Learning Organizational Roles in a Heterogeneous Multi-agent System*

M V Nagendra Prasad[1], Victor R. Lesser[1] and Susan E. Lander[2],

[1]Department of Computer Science
University of Massachusetts
Amherst, MA 01003
{nagendra,lesser}@cs.umass.edu

[2]Blackboard Technology Group, Inc.
401 Main Street
Amherst, MA 01002
lander@bbtech.com

Abstract

In this paper, we present a heterogeneous multi-agent system called L-TEAM that highlights the potential for learning as an important tool for acquiring organizational knowledge in heterogeneous multi-agent systems. No single organization is good for all situations and it is almost impossible to anticipate all the situations and provide the agents with capabilities to react to each of them appropriately. L-TEAM is our attempt to study ways to alleviate this enormous knowledge acquisition bottle-neck.

Introduction

As Distributed Artificial Intelligence matures as a field, the complexity of the applications being tackled are beginning to challenge some of the common assumptions such as homogeneity of agents and tightly integrated coordination. Reusability of legacy systems and heterogeneity of agent representations demand a reexamination of many of the key assumptions about the amount of sharable information and the types of protocols for coordination. Lander and Lesser (Lander & Lesser 1994) developed the TEAM framework to examine cooperative search among a set of heterogeneous reusable agents. TEAM is an open system assembled through minimally customized integration of a dynamically selected subset from a catalogue of existing agents. Reusable agents may be involved in systems and situations that may not have been explicitly anticipated at the time of their design. Each agent works on a specific part of the overall problem. The agents work towards achieving a set of local solutions to different parts of the problem that are mutually consistent and that satisfy, as far as possible, the global considerations related to the overall problem.

TEAM was introduced in the context of parametric design in multi-agent systems. Each of the agents has its own local state information, a local database with static and dynamic constraints on its design components and a local agenda of potential actions. The search is performed over a space of partial designs. It is initiated by placing a problem specification in a centralized shared memory that also acts as a repository for the emerging composite solutions (i.e. partial solutions) and is visible to all the agents. Any design component produced by an agent is placed in the centralized repository. Some of the agents initiate base proposals based on the problem specifications and their own internal constraints and local state. Other agents in turn extend and critique these proposals to form complete designs.

An agent may detect conflicts during this process and communicate feedback to the relevant agents; augmenting their local view of the composite search space with meta-level information about its local search space to minimize the likelihood of generating conflicting solutions(Lander & Lesser 1994). For a composite solution in a given state, an agent can play one of a set of organizational roles (in TEAM, these roles are *solution-initiator*, *solution-extender*, or *solution-critic*). An organizational role represents a set of operators an agent can apply to a composite solution. An agent can be working on several composite solutions concurrently. Thus, at a given time, an agent is faced with the problem of: 1) choosing which solution to work on; and 2) choosing a role from the set of allowed roles that it can play for that solution. This decision is complicated by the fact that an agent has to achieve this choice within its local view of the problem-solving situation.

The objective of this paper is to investigate the utility of machine learning techniques as an aid to the decision processes of agents that may be involved in problem-solving situations not necessarily known at the time of their design. The results in our previous paper(NagendraPrasad, Lesser, & Lander 1995b) demonstrated the promise of learning techniques for such a task in L-TEAM, which is a learning version of TEAM. In this paper, we present further experimental details of our results and some new results since.

*This material is based upon work supported by the National Science Foundation under Grant Nos. IRI-9523419 and EEC-9209623. The content of this paper does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

Organizational Roles in Distributed Search

Organizational knowledge can be described as a specification of the way the overall search should be organized in terms of which agents play what roles in the search process and communicate what information, when and to whom. It provides the agents a way to effectively and reliably handle cooperative tasks. Each agent in L-TEAM plays some organizational role in distributed search. A role is a task or a set of tasks to be performed in the context of a single solution. A role may encompass one or more operators, e.g., the role *solution-initiator* includes the operators *initiate-solution* and *relax-solution-requirement.* A pattern of activation of roles in an agent set is a role assignment. All agents need not play all organizational roles; which in turn implies that agents can differ in the kinds of roles they are allotted. Organizational roles played by the agents are important for the efficiency of a search process and the quality of final solutions produced.

To illustrate the above issue, we will use a simple, generic two-agent example and their search and solution spaces as shown in Figure 1. The shaded portions in the local spaces of the agents A and B are the local solution spaces and their intersection represents the global solution space. It is clear that if agent A initiates and agent B extends, there is a greater chance of finding a mutually acceptable solution. Agent A trying to extend a solution initiated by Agent B is likely to lead to a failure more often than not due the small intersection space versus the large local solution space in Agent B. Note however, that the solution distribution in the space is not known a priori to the designer to hand code good organizational roles at the design time.

During each cycle of operator application in TEAM, each agent in turn has to decide on the role it can play next, based on the available partial designs. An agent can choose to be an *initiator* of a new design or an *extender* of an already existing partial design or a *critic* of an existing design. The agent needs to decide on the best role to assume next and accordingly construct a design component. Due the complexity and uncertainty associated with most real-world organizations, there may be no single organizational structure for all situations. This makes it imperative that the agents adapt themselves to be better suited for the current situation, so as to be effective. This paper investigates the effectiveness of learning situation-specific organizational roles assignments. Roles for the agents are chosen based on the present problem solving situation (we will discuss situations in more detail in the following section).

Learning Organizational Roles

The formal basis for learning search strategies adopted in this paper is derived from the UPC formalism for search control (see (Whitehair & Lesser 1993)) that relies on the calculation and use of the **U**tility, **P**robability and **C**ost (UPC) values associated with each $\langle state, op, final\ state \rangle$ tuple. The Utility component represents the present state's estimate of the final state's expected value or utility if we apply operator *op* in the present state. Probability represents the expected uncertainty associated with the ability to reach the final state from the present state, given that we apply operator *op*. Cost represents the expected computational cost of reaching the final state. Additionally, in the complex search spaces for which the UPC formalism was developed, application of an operator to a state does more than expand the state; it may also result in an increase in the problem solver's understanding of the interrelationships among states. In these situations, an operator that looks like a poor choice from the perspective of a local control policy may actually be a good choice from a more global perspective due to some increased information it makes available to the problem solver. This property of an operator is referred to as its *potential* and it needs to be taken into account while rating the operator. An evaluation function defines the objective strategy of the problem solving system based on the UPC components of an operator and its potential.

We modify the UPC formalism for the purpose of learning organizational roles for agents. All the possible states of the search are classified into a pre-enumerated finite class of situations. These classes of situations represent abstractions of the state of a search. Thus, for each agent, there is a UPC vector per situation per operator leading to a final state. A situation in L-TEAM is represented by a feature vector whose values determine the class of a state of the search. In L-TEAM, an agent responsible for decision making at the node retrieves the UPC values based on the situation vector for all the roles that are applicable in current state. Depending on the objective function to be maximized, these UPC vectors are used to choose a role to be performed next.

We use the *supervised-learning approach* to prediction learning (see (Sutton 1988)) to learn estimates for the UPC vectors for each of the states[1].

Obtaining measures of potential is a more involved process requiring a certain understanding of the system. For example, in L-TEAM the agents can apply operators that lead to infeasible solutions due to conflicts in their requirements. However, this process of running into a conflict leads to certain important consequences such as the exchange of violated constraints. The constraints an agent receives from other agents aid that agent's subsequent search in that episode by letting it relate its local solution requirements to more global requirements. Hence, the operators leading to conflicts followed by information exchange are

[1]For details, the reader is referred to Nagendra Prasad et. al.(NagendraPrasad, Lesser, & Lander 1995a)

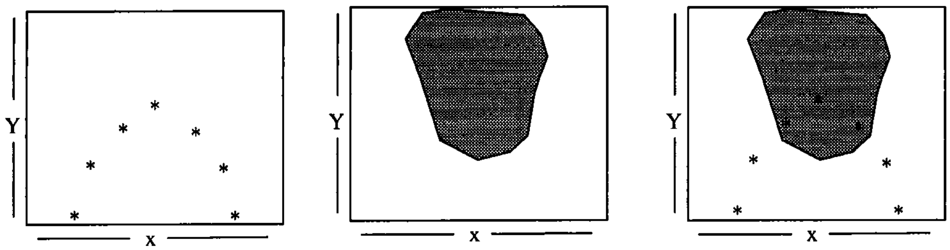

Figure 1: Local and Composite Search Spaces

rewarded by potential. Learning algorithm for the potential of an operator again uses *supervised-learning approach* to prediction learning and is similar to that for utility.

Experiments

To demonstrate the effectiveness of the mechanisms in L-TEAM and compare them to those in TEAM, we used the same domain as in (Lander & Lesser 1994) — parametric design of steam condensers. The prototype multi-agent system for this domain, built on top of the TEAM framework, consists of seven agents: pump-agent, heat-exchanger-agent, motor-agent, vbelt-agent, shaft-agent, platform-agent, and frequency-critic. Problem specification consists of three parameters — required capacity, platform side length, and maximum platform deflection.

Each agent can play a single organizational role in any single design. In this paper, we confine ourselves to learning the appropriate application of two roles in the agents - *solution-initiator* or *solution-extender*. Four of the seven agents — pump-agent, motor-agent, heat-exchanger-agent, and Vbelt-agent — are learning either to be a *solution-initiator* or *solution-extender* in each situation. The other three agents have fixed organizational roles — platform and shaft agents always extend and frequency-critic always critiques.

In the experiments reported below, the situation vector for each agent had three components. The first component represented changes in the global views of any of the agents in the system. If any of the agents receives any new external constraints from other agents in the past m time units (m is set to 4 in the experiments), this component is '1' for all agents. Otherwise it is '0'. If any of the agents has relaxed its local quality requirements in the past n time units (n = 2) then the second component is '1' for all agents. Otherwise it is '0'. Typically, a problem solving episode in L-TEAM starts with an initial phase of exchange of all the communicable information involved in conflicts and then enters a phase where the search is more informed and all the information that leads to conflicts and can be communicated has already been exchanged. This phase shift is represented in the situation vector through the third component: during the initial phase of conflict detection and exchange of information, it is '0' while in the second phase, it is '1'. We used the following objective evaluation function for rating an organizational role:

$$f(U, P, C, potential) \quad = \quad U * P + potential$$

We trained L-TEAM on 150 randomly generated design requirements and then tested L-TEAM and TEAM pairwise on 100 randomly generated design requirements different from those used for training. TEAM was set up so that heat-exchanger and pump agents could either initiate a design or extend a design whereas Vbelt, shaft and platform agents could only extend a design. In TEAM, an agent initiates a design only if there are no partial designs on the blackboard that it can extend. The performance parameter we analyzed was the cost of the best design produced (lowest cost).

We ran L-TEAM and TEAM in two ranges of the input parameters. Range 1 consisted of required-capacity 50 - 1500, platform-side-length 25 - 225, platform-deflection 0.02 - 0.1. Range 2 consisted of required-capacity 1750 - 2000, platform-side-length 175 - 225, platform-deflection 0.06 - 0.1. Lower values of required-capacity in Range 1 represented easier problems. We chose the two ranges to represent "easy" and "tough" problems. Table 2 represents the same organization learned by non-situation-specific TEAM in both the ranges. One can see from Table 3 and Table 4 that the two learned organizations for Range 1 and Range 2 are different. Table 1 shows the average design costs for the three systems - situation-specific L-TEAM (ss-L-TEAM), non-situation-specific L-TEAM (ns-L-TEAM), and TEAM - over the 2 ranges.

Wilcoxon matched-pair signed-ranks test revealed significant differences between the cost of designs produced by all the pairs in the table except between situation-specific L-TEAM and non-situation-specific L-TEAM in Range 1[2] and between non-situation-

[2]Easy problems may not gain by sophisticated mechanisms like situation-specificity

Range	ss-L-TEAM	ns-L-TEAM	TEAM
Range 1	5587.6	5616.2	5770.6
Range 2	17353.75	17678.97	17704.70

Table 1: Average Cost of a Design

agent	pump agent	heatx agent	motor agent	vbelt agent	shaft agent	platform agent	frequency critic
roles	initiator	extender	extender	extender	extender	extender	critique

Table 2: Organizational roles for non-situation-specific L-TEAM after learning

	situation							
agent	1 1 1	1 1 0	1 0 1	1 0 0	0 1 1	0 1 0	0 0 1	0 0 0
pump agent	extender	initiator	initiator	extender	initiator	initiator	extender	initiator
heatx agent	initiator	extender	extender	extender	extender	extender	extender	extender
motor agent	extender	extender	extender	extender	extender	initiator	extender	extender
vbelt agent	extender	extender	extender	extender	extender	extender	extender	extender

Table 3: Organizational roles learned by situation-specific L-TEAM for Range 1

specific L-TEAM and TEAM in Range 2[3].

These experiments highlight some important aspects of role organization:

- Learning is a promising approach to acquiring organizational knowledge about roles in distributed search. Both situation-specific and non-situation specific learning outperforms hand-coded organizational role assignment in our experiments.
- Situation-specificity is an important part of role organization. Situation-specific L-TEAM produces designs that are lower in cost than those produced by non-situation-specific L-TEAM in both the ranges.
- In addition, the role organization produced by learning is different for the two different ranges implying that different nature of the problems being solved by the system may need different role organization.

Conclusion

L-TEAM highlights the potential for learning as an important tool for acquiring organizational knowledge in heterogeneous multi-agent systems. No single organization is good for all situations and it is almost impossible to anticipate all the situations and provide the agents with capabilities to react to each of them appropriately. L-TEAM is our attempt to study ways to alleviate this enormous knowledge acquisition bottleneck.

References

Lander, S. E., and Lesser, V. R. 1994. Sharing meta-information to guide cooperative search among heterogeneous reusable agents. Computer Science Technical Report 94–48, University of Massachusetts. To appear in IEEE Transactions on Knowledge and Data Engineering, 1996.

NagendraPrasad, M. V.; Lesser, V. R.; and Lander, S. E. 1995a. Learning organizational roles in a heterogeneous multi-agent system. Computer Science Technical Report 95–35, University of Massachusetts.

NagendraPrasad, M. V.; Lesser, V. R.; and Lander, S. E. 1995b. Learning experiments in a heterogeneous multi-agent system. In *Proceedings of the IJCAI-95 Workshop on Adaptation and Learning in Multi-Agent Systems.*

Sutton, R. 1988. Learning to predict by the methods of temporal differences. *Machine Learning* 3:9–44.

Whitehair, R., and Lesser, V. R. 1993. A framework for the analysis of sophisticated control in interpretation systems. Computer Science Technical Report 93–53, University of Massachusetts.

[3] In addition to cost of design, we also did experiments on number of cycles (representing the amount of search. ss-L-TEAM out performed both ns-L-TEAM and TEAM on these measures too.

	situation							
agent	1 1 1	1 1 0	1 0 1	1 0 0	0 1 1	0 1 0	0 0 1	0 0 0
pump agent	initiator	initiator	extender	initiator	initiator	initiator	extender	extender
heatx agent	extender	extender	initiator	extender	extender	initiator	extender	extender
motor agent	extender	extender	extender	extender	extender	initiator	extender	extender
vbelt agent	extender	extender	extender	extender	extender	extender	extender	extender

Table 4: Organizational roles learned by situation-specific L-TEAM for Range 2