Intelligent Distributed Systems: New Trends

D. Gaïti

INRIA - Columbia University
Center for Telecommunications Research - New York, NY 10027-6699
Tel.: (212) 854 8928, fax: (212) 316 9068, E-mail: gaiti@ctr.columbia.edu

Abstract

In the Distributed Artificial Intelligence (DAI) field, a multi-agent system was defined as a coordinating intelligent behavior among a collection of autonomous intelligent "agents", how they can coordinate their knowledge, goals, skills and plans jointly to take action or to solve problems. These concepts can be used to control and to manage a distributed system. In this paper, we want to describe the distributed approach of Artificial Intelligence and how to apply this new approach to the management and the control of a distributed system.

1. Introduction

Multi-agent systems are envisaged for the control of distributed systems and network applications that require joint execution, management and allocation of tasks between several agents. Each agent will be processing information concurrently, with new information entering the system at any time. In other words, it seems useful to investigate the Distributed Artificial Intelligence approach when we are in a changing and unpredictable environment where centralized control over the entire network is impossible. This approach can be seen as a new trend to solve distributed system problems linked to the control and to the management of the system.

The paper is organized as follows. In section 2 we describe the distributed architecture coming from different European projects that supports our proposal. Section 3 presents a definition of the DAI concepts and especially the multi-agent concept. The integration in section 4 of these two previous parts provides an intelligent distributed architecture.

2. The distributed architecture

We decided to define a generic integrated management architecture very similar to those described in different European Esprit and Race projects.

This distributed environment is composed of domains. A domain is a set constituted from real resources to be managed in a distributed system and a specific management policy (or a component of this policy). The management process includes the hardware, the software and the human people executing the process.

The second structuration principle concerns the internal structure of the management process (MP). The MP contains two components:

- An abstract representation of the resources and of the activities under control (the MP view).
- A kernel (MP core) that executes the management process. It manages and activates the resources through the abstract representation.

The management is modelled in the following terms:

- The resources whose activity produces a specific function for the treatment of the information;
- A manager that provides a monitoring function, a control function and a review function. Let us note that a manager may be responsible of several resources; each of them being subject to a control and a supervision.

The management is hierarchical and multiple to facilitate a flexible implementation and future extensions. Considering the hierarchical aspect, a manager may itself be managed by a higher level. It turns out that a manager can work with a certain autonomy depending on its own policy. In the same time, a manager may answer to the reception of a control command. The basic model can be extended recursively to different levels of refinement and specialization.

The diagram of Figure 1 shows the level N of the management process. If we consider the manager and the resources of level N as resources of the level N+1, the policy of the level N as the control executed at the level N+1 and the review of the level N as the supervision of the level N+1, we can extend the model at any deep. The relationship between the managed components and the

1071-0485/93 $3.00

managers in the hierarchical management model is shown in Figure 1.

Concerning the "multiple management" aspect, it is necessary to share the resources that are managed simultaneously by several managers. It is also necessary to establish a negotiation between the managers to avoid the conflicts due to a concurrent control of the same resource.

Our management architecture defines the basic principles of the structuration of the real world.

This distinction between the MP view and the MP core allows to separate the management policy (namely, the goal of the management system) from the resources and the managed activities (namely, the managed objects).

The concept of domain allows the description of a large number of management scenarios, from simple managed resources to very complex structures. Indeed, the domains may be structured to reflect a delegate administration in relation with another domain. This delegation may be applied in a recursive way. It means that a domain N may delegate to another domain N-1 the management of some of its real resources. The abstract representation of the N-1 delegate constitutes the MP view of the delegated N. It turns out that this last one can access to a reduce view of the delegated resources through this representation. That decomposition may facilitate the management application.

This architecture allows the domains to be completely autonomous since they are able to manage their own resources. This involves that the description of a managed resource can be provided very easily. Moreover, the possibility to delegate some management tasks from one domain to another one allows the creation of complex schemes. Such an approach allows to model the layered structure with their dependencies.

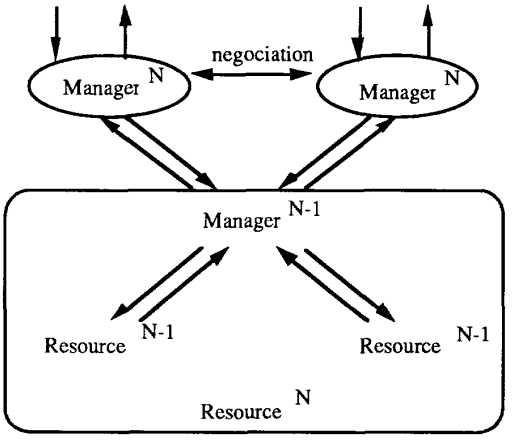

Figure 1: The hierarchical management model

3. Distributed Artificial Intelligence approach

We are interested in the control and the management of the underlying network of the distributed system. For the control, the work is done on a very fast time scale (real time) whereas the management is performed on a low time scale. Real time and distribution problems are two fundamental aspects of network management [1]. Moreover, a network is characterized by its heterogeneity, its layered structure and the distribution of its resources. Thus, we have to formalize such a network by a system in which knowledge is distributed [2].

3.1. DAI: a definition

Distributed Artificial Intelligence (DAI) allows several processes, called agents, to solve a single problem (Figure 2). Each process uses local power and communicates with remote hosts. In the DAI field, the research aim is to propose a scheme of organization allowing the distribution of the resolution of a shared problem. This organization has to describe what are the mechanisms that can be used by agents to cooperate [3].

With DAI, problem solving is a cooperation made by a decentralized group. Agents may be small real processes as they can be complex real ones as applications based upon large knowledge bases. This solving process is

distributed, meaning that each agent has to share a common information set allowing the entire group to reach the solution. This group of agents is decentralized; this means that data and controls are often logically and physically distributed [4].

We are using a DAI scheme to allow the management of heterogeneous networks. The reasons for this choice are the following:

• DAI allows expert systems connections. Several knowledge bases can have a shared part of knowledge. This is easier to use and model.

• DAI is able to solve problems that are too large for one single host.

• DAI is able to solve the problem of working with several experts at the same time.

• DAI seems to be the natural solution. For a distributed problem, it is better to find a distributed solution.

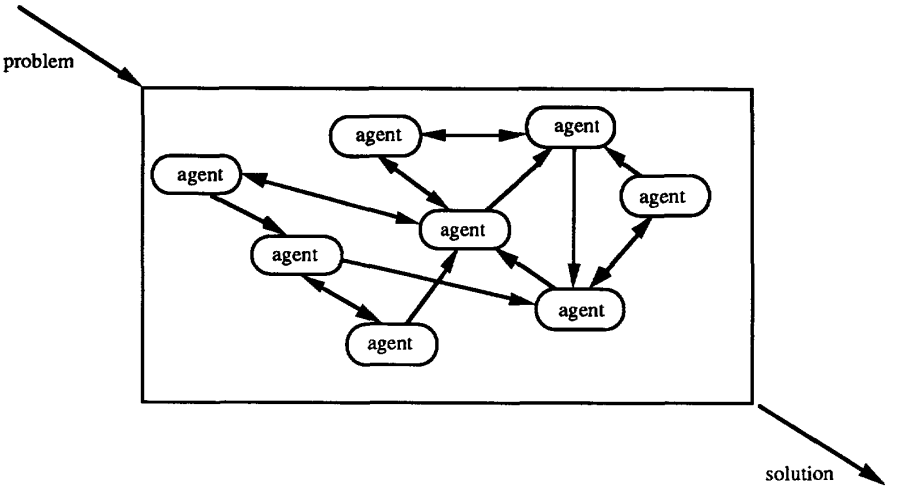

Figure 2: a view of the DAI concepts

In our proposal, we have conceived an intelligent system using Distributed Artificial Intelligence (DAI) techniques, in order to take care of the problem solving inside the kernel of the management process. The system we are proposing is a collection of agents working in parallel and communicating in an organized way.

Indeed, the reasoning process in distributed systems can be represented by an architecture of agents. In this approach, the agents are constituting a network where the knowledge is distributed. This distribution can increase the efficiency of the system. The distribution of the tasks among the different agents (represented by knowledge bases or expert systems) provides a better flexibility against the failures. Moreover, distributed expert systems are able to find solutions more efficiently than centralized system as soon as the problem is complex and large enough.

This technique is well-appropriate to the problem under study due to the large number of potential sources of knowledge and the distributed behavior of the system. The main difficulty comes from this distribution and the cooperation between the agents. This requires appropriate techniques.

DAI can be useful in domains in which action, perception, and/or control are naturally distributed. It is also a way to reproduce the human behavior when a group of experts works together to realize a task. But, the approach we propose is different from all the works done before. We propose a methodology to solve any kind of problems which can appear in the control of networks. Our system is generic even if some specific problems (diagnosis, performance or security management) correspond better to the proposed system. We consider each task of the network management as a distributed problem solving [5].

3.2. Multi-agent systems

Several directions have emerged in the research field related to the study of multi-agent:

• Agents can be considered as autonomous entities that have perception and communication capabilities, as well

as decision abilities. They can act on their environment in an autonomous way. This approach corresponds to the paradigm of "autonomous agents" or of "a robot acting in a multi-robot environment". This organization takes into account real-time constraints and could be applied to control telecommunication systems.

• At the other extreme, agents can be considered as specialized modules interacting together according to a specific architecture. Each agent can be represented by a knowledge base or a specialized procedure. This approach corresponds to the "multi-expert system" paradigm coming from the DAI (Distributed Artificial Intelligence) field.

• Between these two extreme cases where either we are working with a unique system or we deal with independent systems, appear the semi-autonomous multi-agent systems. In this case, the agents operate asynchronously and concurrently on different tasks, with several other agents, with incomplete knowledge about the global environment. This approach is interested in network management where the time is not a strong constraint and where.the agents often need the skill of their partners.

4. A distributed intelligent architecture

4.1. The blackboard architecture

In our project, the DAI architecture follows the principle of the blackboard architecture [6]. It incorporates three components:

• A structured knowledge base called "blackboard" that contains the current state of the solution for a given problem. This shared data area is analogous to the working memory accessed by many production rules in a rule-based system. But this area is divided into separate areas of semantic abstraction called levels.

• A collection of independent agents (or knowledge sources) which may read and write one or several levels. They can be seen as a collection of independent processes able to cooperate to solve a problem [7]. Each agent is specialized in one particular sub-domain.

• A control system that insures the supervision of the actions of the different knowledge sources. This system is also a collection of integrated control-knowledge sources.

This organization is not sufficient to realize a real multi-agent system due to the absence of local memory. We need to develop a real architecture for the agent itself; it cannot be any more just a collection of rules, an entire procedure or a whole rule-based system.

4.2. The agent

The agent itself is organized following the principles of the blackboard architecture. It has three components and each component can be seen as a blackboard:

• A knowledge module: it concerns all the usual data and the knowledge of the domain;

• A control module: it defines the problem solving strategy and the control mechanism to determine next action to perform. It corresponds to the inference engine in the rule-based system;

• A communication module: it gives a model of the other agents in the environment. It also provides an interpreter of messages and a logic for communications with the other agents.

Each agent may be seen as an independent software processor, with its own resources. The agents communicate by sharing the information if we look at a high level, i.e., at the entire multi-agent system level. They communicate through the blackboard that contains, at the beginning, the facts of a given problem. The agents read the information in the blackboard and write in it to reach step by step to a solution.

If we look at a lower level, the communication module in each agent is in charge to receive and to send messages to the other agents to cooperate and to build a solution if they do not have enough knowledge for that.

4.3. The global architecture

First, we have to define the different domains. The domains are composed by a set of managed resources but also encapsulate the components performing management. Therefore, management components and resources may be related to build domains according to different criteria that may be relative to the contained management resources (type, location, functionality, ownership, objectives (management function, applied policy, authority)). For example, if we are interested in the management of N interconnected sub-networks, a domain can be a sub-network. But, a domain may be constituted with the machines of a manufacturer X or may be constituted with the software and hardware of the layer n of the OSI architecture (on all the network or on a sub-network or only on a part of the network).

The domain is managed by one or several agents that can be situated in any node of the network belonging to the domain. These agents work in a coordinated manner on a common data structure: the blackboard. This one is situated in the MP core where all the management functions are provided. The agents may work in group on

a given problem (in the blackboard) or may work alone or with some other agents of their environment to solve punctual problems when they have the appropriate knowledge.

We describe in Figure 3 an example of a possible structuration of several interconnected sub-networks. The bold line shows a domain situated on two sub-networks and including only the machines of the manufacturer X. These machines may also belong to other domains through different views of these resources.

The blackboard of this domain is available in the node processing the MP core. Now, the blackboard and the agents of the different domains may cooperate to insure the global management of the system.

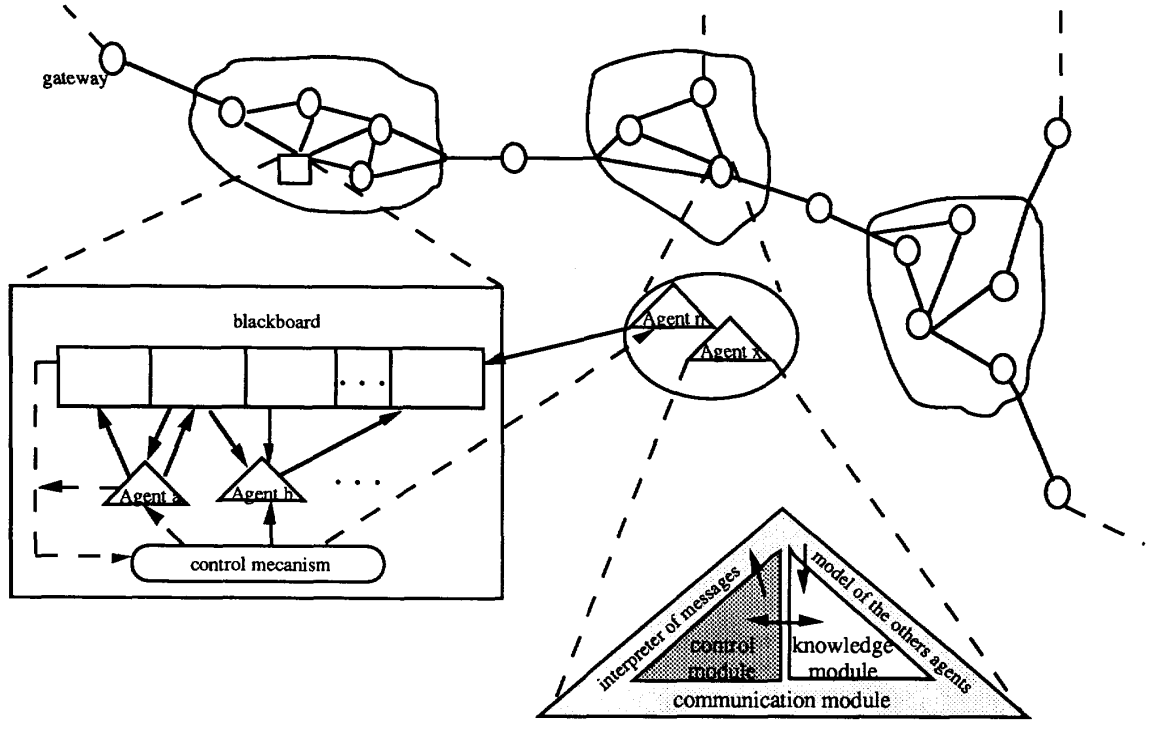

Figure 3: the global architecture

5. Conclusion

The distributed architecture we proposed in this paper may be seen as an intelligent platform that could take in charge network management operations of the underlying network. Many other issues could be investigated. We propose in the followings three directions of research:

- How the different agents of the same domain or from different domains could cooperate efficiently. The notion of a group of agents should be defined when all the information is broadcasted to the other agents of the group. This property could be used in naming, dictionary problems or just distributed schedule-board. In this last area a solving problem can be the organization of a common meeting.
- A problem may rise when agents are not well prepared to handle conflicting situations successfully. A hierarchy of agents can be implemented in this situation or topological relationships may be performed.
- Real time problem is also an important issue when the control is relied on. New types of agents should be developed.

References

[1] S. Rabie, A. Rau-Chaplin and T. Shibahara, "DAD: a Real-time Expert System for Monitoring of Data Packet Networks", IEEE Network Magazine, vol. 2(5), 1988.

[2] D. Gaïti and I. Rahali, "Applying Artificial Intelligence Techniques to the Management of Heterogeneous Networks", In Network Management and Control, pp 189-200, Ed. A. Kershenbaum, M. Malek, M. Wall, Plenum Publishing corporation, 1990.

[3] J. Ferber and L. Gasser, " Intelligence Artificielle Distribuée", Tutorial of the 11th International Conference Expert Systems and their Applications, Avignon, 1991.

[4] L. Gasser, "Distributed Artificial Intelligence", AI Expert, July 89.

[5] P. Lebouc and P.E. Stern, "Distributed Problem Solving in Broadband Telecommunication Network Management", Proceedings of the Specialized Conference on Artificial Intelligence, Telecommunications and Computer Systems, 11th International Conference Expert Systems and their Applications, Avignon, 1991.

[6] R. Engelmore, Robert, et al., "Blackboards systems", Reading, Mass.: Addison-Wesley, 1988.

[7] H. Lâasri and B. Maître, "coopération dans un univers multi-agents basé sur le modèle du blackboard: études et réalisations", Thèse de l'université Nancy 1, 1989.