Multiagent Systems: An Emerging Subdiscipline of AI

VICTOR R. LESSER

Computer Science Department, University of Massachusetts, Amherst, MA 01003
⟨lesser@cs.umass edu⟩

As more AI applications are being formulated in terms of spatially, functionally, or temporally distributed processing, multiagent systems (or what was previously called distributed AI) are emerging as an important subdiscipline of AI. This is especially true as the outlines of a potential model for computing in the next century are beginning to coalesce: a model in which networks of interacting, real-time, intelligent agents could seamlessly integrate man and machine. Agents in these networks need to be highly adaptive due to their "open" operating environments, where the configuration and capabilities of other agents and network resources would change dynamically. Agents in such environments would aim to produce the best possible result given their available processing, communication, and information resources. As part of this model, we see agents eventually using high-level content languages for rich and succinct communication with other agents. Consequently, problem solving for effectively interacting with other agents would be as, or more, complex than the agent's domain problem solving.

The current set of multiagent applications can be classified into three broad areas. Distributed situation assessment applications, such as distributed network diagnosis, emphasize how (diagnostic) agents with different spheres of awareness and control (network segments) should share their local interpretations to arrive at consistent and comprehensive explanations and responses. Distributed resource planning and allocation applications, such as distributed factory scheduling, emphasize how (scheduling) agents (associated with each workcell) should coordinate their schedules to avoid and resolve conflicts over resources and to maximize system output. Distributed expert systems applications, such as concurrent engineering, emphasize how agents negotiate over collective solutions (designs) given their different expertise and criteria. The next generation of applications alluded to will probably involve all the emphases of these generic applications and more.

In general, multiagent systems are computational systems in which several semi-autonomous agents interact or work together to perform some set of tasks or satisfy some set of goals. These systems may involve computational agents that are homogeneous or heterogeneous and they may involve activity on the part of agents having common goals or goals that are distinct. Research and practice on these systems generally focus on problem solving, communication, and coordination aspects, as distinct from low-level parallelization or synchronization issues that are more the focus of distributed computing.

The design, implementation, and assessment of multiagent systems raise many specific issues. These include coordination strategies that enable groups of agents to solve problems effectively; negotiation mechanisms that serve to bring a collection of agents to an acceptable state; techniques for conflict detection and resolution; protocols by which agents may communicate and reason about interagent communications; and mechanisms whereby agents can maintain autonomy while still contributing to overall system effectiveness.

The need to interact in such systems arises because agents are solving subproblems that are interdependent, either through contention for resources or through relationships among the subproblems' goals. For agents to achieve compatible (nonconflicting) and "optimal" solutions to their interdependent subproblems with minimum use of resources requires them to have sufficiently current, complete, and consistent views of the overall problem(s) and of one another. Obtaining this information is often not practical due to:

(1) Limited communication bandwidth and the computational costs of packaging and assimilating communicated information.

(2) The heterogeneity of agents, which makes it difficult to share information, and the potential for competitive agents who, for their own self-interest, are not willing to share certain information.

(3) The dynamic character of the environment due to changing problems, agents, and resources, and the inability to predict with certainty the outcome of agents' actions.

In order to deal with uncertainty in agents' decisions due to this lack of appropriate information and sufficient computational resources to process available information fully, a number of specific techniques, both formal and heuristic, have been developed. The general principles guiding the development of these techniques are:

(1) The system design goal of producing an "optimal" answer with minimal use of communication and processing resources while at the same time being able to respond gracefully to a dynamically changing environment is often unrealistic for most real-world tasks. Instead, "satisficing" criteria for successful performance are often adopted based on achieving an "acceptable" answer using a "reasonable" amount of processing resources. An associated corollary is that there is no one best coordination strategy over a wide range of environments. Agents must be able to balance the level of certainty of their control decisions against the level of resources required to make the decisions, based on the characteristics of the environment.

(2) The resolution of uncertainty in agents' decisions should be an integral part of agent problem solving. This process of resolution is in general a multistep, incremental process (sometimes thought of as negotiation) involving a dialogue among agents using information at multiple levels of abstraction. Furthermore, resolution of all uncertainty may not be necessary to meet the criteria of "satisficing" performance.

(3) Sophisticated local control is necessary in many cases for effective agent interaction. Agents need to reason explicitly about the intermediate states of their computation (in terms of what actions they expect to take in the near term, what information from other agents would be valuable for further progress in their local problem solving, etc.). Agents also need to be able to acquire, represent, and reason about beliefs concerning the state of other agents and to use assumptions about the rationality of other agents' problem solving in their reasoning.

(4) Organizing the agents in terms of roles and responsibilities can significantly decrease the computational burden of coordinating their activities. However, these assignments should not be so strict that an agent does not have sufficient latitude to respond to unexpected circum-

stances, nor should they be necessarily fixed for the duration of problem solving. Organizational control should be thought of as modulating (circumscribing) local control rather than dictating.

Multiagent systems research is still in its infancy; just this year the first international conference was held on this subject. I expect significant intellectual strides to occur in the near term as the strengths or weaknesses of current ideas are more fully evaluated and as more formal and generic theories, frameworks, and analysis tools are developed. I believe that multiagent systems concepts will be crucial to the successful engineering of large and complex non-AI-based software systems, and that engineering these systems requires sustained dialogue among the multiagent, software engineering, and distributed processing communities.

REFERENCES

BOND, A. AND GASSER, L., Eds. 1988. *Readings in Distributed Artificial Intelligence.* Morgan-Kaufmann, San Mateo, CA.

HUHNS, M. N. Ed 1987. *Distributed Artificial Intelligence.* Pitman Publishing Ltd., London

HUHNS, M. AND GASSER, L., Eds. 1989. *Distributed Artificial Intelligence*, vol. 2, Pitman Publishing Ltd., London, 1989.

IEEE Trans. Syst. Man. Cybern. 1991 *Special Issue on Distributed AI, C-21*, 6 (Nov/Dec).

O'HARE, G. AND JENNINGS, N. Eds. 1995. *Foundations of Distributed Artificial Intelligence.* Wiley Inter-Science, New York.

1995. *Proceedings of the First International Conference on Multiagent Systems* (San Francisco, June), AAAI Press