AAAI Workshop on Cooperation Among Heterogeneous Intelligent Agents

Mark Adler, Edmund Durfee, Michael Huhns, William Punch, and Evangelos Simoudis

■ *Recent attempts to develop larger and more complex knowledge-based systems have revealed the shortcomings and problems of centralized, single-agent architectures and have acted as a springboard for research in distributed AI (DAI). Although initial research efforts in DAI concentrated on issues relating to homogeneous systems (that is, systems using agents of a similar type or with similar knowledge), there is now increasing interest in systems comprised of heterogeneous components. The workshop on cooperation among heterogeneous intelligent agents, held July 15 during the 1991 National Conference on Artificial Intelligence, was organized by Evangelos Simoudis, Mark Adler, Michael Huhns, and Edmund Durfee. It was designed to bring together researchers and practitioners who are studying how to enable a heterogeneous collection of independent intelligent systems to cooperate in solving problems that require their combined abilities.*

Recent attempts to develop larger and more complex knowledge-based systems have revealed the shortcomings and problems of centralized, single-agent architectures and have acted as a springboard for research in distributed AI (DAI). Although initial research efforts in DAI concentrated on issues relating to homogeneous systems (that is, systems using agents of a similar type or with similar knowledge), there is now increasing interest in systems comprised of heterogeneous components. The workshop on cooperation among heterogeneous intelligent agents, held July 15 during the 1991 National Conference on Artificial Intelligence, was organized by Evangelos Simoudis, Mark Adler, Michael Huhns, and Edmund Durfee. It was designed to bring together researchers and practitioners who are studying how to enable a heterogeneous collection of independent intelligent systems to cooperate in solving problems that require their combined abilities.

Among the workshop's principal topics of interest were (1) architectures and frameworks for combining and integrating independent intelligent systems to enable them collectively to solve more complex problems, (2) knowledge-level analyses of the components that belong to such cooperative systems, (3) formalisms and frameworks for representing an agent's individual knowledge as well as the knowledge it uses for integration, (4) identification of the types of cooperation and negotiation that can occur among heterogeneous agents and their implications, (5) descriptions of existing cooperative intelligent systems and experiences in using these systems, and (6) computer environments that facilitate cooperation among human problem solvers of diverse abilities.

...there is... increasing interest in systems comprised of heterogeneous components.

Although papers were not received on every topic of interest, the quality of the submitted work was high. Fifty submissions were received, and 43 contributors were invited to the workshop. The workshop had four sessions that covered the topics of cooperation, negotiation, architectures, and applications. Each session consisted of two papers, each of which was allotted 25 minutes for presentation, and a 25-minute discussion that was led by a member of the organizing committee. This format was adopted because the organizers believed that the single day that was allocated to the workshop was conducive to in-depth presentation of a small number of papers that would lead to issue-oriented discussions. The sections in this article bring forward points that were made during the workshop's four sessions.

Defining Agent Heterogeneity

One of the goals of the workshop was to define what was meant by agent heterogeneity. We summarize the group's collective contributions to this definition by saying that *agent heterogeneity* is exhibited in one or more of the following four topics: (1) problem space, (2) knowledge, (3) implementation philosophy, and (4) agent interaction.

Problem space: A DAI system can be classified as a distributed problem solving or multiagent system. The difference is one of approach. Distributed problem solvers appropriately divide the work required to solve a problem among a set of cooperating agents, that is, a kind of divide-and-conquer approach. Multiagent systems coordinate a number of agents (goals, knowledge, plans, and so on) to solve problems, that is, an emergent system where the whole is greater than the parts. Each approach has something different to say about how the task is decomposed into individual agents, whether driven from the top down as in distributed problem-solving systems or combined from the bottom up as in multiagent systems.

Knowledge: Heterogeneous agents can differ in their content, representation, and use of available knowledge. For example, agents could represent the same knowledge differently to optimize their particular use of it, or agents could obtain knowledge from different sources and introduce information conflicts.

Implementation philosophy: A DAI system can use as agents a collection of existing knowledge-based systems that have been developed under a variety of implementation philosophies. Each agent can use a different internal architecture; this approach has ramifications for the types of agent models that need to be created by the other agents to enable agent cooperation. In particular, representations must be agreed on (either before invocation or as a result of run-time negotiation) that would allow agents to share knowledge, partial conclusions, and other important data. Methods must also be created for agents to assimilate this knowledge into their normal mode of processing.

Agent interaction: Given that the coordination among heterogeneous agents is potentially less restricted than with previous systems, more emphasis must be placed on negotiation and communication among agents, allowing them to work out for themselves the modes of interaction they can achieve.

Cooperation

Cooperation was defined in three ways in the workshop's papers: (1) agents (systems) that have been placed under a framework because they can perform problem solving in a common domain, (2) agents working together to improve their individual performance, and (3) agents working together to improve the collective performance of the system to which they belong.

Generally speaking, one can classify the systems that achieve the first two types of cooperation as multiagent systems, whereas distributed problem-solving systems are more likely to use the third type of cooperation. It would appear that as one moves from the first to the third type of cooperation, the level of heterogeneity in the systems decreases. If an agent determines that it needs to cooperate with one or more of the other agents in the framework, then it might be necessary for the agent to possess and employ a model of the behavior of each agent that it wants to communicate with. This type of modeling can lead to more homogeneous systems. Alternatively, some overseeing agent might be required to translate communications among such agents, preserving their heterogeneous nature but requiring the overseer to be knowledgeable about all the agents in the system, which hinders the ease of agent integration and the portability.

Near-term solutions for achieving cooperation among heterogeneous agents involve retroactively fitting expert systems under a particular framework (this type of fitting can be hard wired; alternatively, one can create a special type of agent that is able to act as a broker to each of the existing agents that need to participate in the DAI system) and requiring each agent to understand global messages and information, typically in a blackboard architecture, so it can cooperate with other agents.

One of the goals… was to define what was meant by agent heterogeneity.

The following strategies for cooperation, ordered from centralized control strategies to more distributed approaches, were proposed in the papers of this topic group:

First, the system designer hard wires the translation of information and knowledge that is exchanged among the agents and directs all the necessary traffic.

Second, a globally accessible data structure provides information to agents, which, in turn, are able to perform the necessary translations and deal with the generated traffic. "Design of Organizations in Distributed Decision Systems" by F. Farhoodi, J. Proffitt, P. Woodman, and A. Tunnicliffe described the system CADDIE that employs this approach in the area of command and control. The paper was concerned with the efficient and practical representation and the use of knowledge about organizations in distributed decision systems.

Third, a globally accessible data structure exists that contains only universally understandable information and knowledge. "Multi-Agent Proposal Generation and Compromise in Cooperative" by S. Lander, V. Lesser, and M. Connell proposed a model, CEPS (cooperating expert problem solving), that follows this approach. The CEPS framework provides mechanisms to enable appropriate interaction, allowing agents to react to each other's proposals and share insights on the viability of the proposals.

Fourth, agents perform all necessary exchanges directly.

Fifth, agents negotiate and converge on decisions by making deals under various types of pressure.

Sixth, agents converge on decisions by making deals using probabilistic methods.

Negotiation

Given the heterogeneity among agents, it is expected that the knowledge that is contained in each agent might be incomplete, and goals of the agents might be conflicting. To resolve their conflicts and exchange pieces of information, such as partial conclusions and updated facts, agents must negotiate their processing. Research on negotiation has revealed the following general attributes and principles:

First, negotiation involves a small set of agents.

Second, negotiation involves, at a minimum, the following protocol actions: propose, evaluate, revise, and accept.

Third, under certain architectures, conflict-resolution knowledge has been encoded in a special negotiator agent. This approach was followed in work by K. Sycara on PERSUADER, M. Klein in conflict resolution, K. Werkman on the distributed fabricator interpreter, and W. Robinson on OZ (in this session).

Fourth, negotiation requires a common language in which the negotiations can be couched.

Fifth, negotiation requires a common framework—an abstraction of the solution—to which the participants contribute.

Sixth, negotiation can require models of other agents and a unified negotiation protocol, as defined in work by J. Rosenschein.

The papers accepted on this topic fell into three categories. The first category included papers that discussed general models of negotiation in DAI environments. Both of the papers presented in this session were from this category. The papers in the second category described means of negotiation to be used during planning. Planning is important in achieving high degrees of coordination among agents. The third category consisted of papers that described models of negotiation that were used in specialized architectures and sytems.

Two papers were presented in this session: "A Decision-Theoretic Perspective of Multiagent Requirements Negotiation" by W. Robinson and "Cooperation of Heterogeneous Agents through the Formation of Shared Mental Models" by K. Sycara and C. M. Lewis. Robinson's paper described work to automate cooperative requirement engineering. This work is based on a model, MPSD (multiple-perspective specification model), that captures individual requirements. Specifications are generated from these requirements and integrated through an automated negotiation process. Resolutions based on analytic compromise, heuristic dissolution, and heuristic compensation are automatically generated.

The work of Sycara and her colleagues focuses on fusing expertise, where each individual agent has deep knowledge in its specialty area but only limited understanding of other domains. Furthermore, each agent might not know what the others need to know. This paper presented characteristics of shared mental models and their use in cooperative problem solving.

Architectures

Even though the predominant architecture for DAI systems is blackboard based, a number of different architectures for achieving cooperation among heterogeneous agents were proposed by the workshop's participants. A number of these architectures were domain and task specific. A central theme in the task-specific architectures was their support of communication among the member agents. "What Architecture for Communications among Computational Agents?" by T. Bouron presented an analysis of different communication approaches in the context of computational agents and described the COMMAS architecture. COMMAS was designed to study issues of communication among multiagent systems. The model is based on the theory of speech acts. It defines communications as actions on goals and beliefs of agents. COMMAS introduces specific knowledge structures for action, commitment, and dialogue. Finally, it includes heuristics that enable an agent to decide the information and knowledge it should communicate to the other member agents.

A central problem for organizations that possess and utilize a variety of heterogeneous expert systems to accomplish a particular task is the integration of these systems into a framework that allows cooperation among the systems. The various constraints that exist in most real-life settings necessitate the development of domain-specific architectures. In "Heterogeneous Knowledge-Based Systems and Situational Awareness," D. Rochowiak and L. Interrante presented one such architecture for the domain of air-traffic control. This architecture, a blackboard-based variant, includes a superagent with situational awareness, which facilitates cooperation. The paper also examined the characteristics of the agents that participate in this architecture from three dimensions: representation, function, and domain. Finally, it investigated how the capabilities of the agents are affected by variations along these three dimensions.

Applications

The last session included papers describing implemented, task-specific frameworks that combine and control a variety of existing knowledge-based systems. "A Control Architecture for Run-Time Method Selection" by A. Goel presented a method for controlling a collection of agents, all of which have knowledge from the same domain, but each of which uses a different problem-solving method, for example, model-based reasoning and case-based reasoning. Control in this type of system presents different challenges than blackboard-based control. The author proposed an architecture for selecting reasoning methods while a problem is being solved. The various problem-solving methods are organized into a memory by the tasks to which they apply.

A major problem when integrating autonomous, heterogeneous knowledge-based systems with the requirement that these systems perform cooperative problem solving is the definition of an appropriate communication language to be used among them. Researchers at Stanford University are working on two such languages, KIF and ONTOLINGUA. B. Buteau in "A Knowledge-Exchange Language for Heterogeneous Systems" presented a third such effort, KXL, for the domain of military threat assessment and warning. KXL was specifically designed for the exchange of objects and relations among the participating agents.

Conclusions

Most theoretical progress in DAI, as reported in the workshop, continues to be in the domains of the homogeneous agent (despite the workshop title), that is, domains where agents share the same structure although they might differ in point of view or in the knowledge they contain. In particular, the majority of the systems and architectures that came from industry looked remarkably similar, with a centralized single-level organization of agents built around a powerful controller agent. It appears that this architecture is the simplest to implement and also best fits the industrial problems currently deemed suitable for DAI.

Unfortunately, the systems described in the submitted papers were mostly in prototype stages with uncertain futures. Conspicuously absent in the papers were validations and evaluations of the systems in real settings, especially as compared with more homogeneous agent systems. This

Most theoretical progress in DAI… continues to be in the domains of the homogeneous agent…

sort of work will be especially important if the future of DAI systems is to remain bright.

Finally, it is clear from the diversity of the papers and the participants that the area of heterogeneous agents and DAI is still looking for clear definitions, typically difficult to find in an area such as AI. More importantly, the motivations of the different groups at the conference revealed two different thrusts. Those interested in building bigger and better knowledge-based systems were busy experimenting with various techniques that might help the system-building and problem-solving processes. Others were clearly more interested in analyzing the advantages of a particular technique (communication, control, negotiation) for pushing the theory side of DAI forward. A merging of these two thrusts or even a merging of different factions within each thrust must come for more rapid progress to be made.

For more information on the workshop, contact Evangelos Simoudis, Lockheed AI Center, 3251 Hanover Street, Palo Alto, CA 94304, <simoudis@titan.rdd.lmsc.lockheed.com>.

Mark R. Adler is a software principal engineer in the Digital Services Research Group located in Digital Equipment Corporation's AI Technology Center in Marlboro, Massachusetts. He received B.S. degrees in mathematics and computer science from the Massachusetts Institute of Technology and a Ph.D. in AI from the University of Edinburgh, Scotland, in 1977. His research at Digital has focused on ways to improve enterprise-wide services, including the use of distributed AI techniques and the use of symbolic models to represent the capabilities of components for advice and diagnosis. He is a member of the American Association for Artificial Intelligence.

Michael N. Huhns is a senior member of the Information Systems Division at MCC Corp., where he has been conducting research in enterprise modeling, schema integration, machine learning, and distributed AI. He received a B.S. in electrical engineering from the University of Michigan in 1969 and M.S. and Ph.D. degrees in electrical engineering from the University of Southern California in 1971 and 1975, respectively. He is the author of 84 technical papers in machine intelligence and is an editor of *Distributed Artificial Intelligence*, volumes 1 and 2.

Bill Punch is currently an assistant professor in the Artificial Intelligence/Knowledge Base Systems Lab at Michigan State University. He received his M.S. in computer science and his B.S. in biochemistry from The Ohio State University. His interest has been the investigation of diagnostic reasoning in a number of domains and the subsequent extraction of general principles of diagnosis. His most recent work investigates the principles of integrated reasoning, that is, reasoning that combines multiple methods and approaches to achieve higher-level problem-solving goals.

Evangelos Simoudis is a research scientist at the Lockheed Artificial Intelligence Center. He received a B.A. in physics from Grinnell College in 1981, a B.S. in electrical engineering from the California Institute of Technology in 1983, an M.S. in computer science from the University of Oregon in 1985, and a Ph.D. in computer science from Brandeis University in 1991. His research focuses on the use of machine learning techniques for diagnosis and design tasks and the use of distributed AI models to achieve cooperation among heterogeneous problem solvers. He is a member of the American Association for Artificial Intelligence.

Edmund H. Durfee is an assistant professor in the Department of Electrical Engineering and Computer Science at the University of Michigan. He received his Ph.D. in computer and information science from the University of Massachusetts at Amherst in 1987 and was a 1991 recipient of a Presidential Young Investigator Award from the National Science Foundation. His interests are in distributed AI, planning, blackboard systems, and real-time problem solving.