Reconfiguration of Multi-Agent Planning Systems*

Diane J. Cook
University of Texas at Arlington
Department of Computer Science Engineering
Arlington, TX 76019
cook@cse.uta.edu

Abstract

Many multi-agent planning systems can perform more effectively than single-agent systems due to redundancy, concurrency, and heterogeneous capabilities. However, lack of an effective coordination mechanism can diminish the advantages of using multiple agents. Researchers have suggested a variety of coordination configurations, such as centralized planning, distributed planning, and local planning. While each configuration offers some advantages, no single planning configuration is optimal for all planning scenarios.

In this paper, we identify the need for reconfiguration in multi-agent planning systems. Instead of selecting just one planning configuration, we design a system that can make use of a number of planning modes. In order to determine the best configuration for a given problem, a decision-theoretic model is designed. This models selects the configuration that will maximize the expected payoff for the system. Reconfiguration of the multi-agent system can be initiated by a single or by multiple agents, and can occur between plans or during plan execution.

Introduction

Multi-agent planning systems can offer advantages of distributed capabilities, concurrency, and redundancy over single-agent planning systems. However, without the proper assignment of tasks to agents and without an effective structure for coordination and communication, the advantages of concurrency and distributed capabilities cannot be realized. In fact, if coordination is not performed well, agents will destructively interfere with each other's work instead of contributing toward the overall goal of the system.

Researchers to date have proposed a variety of plan generation and plan coordination architectures. Although each approach has its own advantages and disadvantages, no one planning configuration will prove optimal for all situations.

*This work is supported in part by the Advanced Research Projects Agency and by the National Science Foundation.

We propose to take advantage of each of the three common multi-agent planning configurations: centralized, distributed, and local planning. By identifying the advantages of each approach, we can build a decision-theoretic model to select the planning configuration that will maximize the overall utility of the system.

In the next section, we review three common multi-agent planning configurations. We then motivate the need for reconfiguration of multi-agent systems, and describe our proposed approach. Finally, we evaluate the approach and discuss areas for future research.

Multi-Agent Planning Configurations

There are three common approaches to generating and coordinating multi-agent plans. Each of these configurations are defined by the choice of agents responsible for plan generation and coordination and the methods used to assign and coordinate multi-agent tasks. In the centralized approach to multi-agent plan generation, one agent acts as the plan generator and coordinator. Though individual entities may form their own plans, the central agent analyzes the plans for potential conflicts and manages the synchronization of the individual plans. The relationship between agents in each of these configurations is shown in Figure 1.

Georgeff (1986) offers a formalism by which a central planning agent can reason about interactions between agents and their tasks. This formalism extends situation calculus to explicitly reason about simultaneous events and effects of actions on other actions. Kamel and Syed (1992) use a single planner to distribute tasks among homogeneous agents in a way that maximizes overall performance. They do not, however, consider the problem of resolving conflicts between these agents. Cammarata, McArthur and Steeb (1983) also rely upon a central agent to coordinate an arbitrary number of plans. In this case, each agent represents the actions for one aircraft, and a conflict between agents exists when the agents violate minimum separation requirements.

Centralized planning offers the advantages of reduced communication costs and effective management

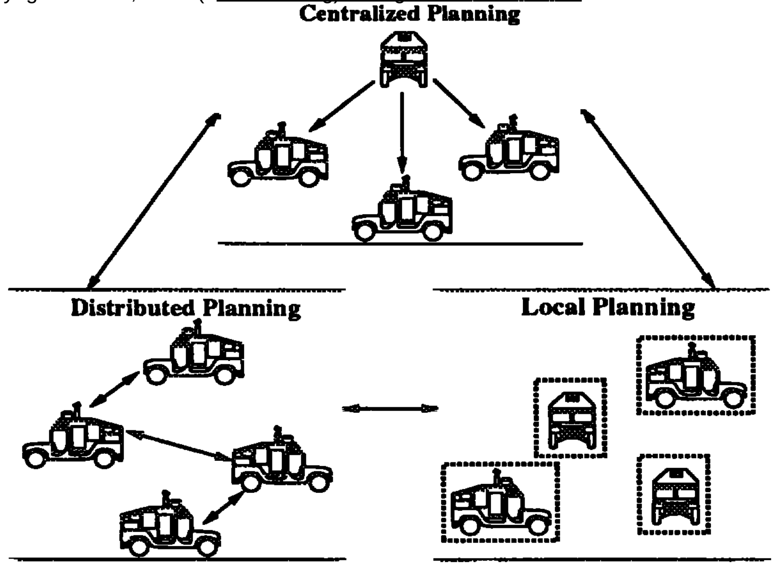

Figure 1: Centralized, Distributed, and Local Planning Configurations

of resources and agents (if the central agent has perfect knowledge about the agents' capabilities and environments). However, the ability to generate and effectively coordinate plans is dependent on a sole agent. If that agent is damaged or otherwise inaccessible, then the entire mission is jeopardized. In addition, the traffic surrounding the central agent can cause significant delays in the overall plan coordination and distribution process.

In the distributed planning approach, agents are responsible for generating and coordinating their own actions using information obtained from other agents about their intentions and approaches. When multiagent planning is performed in a distributed manner, agents may not have a global view of the environment and agents' individual goals, so detecting and resolving interactions is more difficult. Communication costs tend to be much greater in distributed planning systems, and thus conflict resolution may in some cases be slower. On the other hand, the approach is more robust because it does not depend on a single central agent. Agents can respond more quickly to their own needs and the needs of neighboring agents when each decision does not have to be routed through a central decision maker.

Corkill (1979) has developed a distributed hierarchical planner where agents develop models of their neighbors. The agents build plans cooperatively by synchronizing actions at each level of the hierarchical procedural net. In a similar fashion, von Martial (1991) allows agents to coordinate activities at a high level of abstraction before deciding on a specific course of action. Von Martial detects both positive and negative interactions between distributed agents. Durfee and Montgomery (1991) also coordinate plans hierarchically. In this approach, abstract plans are communicated among agents and are decomposed along the dimensions of *space*, *time*, *agents*, *tasks*, *resources*, and *goals* only when the potential for conflicts exists in the abstract plan.

Moehlman and Lesser (1990) demonstrate techniques for explicit negotiation of resources in the Phoenix fire-fighting system. Here, agents may only have local views of the world, but decisions are always made cooperatively and with a view toward achieving high-priority global goals. While most of the distributed planning systems eventually rely on a ranking of agents, the approach described by Ephrati and Rosenschein (1993) suggests using a voting mechanism to collectively select a set of actions from among an agent's preferences. Although distributed plan generation and coordination is an expensive process, Kosoresow (1993) is able to determine bounds for convergence times in distributed planning systems where agents iteratively post and synchronize their plans. Plans are revised between postings using a Markov process.

In multi-agent systems that use local plan generation and coordination techniques, decision making is performed exclusively within each agent. Each local agent may request and send information to other agents, but the final plan is generated completely by the individual entity. In the most simple case, classical single-agent planning systems may act as local planners in a multi-agent environment. However, because no collaborative effort toward synchronizing plans is performed, agents must monitor plan execution and be prepared to dynamically adjust their plans during execution. This approach minimizes the amount of communication required between agents, and does not encounter plan generation delays due to synchronization. On the other hand, the performance of plan execution is usually quite poor due to the fact that agents have to work around other active entities in their environment.

Sycara et al. (1990) adapt single-agent planning techniques to multi-agent environments by using *textures* to graphically define the complexity and importance of local decisions. By using textures to predict the impact of local decisions on global system goals, local planning can be performed with a focus toward achieving and maintaining global goals. Other researchers have introduced methods of reducing possible conflicts in local planning by introducing execution policies. Lansky and Fogelsong (1987) identify *regions of activity* in the environment where conflicts are likely to occur, and define constraints for acting within that region. Shoham and Tennenholtz (1992) define social laws of interaction that reduce conflicts, in the way that traffic laws reduce harmful interactions between vehicles in society.

Reconfiguration of systems

Reconfiguration of a multi-agent planning system is not only necessary, it is also a common practice in domains such as military plan generation and execution. A configuration that satisfies one goal may be totally inappropriate once the goal is satisfied or when the mission is interrupted by an emergency situation. For example, Figure 2 shows a Tactical Road March task that is interrupted by detection of multiple targets. Because of this unforeseen threat, the original planning and coordination configuration are no longer desirable. Instead, the agents may need to spread out and cover the area to flush out all remaining targets. This change in configuration affects the communication mechanism, the method of generating task plans, and the mode of control.

We are designing a decision-theoretic model that determines the optimal planning configuration given parameters of the system configuration, the planning environment and the current task. The choice of planning configuration is guided by the principle of utility maximization. Whichever planning mode maximizes the expected payoff given the current system parameters is selected.

The utility for a particular planning configuration is calculated as a function of the weighted system parameters. The parameters are represented by P_i^c, where c represents a given planning configuration (centralized, distributed, or local) and P_i represents one of the six parameters listed below.

- P_1 = The communication bandwidth between agents,
- P_2 = Number of resources (potential for resource contention),
- P_3 = Number of agents active in the system,
- P_4 = The number of steps in the plan to be coordinated,
- P_5 = The reliability of the agents, and
- P_6 = The cost of adjusting the plan (if the plans are not fully synchronized during generation).

The utility of configuration c is calculated as

$$Utility(c) = F(w_i^c P_i^c),$$

where i ranges over the system parameters. While F may represent a number of combination functions, we are experimenting first with a linear combination of parameters, or

$$Utility(c) = \sum_i (w_i^c P_i^c).$$

We have implemented plan generation systems for each of the three main configurations: centralized, distributed, and local planning. The appropriate weights w_i for each of these system parameters can been determined by generating artificial planning test cases and calculating the system payoff for each planning mode as the system parameters change. This decision-theoretic model can be used to select a configuration at the beginning of the plan generation. The model can also be used to reevaluate the current planning configuration if the system parameters or environment changes.

All three planning configurations contain the components of detection, planning, and control. The first aspect of reconfiguration is detection of the need to reconfigure. For example, if one agent senses a threat, it may be useful for several agents to come to the aid of the threatened entity. The detection algorithm will alert the system of the need to reconfigure and will give its rationale. The second aspect of the process is to plan the reconfiguration. Reconfiguration may involve changing the current plans of individual agents. This transition must be made gracefully, either finding a stopping point in the old plan and initiating the new, or interleaving steps of the two plans until the transition is complete. The controller manages the reconfiguration once it is initiated, making sure that the desired new configuration is achieved.

Figure 3 identifies the agent(s) responsible for detecting and controlling a system reconfiguration. For example, a switch from centralized mode to local mode is determined by a command from the leader, or by the end of the current task and the beginning of a new task in local mode, or by a predetermined time of day. On the other hand, the system may be reconfigured from distributed to local mode should a single agent request the switch (possibly due to detection of a nearby threat).

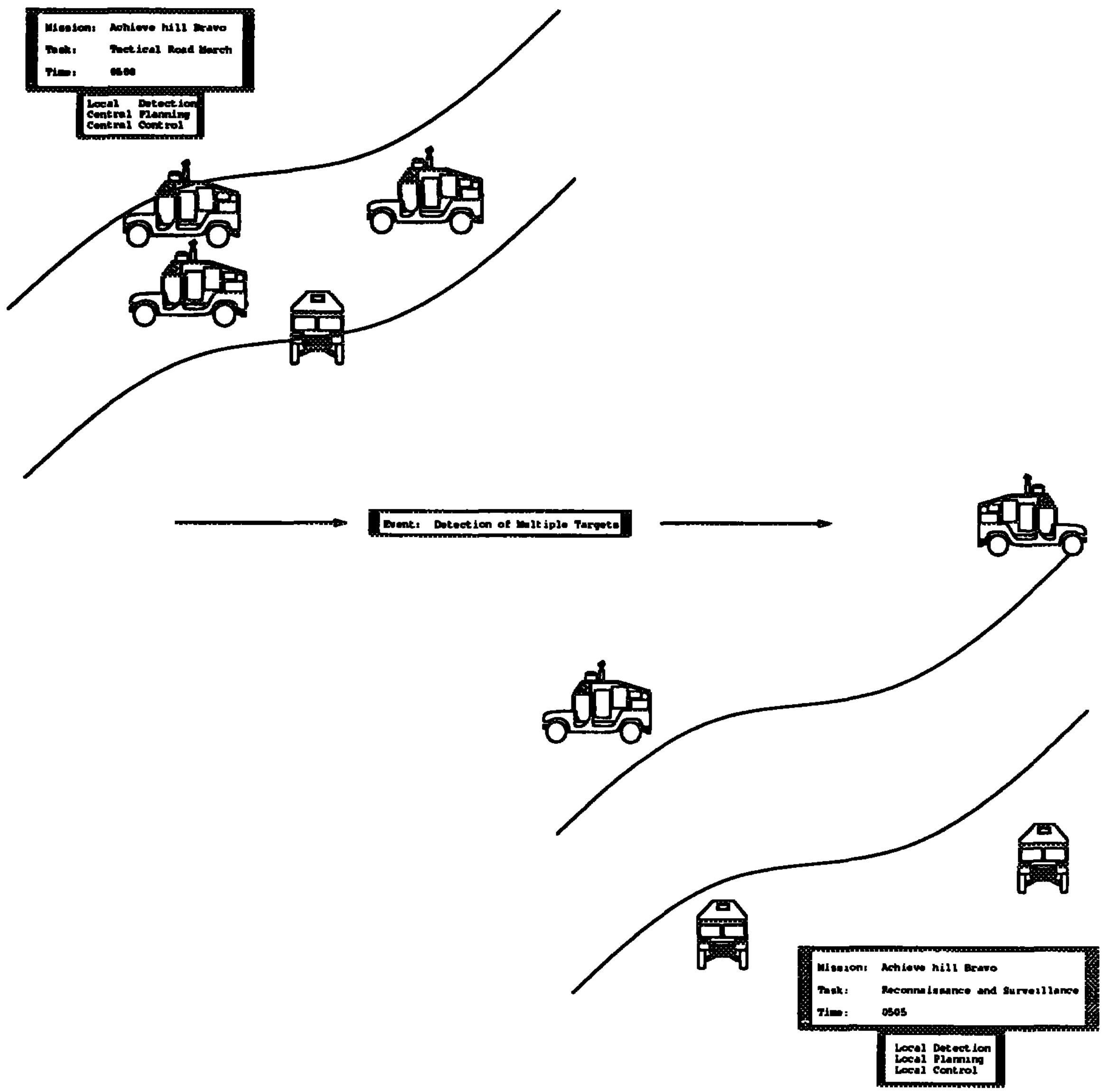

Figure 2: Need for reconfiguration

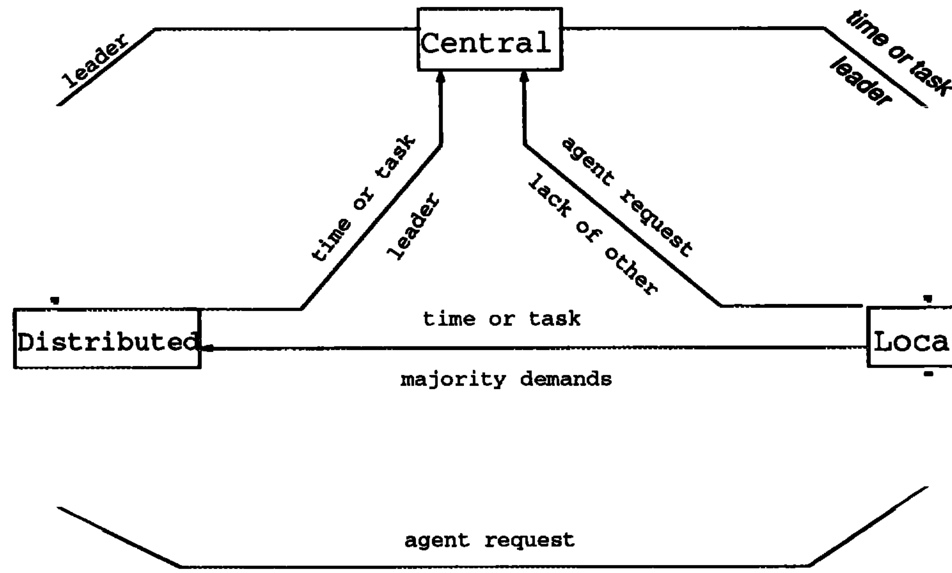

Figure 3: Control of reconfiguration

It is possible that detection of a reconfiguration need, planning for the reconfiguration, and control of reconfiguration may all be executed by different groups of agents. The table below shows a possibility where a local agent detects the need for reconfiguration, and a leader generates and controls the plan for reconfiguration. Allowing for all of these combinations is an area we will pursue in the future.

	Central	Distributed	Local
Detection			x
Planning	x		
Control	x		

Conclusions

In this paper, we have described a methodology for using a variety of planning configurations in multi-agent planning systems. Each of the three common planning configurations – centralized planning, distributed planning, and local planning – offer significant advantages and disadvantages. These tradeoffs prevent one single configuration from being optimal for all scenarios.

We are researching a method for incorporating all of these planning configurations into a multi-agent system. By building a decision-theoretic model that calculates the payoff for each configuration, the best configuration for a given plan can be selected. By controlling the switch between planning modes, reconfiguration can be performed between entire plans or during plan execution.

There are numerous directions for future research in this area. We will discuss two such extensions in this section. As suggested in the paper, detection, planning and control of reconfiguration itself can be initiated and directed by various groups of agents. Considering these tradeoffs and incorporating the results into our system is an area for future research.

This paper described a model that will select one of the three described planning configurations. In reality, there are many such configurations. Many of these configurations are a hybrid. For example, all agents in one room may best be controlled by a central authority, while agents in another room plan most effectively in distributed mode. A future version of our model will be able to select any combination of modes that are plotted along a continuum of planning configurations.

References

Cammarata, S.; McArthur, D.; and Steeb, R. 1983. Strategies of cooperation in distributed problem solving. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, 767–770.

Corkill, D. D. 1979. Hierarchical planning in a distributed environment. In *Proceedings of the Sixth International Joint Conference on Artificial Intelligence*, 168–175.

Durfee, E. H., and Montgomery, T. A. 1991. Coordination as distributed search in a hierarchical behavior space. *IEEE Transactions on Systems, Man, and Cybernetics.*

Ephrati, E., and Rosenschein, J. S. 1993. Multi-agent planning as a dynamic search for social consensus. In

Proceedings of the International Joint Conference on Artificial Intelligence, 423–429.

Georgeff, M. 1986. A representation of events in multi-agent domains. In *Proceedings of the National Conference on Artificial Intelligence*, 125–129.

Kamel, M., and Syed, A. 1992. Task decomposition and plan generation in multiagent domains. *Computer-Integrated Manufacturing* 9(1):71–83.

Kosoresow, A. P. 1993. A fast first-cut protocol for agent coordination. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, 237–242.

Lansky, A. L., and Fogelsong, D. 1987. Localized representation and planning methods for parallel domains. In *Proceedings of the National Conference on Artificial Intelligence*, 240–245.

Moehlman, T., and Lesser, V. 1990. Cooperative planning and decentralized negotiation in multi-fireboss phoenix. In *Proceedings of the Workshop on Innovative Approaches to Planning, Scheduling and Control*, 144–159.

Shoham, Y., and Tennenholtz, M. 1992. On the synthesis of useful social laws for artificial agent societies. In *Proceedings of AAAI*, 276–281.

Sycara, K.; Roth, S.; Sadeh, N.; and Fox, M. 1990. Managing resource allocation in multi-agent time-constrained domains. In *Proceedings of the Workshop on Innovative Approaches to Planning, Scheduling and Control*, 240–250.

von Martial, F. 1991. *Coordinating Plans of Autonomous Agents*. Springer-Verlag.