraft, the organization of the spacecraft into different mission roles is an important task. Particularly in the event of spacecraft failure, it would be desirable to have a constellation that is capable of reorganizing itself in an appropriate way.

The rest of the paper is organized as follows. The next section presents layered learning, the general research methodology expounded in this paper. The following section presents space-related application areas in which this layered learning can be used. Then previous and current research regarding layered learning is summarized. The penultimate section presents related work and the final section concludes.

Methodology

Along with distributed techniques studied by all researchers in Multiagent Systems, space domains require real-time control in noisy environments. As of yet, there has been little work with multiagent systems in such complex domains. Because of the inherent complexity of this type of multiagent system, Machine Learning is an interesting and promising area to merge with Multiagent Systems. Machine Learning has the potential to provide robust mechanisms that leverage upon experience to equip agents with a large spectrum of behaviors, ranging from effective individual performance in a team, to collaborative achievement of independently and jointly set high-level goals.

The research presented here focuses on learning in this particularly complex class of multiagent domains. The principal question to be answered is

> **Can agents learn to work together in a real-time, noisy environment?**

To be precise, the prospects of using machine learning techniques to improve an agent's behavior are examined in domains with the following characteristics:

- several independent agents with the same well-defined high-level goal;
- a need for real-time decision-making;
- sensor and actuator noise.

The agents are assumed to have at their disposal the following resources:

- sensors of the environment that give partial, noisy information;
- the ability to process the sensory information and use it to update a world model;
- noisy actuators that affect the environment;
- low bandwidth communication capabilities;
- a low-level protocol for communication;

As a basis for learning at the multiagent level, agents are also organized into a flexible team structure that allows them to dynamically adjust their overall formation as well as their individual roles within the formations.

It is assumed that, due to the complexity of the environment, agents in domains with the above characteristics are unable to learn effective direct mappings from their sensors to their actuators, even when saving past states of the world. Thus, the approach taken is to break the problem down into several behavioral layers and to use machine learning techniques when appropriate. Starting with low-level behaviors, the process of creating new behavior levels and new machine learning subtasks continues until high level strategic behaviors that take into account both teammate and opponent strategies are reached. At every behavior level, the end product should be a method for choosing an action every time the agent gets sensory information. The appropriate behavior granularity and the aspects of the behaviors to be learned will be determined a priori as a function of the specific domain. This new approach is called *layered learning* (see Figure 1).

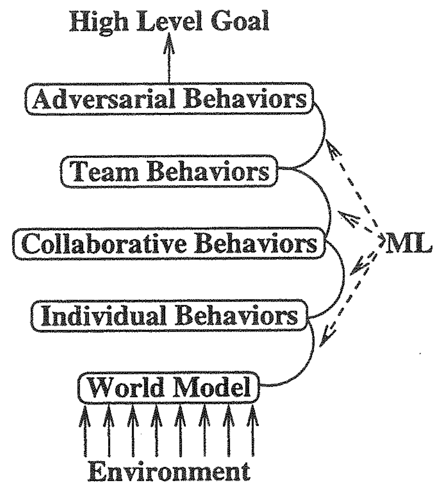

Figure 1: An overview of the layered learning framework. It is designed for use in domains that are too complex to learn a mapping straight from sensors to actuators. It uses a hierarchical, bottom-up approach. Not all multiagent domains require adversarial behaviors on top of the team behaviors.

Application Areas

For missions involving constellations of spacecraft to be truly autonomous, ***distributed, adaptive, autonomous control*** is necessary. Instead of centralized control of spacecraft with fixed control algorithms, methods are required for the coordination of adaptive spacecraft in a distributed fashion. Centralized control of the entire constellation from a single craft risks complete mission failure if just a single craft fails. Furthermore, distributed control facilitates variable fleet sizes, possibly up to a hundred spacecraft for a single mission (Chien 1996). The ability for autonomous spacecraft to learn, or adapt to their environments, would allow mission profiles to be framed more loosely and to apply to a wider range of environments (Chien 1996).

Interferometry for Imaging Distant Objects

A particularly good example of the potential benefits of multiagent learning techniques to space-related problems is interferometry missions for imaging distant objects[1]. As described in (Chien 1996), consider

> a network of interferometer bearing spacecraft in an earth-orbiting configuration, using solar power, with limited data storage, and an interferometer for imaging distant objects (such as planets around nearby stars).

In this scenario, distributed, adaptive techniques have several potential benefits:

- If any subset of the spacecraft fail or temporarily must devote their resources to spacecraft safety needs, the remaining spacecraft could continue performing interferometric mapping. Although they may need to shift to a different target, valuable observation time would not be wasted.
- If an unexpected scientific observation opportunity arises, some or all of the constellation could shift attention to the unexpected event. Lower priority goals could be handled with remaining resources or deferred.
- If any of the spacecraft in the constellation begins degrading such that it is still operational, but with limited capacities or increased time needs for certain activities, an adaptive planner could reconfigure the

[1]For example, the New Millennium Deep Space 3 Mission is a constellation of interferometer bearing spacecraft.

constellation so as to make optimal use of each of the spacecraft to maximize interferometric mapping capabilities.

Notice that in this application area the constellation must be able to handle *noisy* real-world data about spacecraft state and imaging conditions. Furthermore, the constellation should be able to respond to unexpected events quickly, or in *real-time*. Otherwise opportunities may pass before they are seized.

Other Opportunities

Several other potential multiagent learning applications exist in real-time, noisy situations:

- As NASA moves towards more frequent missions, the frequency of inter-space meetings between multiple shuttles and space stations will also increase. Distributed scheduling is useful (i) so that the mission is robust to the failure of any individual spacecraft (i.e. the one with the scheduler), and (ii) to avoid the necessity of forwarding large amounts of local data to a centralized source. However, if each shuttle and each station is equipped with its own scheduler, then there will need to be methods for these schedulers to coordinate their plans so that all constraints and goals can be met.
- Within a complex system such as a space station, the same distributed, adaptive systems can offer the same advantages as they do for interferometric missions. If one subsystem fails or degrades, appropriate and timely reconfigurations of goals and/or resource allocations may be necessary.
- On planetary missions, it may be desirable to deploy several rovers to explore, possibly out of radio range from a mothership. Again, multiagent learning in real-time, noisy situations could increase the effectiveness or enable the eventual success of such missions.

Previous Work

Current and past research on layered learning is primarily in the emerging domain of simulated Robotic Soccer (Kitano *et al.* 1997). An advantage of using this domain instead of immediately developing techniques in a space-related application area is that the simulator already exists and it is being used by several researchers around the world. Thus competing systems are being developed elsewhere, which makes realistic testing particularly accessible.

Layered learning has already been successfully implemented in the robotic soccer domain (Stone and Veloso 1998). First, clients (i.e. programmed players) use a neural network to learn to intercept a moving ball. Second, clients use a decision tree to learn the likelihood of completing a pass to a given teammate. A key aspect of these two behaviors is that the clients use the trained neural network when attempting to receive the ball during decision tree training (Stone and Veloso 1998). Such interaction of learned behaviors at different behavior levels is essential to layered learning.

Additional high-level learned behaviors are also being developed. Clients can use the output of the decision tree to choose whether to pass, shoot, or dribble the ball in a given situation. They can also learn to position themselves effectively on the field, both individually and as a team.

These positioning behaviors entail strategic reasoning to counteract opponent strategies. As a basis for this type of behavior, a flexible formation structure has been developed (Stone and Veloso 1997). In this structure, each agent has the knowledge required to fill any team role in any of several team formations. Furthermore, the agents are equipped with low-level communication protocols allowing for seamless formation adjustments and dynamic role adjustments. The agents are also equipped with several pre-defined multiagent, multi-step plans that can be instantiated in the appropriate situations. This flexible formation has been successfully implemented on real robots as well as in simulation (Stone and Veloso 1997). Several learning opportunities exist within this flexible formation structure, including when to switch formations and/or positions and how to refine the pre-defined plans based on the current state of the world.

The existing learned behaviors in the robotic soccer domain could naturally map onto spacecraft constellation control. A neural network could be trained to do low level control of behaviors, while a decision tree could allow goal-driven commanding in a symbolic form. Higher-level collaborative behaviors could then be used to control multi-craft interactions such as relative positioning.

Of course, machine learning should not be used for processes that are easily automated without it. Layered learning prescribes the use of learning when hand-coding is non-trivial. Thus, low-level spacecraft control may not need to be learned. On the other hand, past control algorithms have been developed only with significant time and effort. Machine learning could potentially reduce both the time and effort required to develop such algorithms for new spacecraft. Furthermore, the complexity of coordinating the control of several spacecraft suggests many multiagent learning opportunities at the higher levels of control.

Related Work

The layered learning approach is somewhat reminiscent of Brooks' Subsumption Architecture (Brooks 1986) which layers control modules, allowing high-level controllers to override lower-lever ones. Each control level is capable of controlling the robot on its own up to a specified level of functionality. Brooks implements his approach on real robots, building controllers for simple tasks such as avoiding collisions and wandering around.

Mataric brings the Subsumption Architecture to a multiagent learning domain, building controllers on top of a set of learned *basis behaviors* (Mataric 1995). Mataric's basis behaviors are chosen to be necessary and sufficient for the learning task, while remaining as simple and robust as possible. Since Mataric's robots were to learn social behaviors such as flocking and foraging, they were equipped with basis behaviors such as the ability to follow each other and the ability to wander without running into obstacles.

While layered learning also makes use of multiple behavior layers, the tasks considered are much more complex: the agents must be able to generalize across situations, handle adversaries, and achieve complex goals. In order to move quickly to high-level behaviors, the commitment to have every layer be completely able to control the robot is abandoned. Instead, many situation-specific (but as general as possible) behaviors are produced which are then managed by higher-level behaviors. Nevertheless, the idea of building higher levels of functionality on top of lower levels is retained. It is in producing the situation-specific behaviors that machine learning techniques are used.

In many multiagent domains, there is the need or opportunity for teammates to assume different roles in a joint endeavor. As described above, a flexible teamwork structure has been developed. In related work, Tambe discusses a framework in which agents can take over the roles of other teammates in a helicopter-combat domain (Tambe 1996a). In the learning context, Prasad et al. have created design agents that can learn which role to fill (Prasad *et al.* 1996).

In addition to reasoning about roles of teammates, Tambe's combat agents can also reason about the roles that opponents are playing in team behaviors (Tambe 1996b). By recognizing an opponent's action as a part of a larger team action, an agent is able to more easily make sense of the individual opponent's behavior with the goal of being able to predict the opponent's future actions. Tambe's work enhances previous work that aims at having agents deduce other agents' intentions through observation (Huber and Durfee 1995).

Conclusion

Layered learning as well as its associated flexible team structure are proving themselves useful in the robotic soccer domain. Since many of the complexities of this domain match the complexities of several space-related applications, there is great promise of transferring the layered learning methodology to such applications. In particular, applying multiagent learning techniques to the goal of achieving spacecraft constellation autonomy is an exciting prospect, both because it has the potential to increase the range of possible missions and because it could significantly reduce mission budgets.

Future work includes identifying the precise levels of spacecraft behavior for which machine learning will be useful. Beginning at the lowest behavior-levels which are difficult to hand-code, learned layers can then be built in a bottom-up, hierarchical manner.

Acknowledgements

This research is conducted jointly with Dr. Manuela Veloso. Thanks to Dr. Steve Chien for his contributions and comments.

References

Rodney A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, RA-2:14–23, 1986.

Steve Chien. Distributed, adaptive, self-commanding technology for spacecraft autonomy, 1996. New Technology Document for the New Millennium Autonomy IPDT.

Marcus J. Huber and Edmund H. Durfee. Deciding when to commit to action during observation-based coordination. In *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95)*, pages 163–170, Menlo Park, California, June 1995. AAAI Press.

Hiroaki Kitano, Yasuo Kuniyoshi, Itsuki Noda, Minoru Asada, Hitoshi Matsubara, and Ei-Ichi Osawa. Robocup: A challenge problem for ai. *AI Magazine*, 18(1):73–85, Spring 1997.

Maja J. Mataric. Designing and understanding adaptive group behavior. *Adaptive Behavior*, 4(1), December 1995.

M V Nagendra Prasad, Victor R. Lesser, and Susan E. Lander. Learning organizational roles in a heterogeneous multi-agent system. In *Adaptation, Coevolution and Learning in Multiagent Systems: Papers from the 1996 AAAI Spring Symposium*, pages 72–77, Menlo Park,CA, March 1996. AAAI Press. AAAI Technical Report SS-96-01.

R. Ridenoure, June 1995. New Millenium Mission Operations Study.

Peter Stone and Manuela Veloso. Task decomposition and dynamic role assignment for real-time strategic teamwork. In *submitted to the Third International Conference on Multi-Agent Systems*, November 1997. Draft available by request or from http://www.cs.cmu.edu/ pstone/pstone-papers.html.

Peter Stone and Manuela Veloso. A layered approach to learning client behaviors in the robocup soccer server. *To appear in Applied Artificial Intelligence (AAI) Journal*, 1998.

Milind Tambe. Teamwork in real-world, dynamic environments. In *Proceedings of the Second International Conference on Multi-Agent Systems (ICMAS-96)*, Menlo Park, California, December 1996. AAAI Press.

Milind Tambe. Tracking dynamic team activity. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, Menlo Park, California, 1996. AAAI Press.