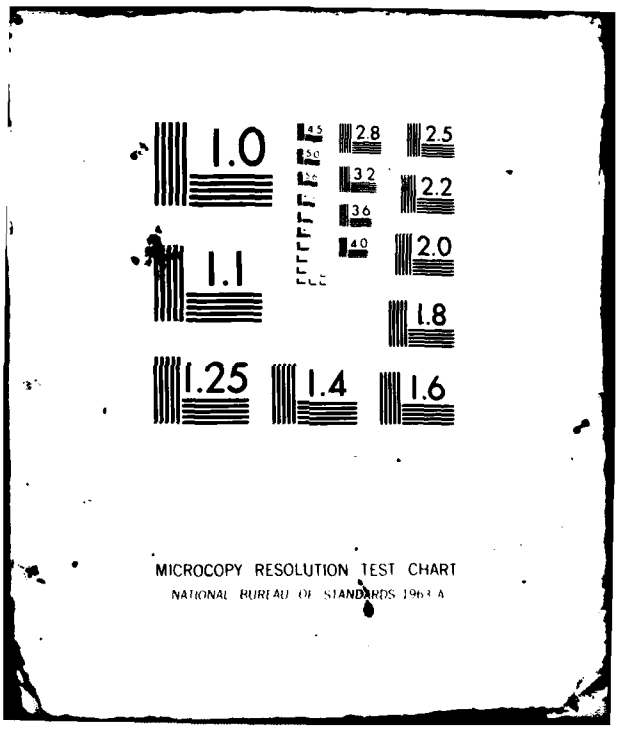
AD-A108 734 SRI INTERNATIONAL MENLO PARK CA ARTIFICIAL INTELLIGE--ETC F/G 6/4
DISTRIBUTED ARTIFICIAL INTELLIGENCE,(U)
MAR 81 N J NILSSON N00014-80-C-0296

UNCLASSIFIED NL

END
DATE FILMED
1 82
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

DISTRIBUTED ARTIFICIAL INTELLIGENCE*

Nils J. Nilsson
Director
Artificial Intelligence Center
SRI International
Menlo Park, California 94025

March 25, 1981

* The research described in this paper is based on work performed under Office of Naval Research Contract No. N00014-80-C-0296, SRI Project 1350.

ABSTRACT

This report examines a number of advanced military information processing problems that entail computational tasks distributed over space. Communicational restrictions and other factors in these applications make it appropriate to consider networks of loosely coupled distributed artificial-intelligence (DAI) systems. Among the important conceptual difficulties of designing such networks is the problem of representing and using information about what one part of the network "believes" about another part. We consider in some detail various aspects of this problem and briefly describe some potential solutions.

I DISTRIBUTED PROBLEMS

Recent applications of artificial-intelligence (AI) techniques have involved tasks that were relatively localized in space. Some prominent examples of such applications may be found in factory automation, photo interpretation, intelligent database access, expert consulting systems, automatic programming, and natural-language processing systems. Yet there are several important problems whose intrinsic spatial distribution dictates a corresponding distribution of the computational resources needed in solving the problem. In this chapter we give examples of some of these "distributed" problems and discuss some current research work in distributed artificial intelligence (DAI).

Our first example of a distributed problem is a "sensor net" whose nodes are dispersed over an area perhaps thousands of miles in extent. Each node might contain, for example, radar, infrared, acoustic, or other sensors and computational resources to process the various signals it receives. The processing done at each node would normally be conditioned by information received (over a communication channel) from other nodes. Networks such as these might be designed to cooperatively detect and track objects such as aircraft, missiles, or marine vessels. It is not hard to imagine versions of such a system in which

communication between nodes would be restricted. We can probably assume that the computational power at each node would be quite high. An important consideration is that the performance of the entire net should degrade only modestly if a small number of nodes became inoperative.

Although we have tacitly assumed that the nodes in the sensor net are immobile, they could in fact move around. One might consider, for example, a pack of small, autonomous submarines tracking an enemy submarine. Members of the pack would have various and changing functions. Some would maintain sensory contact with the target; others would radio positional information to patrolling aircraft and/or orbiting satellites. Still others would be specialists in relocating and communicating the current positions of temporarily lost targets. Destroyed or malfunctioning nodes could be replaced by dropping replacements into the sea in the general location of a target. Tracking of high-priority targets could be refined by adding more nodes as needed.

A second example of a dispersed problem is air traffic control. Ground control stations are distributed over a large area, as are the aircraft being controlled. We can assume powerful computational facilities both on board the aircraft and at the ground stations. This resembles the sensor net problem for detecting and tracking aircraft, except that we can assume that in the air traffic control problem the aircraft are usually cooperative. Another difference is that the control network must plan well-coordinated, appropriate commands to the aircraft to direct their flight patterns.

Another distributed system is a reconfigurable communications network. Nodes of the network might be mobile and contain computers. To send a message from one node to another would typically involve standard routing computations. Such computations would be more complex in a network with mobile nodes (some on satellites, some on aircraft, some on trucks). Line-of-sight and security considerations would limit the permissible routes. Indeed, nodes of the network might be directed to change location to serve certain special communication purposes. In

this case too, communication between nodes might be restricted and, if individual nodes were to fail, it would be important for the network to degrade only gradually.

The requirements of military command and control constitute another example of a highly distributed problem. A carrier task force, for example, might be spread over an area of hundreds of square miles. Computational facilities will exist in each of the component ships and in the aircraft aloft. How should the task force use these computational resources and the distributed information available to it to the best advantage in pursuing its mission? Clearly, conventional hierarchical organizations are not well suited to the distributed nature of the problem. New strategies must be evolved that take this special factor into account.

Analogous situations can also be found in military logistics. Supply, manufacturing, and maintenance and repair centers are typically scattered over wide areas. Further steps to automate logistics functions will require that their distributed nature be acknowledged explicitly.

There are many other examples of distributed problems: cruise missile squadrons, planetary probes, automated offices, and factory robots all involve components distributed over space. The point we consider it essential to stress here is that the effective solution of these problems will require new advances in artificial intelligence research.

II TECHNICAL ISSUES FOR DAI

Whether it is better to consider a distributed computational complex to be composed of a single, integrated system or of several, cooperating subsystems depends on the degree to which the components are coupled to one another. When the coupling is extremely loose, it seems more useful to think of the system as comprised of autonomous but cooperating subsystems.

There are several factors that militate in favor loose coupling among the components of a distributed system. First, security, channel capacity, and terrain all impose communicational restrictions upon the nodes of distributed systems. Second, the problem of designing and maintaining large complex systems is eased by modular designs in which the system is decomposed into relatively independent segments. Furthermore, loosely coupled DAI systems have an inherent reliability/extensibility feature that makes such systems no worse than linearly sensitive to either loss or gain of components. If individual components of a loosely coupled system fail or are destroyed, the remaining ones can be reorganized to carry on without any drastic effects on overall performance. Conversely, if components are added, the entire system does not have to be redesigned to use them effectively. Finally, many distributed systems consist of humans integrated with computer components. For such systems it is important that the latter explicitly acknowledge the sophisticated role played by their human conjuncts.

There are several approaches to designing DAI systems, and we shall have space here to describe only some of the work being done at SRI. Other approaches are briefly summarized in a paper edited by Davis [1]. At SRI we are investigating the technical problems inherent in designing systems composed of several semiautonomous, cooperating AI systems. Each component system, or agent, is assumed to have at best a partial view of the entire problem and can itself contribute only part of the solution. We are not imposing any sort of rigid hierarchical structure on the agents, because we want to investigate the phenomenon Warren McCulloch called "the redundancy of potential command," in which that agent with the most relevant knowledge about a particular problem is the one that contributes most to its solution. In McCulloch's words: "The problem remains the central one in all command and control systems. Of necessity, the system must enjoy a redundancy of potential command in which the possession of the necessary urgent information constitutes authority in that part possessing the information." [2]

Instead of relying on predesigned communication and organization protocols, we expect to be able to achieve communication efficiency and organizational flexibility by requiring each agent to plan its own actions, taking into account the expected actions and knowledge of the other agents. We include communicative actions among those an agent can perform. Thus, an agent explicitly plans to inform and then does inform other agents about facts it believes these other agents need to know. In addition, an agent explicitly plans to ask and then does ask questions of another agent whose answers, already known by the latter, need to be known by the first agent. Communicative actions can also be used to request that another agent perform a certain task or achieve a certain goal. In general, communicative actions affect an agent's cognitive state either by changing its knowledge or by changing its goals.

We shall assume that the agents also have the ability to generate and execute plans consisting of ordinary (noncommunicative) actions that affect the world inhabited by the agents. So, for example, a reconnaisance submarine performs a communicative action when it informs one of its fellows about the presence of a target; it performs an ordinary action when it surfaces. Of course, communicative and ordinary actions will typically be intertwined in complex sequences. When planning action sequences, agents must take into account the possible actions of other agents. One way agents can predict what actions other agents may take is to know the goals of other agents and then to calculate what actions the latter might take to achieve their goals. Conversely, one way agents can know the goals of other agents is to observe their actions and then make hypotheses about the goals toward which those actions are directed.

To gain the flexibility and efficiency inherent in systems of agents that plan their own actions, each agent must have the ability to represent and use certain complex types of knowledge. Each agent must know the actions it can perform and the preconditions and probable effects of these actions. It must know the current "state of the

world." It must have knowledge about the beliefs and goals (the cognitive state) of other agents. It must know about the actions that other agents can take. Techniques for representing and using this sort of knowledge are currently being investigated in artificial intelligence research projects. The major difficulties to be overcome are how to generate and execute plans in dynamic worlds (in which there are other agents of change), and how to represent and use knowledge about the cognitive states of other agents. To illustrate some of the subtleties that surround these problems, in the next section we shall discuss in some detail the representation of knowledge about another agent's beliefs.

III REPRESENTING KNOWLEDGE ABOUT ANOTHER AGENT'S BELIEFS

One of the most important problems in artificial-intelligence research is the problem of how to represent knowledge so a computer system can use it effectively. Intelligent agents of the kind we have been discussing need ways to represent the world around them. One technique for describing the world in a precise way is to use a logical formalism like the first-order predicate calculus to make statements about the world. We do not need to go into much detail here about what the first-order predicate calculus is[3]. Suffice it to say that it is a precise language that can be used in a computer system to represent certain English sentences like "The Nimitz is in the Mediterranean," "There are presently no Japanese ships in the Baltic," and "Every American oiler in the Atlantic is carrying a full cargo." We would typically expect a very large number of such statements to be used by each agent to describe its world.

Although it is rather straightforward to represent a wide variety of sentences in a predicate calculus formalism, some topics present particular difficulties. Among these are propositional attitudes. A propositional attitude is a relation between an agent and a sentence. For example, to say that agent Al believes that the Nimitz is in the

Mediterranean is to state a relationship (or attitude) of belief between agent A1 and the sentence "The Nimitz is in the Mediterranean." This problem is a central one in DAI systems because it is quite important for one agent to be able to represent knowledge about what other agents believe.

Let us examine just a few of the difficulties and discuss some possible solutions that are now receiving special attention in artificial intelligence research. One might think that one could finesse the problem by attaching to an agent a special "database model" of each of the other agents. As an example, suppose agent A0 believes that the Nimitz is in the Mediterranean, that agent A0 believes that agent A1 believes it is in the Baltic, and that agent A0 believes that agent A2 believes it is in the Atlantic. In our predicate calculus language, we might represent A0's beliefs by a database of statements that in turn contains two other databases of statements:

A0's database:
==========

LOCATION(NIMITZ, MED)
other statements of what A0 believes about its world

model of A1's database
==========

LOCATION(NIMITZ, BALTIC)
other statements of what A0 believes A1 believes about its world

model of A2's database
==========

LOCATION(NIMITZ, ATLANTIC)
other statments of what A0 believes A2 believes about its world

==========

This straightforward approach is appealing but, as Moore [4] has pointed out, it suffers from fatal problems. There is just no way to

use this "database approach" so that we can simultaneously distinguish among the statements "A0 believes that A1 believes the Nimitz is not in the Mediterranean," "A0 believes that A1 doesn't believe the Nimitz is in the Mediterranean," and "A0 doesn't know whether or not A1 believes the Nimitz is in the Mediterranean."

Another problem with the database approach is that it is difficult to use it efficiently to distinguish between the two statements "A0 believes that A1 either believes the Nimitz is in the Mediterranean or believes the Nimitz is in the Baltic" and "A0 believes that A1 believes that either the Nimitz is in the Mediterranean or the Nimitz is in the Baltic." (These statements are different and the difference could be crucial! In the first case, A0 believes that A1 itself is sure about the Nimitz's location, even though A0 isn't sure which of these definite statements A1 believes. In the second case, A0 believes that A1 isn't sure about the Nimitz's location.)

Another approach to the problem of representing propositional attitudes is to use a modal logic to reason about relations between an agent and propositions. Thus, we might formally represent the statement "A1 believes that the Nimitz is in the Mediterranean" by the formula

BEL[A1, LOCATION(NIMITZ, MED)] ,

where we take BEL to be a modal operator.

Although modal logics have been thoroughly studied in the literature of philosophical logic, we do not yet have adequate computational techniques for automatic reasoning in modal logic. One of the difficulties is that operators like BEL must have a property called referential opacity. This property simply means that one cannot employ the usual rule of substituting a term for its equal within the scope of a BEL operator. For otherwise, from the two statements "A1 believes the Nimitz is in the Mediterranean" and "the Mediterranean is the Roman Sea," we could deduce that "A1 believes that the Nimitz is in the Roman Sea." We would not normally want to deduce this latter statement from

the first two, because A1 might not know that the Mediterranean and the Roman Sea were one and the same.

Recent research has explored ways in which ordinary first-order predicate calculus can be used to represent statements of belief. Moore [4] represents belief statements in terms of possible worlds. His method is based on representing the possible-world semantics of modal logic within ordinary first-order logic. His way of representing a statement like "A1 believes that the Nimitz is in the Mediterranean" is tantamount to expressing it in a manner something like: "In all the possible worlds that are consistent with what A1 believes, the Nimitz is in the Mediterranean." Moore also applies his technique in exploring the relationship between knowledge and action, a topic that is very important for DAI research. Appelt [5] has developed an automatic system for planning "communicative actions," based in part on Moore's formalism for representing statements about what other agents believe.

Another technique for representing statements about what agents believe is simply to express a relationship between an agent and a string of symbols that encodes the statement believed. Konolige [6] has investigated this technique in the context of DAI applications. In this method, the statement "A1 believes that the Nimitz is in the Mediterranean" would be expressed in a manner something like "A1's list of statements contains the statement 'The Nimitz is in the Mediterranean'." It is important to notice that this approach uses a statement that explicitly refers to another agent's list of statements and to a specific statement asserted to be in that list. It thus differs from the database approach in which no explicit mention is made of statements (as such) and of databases (as such). Current research is exploring efficient ways of using formalisms such as these to represent and use knowledge about what other agents believe.

Problems of representing and reasoning about belief statements are not the only ones in DAI research. We examined these problems here merely to illustrate some important new conceptual research tasks on which progress must be made before flexible DAI systems can be employed.

Before concluding this section on DAI research issues, we might mention also the problem of generating plans in which the component actions can be executed in parallel and in which other agents are simultaneously executing their own plans. New planning formalisms are needed that are sufficiently powerful to express these possibilities. Recent work by Rosenschein [7], employing propositional dynamic logic, is laying a foundation for plan synthesis procedures that are better suited to DAI problems.

IV THE IMPACT OF DAI ON AI

Besides the intrinsic interest in the development of DAI techniques for the kinds of applications mentioned by us at the beginning of this chapter, there are several reasons work on DAI can be expected to contribute to (and may even be a prerequisite for) progress in ordinary artificial intelligence. First, to be sufficiently "intelligent," a system may have to be so complex and contain so much knowledge that it will be able to function efficiently only if it is partitioned into many loosely coupled subsystems. Kornfeld and Hewitt's "scientific community" metaphor [8], Minsky's "society of minds" [9], and (to some degree) "frame-based" systems [10] all proclaim "no AI without DAI."

Work in DAI also helps sharpen our intuitions and techniques for explicit reasoning about knowledge, actions, deduction, and planning. In our opinion, we have yet to devise entirely satisfactory methods for representing beliefs, plans, and actions so that these concepts can be reasoned about. The objective clarity gained by considering how one AI system can reason about another should illuminate our study as to how an AI system can reason about itself.

The methods used by one AI system for reasoning about the actions of other AI systems will also be useful for reasoning about other dynamic (but unintelligent) processes in the environment. (It should be noted that such AI systems might occasionally make the mistake of

attributing planned behavior to purposeless processes and thus might develop animistic theories about its environment.) Previous work in AI planning methods largely dealt with only static environments.

DAI work will contribute to our understanding of the process of natural-language communication. The "communicative acts" performed between intelligent systems serve as an abstract model of some aspects of natural-language generation and understanding. Viewing the process abstractly may clarify certain problems in natural-language communication.

Perhaps most importantly, an AI system that can reason about other AI systems can also reason about its human user so as to maximize its utility to that user.

V CONCLUSIONS

We have described certain computational tasks that are spatially dispersed. Because of communicational restrictions and other factors, we are led to consider networks of loosely coupled distributed artificial intelligence (DAI) systems. Among the important conceptual problems in designing such networks is the complex task of representing and using information about what one part of the network "believes" about another part. We have considered various aspects of this problem in some detail and have briefly outlined some potential solutions.

One ultimate objective of our research in DAI is to design networks of systems that exhibit what McCulloch called a "redundancy of potential command"; that is, we want these networks to have some ability to organize and reorganize themselves according to the amount and quality of knowledge possessed by each node. Those nodes with the most relevant information about the problem at hand should have the most control. It is this flexible feature that will make DAI systems especially useful in the kinds of applications described at the beginning of this chapter.

REFERENCES

1. R. Davis, "Report on the Workshop on Distributed AI," ACM SIGART Newsletter, No. 73, pp. 42-52 (October 1980).

2. W. McCulloch, "What's in the Brain that Ink May Character?" Embodiments of Mind, pp. 387-397, (MIT Press, Cambridge, Massachusetts, 1965).

3. N. J. Nilsson, Principles of Artificial Intelligence, (Tioga Publishing Company, Palo Alto, California, 1980).

4. R. C. Moore, "Reasoning about Knowledge and Action," Technical Note 191, Artificial Intelligence Center, SRI International, Menlo Park, California (October 1980).

5. D. E. Appelt, "A Planner for Reasoning About Knowledge and Action," Proc. AAAI, Stanford University, Stanford, California, pp. 131-133 (August 1980).

6. K. Konolige, "A First-Order Formalization of Knowledge and Action for a Multiagent Planning System, Technical Note 232, Artificial Intelligence Center, SRI International, Menlo Park, California (December 1980).

7. S. J. Rosenschein, "Plan Synthesis: A Logical Perspective," Proc. IJCAI-81, Vancouver, B.C., Canada (1981).

8. W. A. Kornfeld and C. Hewitt, "The Scientific Community Metaphor," unpublished memorandum, Massachusetts Institute of Technology AI Laboratory, Cambridge, Massachusetts (1980).

9. M. Minsky, "K-Lines: A Theory of Memory," Massachusetts Institute of Technololgy AI Laboratory Memo No. 516, Cambridge, Massachusetts (June 1979).

10. M. Minsky, "A Framework for Representing Knowledge," in Psychology of Computer Vision, P. H. Winston (ed.), pp. 211-277 (McGraw-Hill Book Company, New York, New York, 1975).