IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
ITP4317

Basic Organization Structure Model for Cooperative Information Processing

Yao Li, Zhang Weiming, Wang Changying and Gong Yong
Department of Management Science and Engineering, School of Humanities and Management
National University of Defense Technology, China, Tel: 860 (731) 422-5384, Liyao@nudt.edu.cn

ABSTRACT

In this paper, we present a Basic Organization Structure (BOS) model, which can support complicated cooperative strategies in large mutual networks. The model is a cooperative knowledge-representing framework for cooperative problem solving in current LAN architecture and hierarchical organization. BOS is mainly used to support the cooperative problem solving among the coarse-grained, loosely-coupled, and groups of semiautonomous agents. The essential characteristics, knowledge representations, and computational models of the BOS model are illuminated in this paper. And BOS has been implemented and tested on the real-world application of Distributed Traveling Information Management.

INTRODUCTION

Agents and agent-based computing have been an active research area for many years, but it is only until now that agent technologies are beginning to be applied to the development of large-scale and complex commercial, industrial, military, educational and medical treatment information systems. Although the agent-based approach offers a natural and powerful means of conceptualizing, designing and building complex, distributed information systems, how to build actual agent-based applications or multi-agent systems, are still in their infancy.

To solve the information-processing problems cooperatively and efficiently, we present the Basic Organization Structure model (BOS) which can support the complicated cooperative strategies under uncertain conditions. BOS is based on the organization and the architecture framework in multi-agent system founded by means of Assumption-based Cooperative Problem Solving (ACPS)[1], which has been presented by us formerly. By accumulating the evidences and eliminating the contradictions in the cooperation dynamically, ACPS is used to support the continuous cooperative problem solving among multi-agents under uncertain conditions. The research target is to develop a basic organization structure for the distributed cooperative information system so as to build the distributed cooperative information system rapidly and effectively. And the key research is how to organize the local agents and solve problems within the network, so that in the fixed period and under the condition of limited bandwidth, the agent can effectively cooperate and process the incomplete, inaccurate, and complicated data information to get the solution satisfied by the users.

The purpose to establish organizations is to make the members in an organization cooperate effectively to realize the goals. Nowadays, many DAI researchers believe that when designing a multi-agent system, an organization layer should be considered carefully and added to the system structure. This organization layer should include at least the organization knowledge, the problem solving strategies, and the corresponding mechanisms to control and monitor the cooperative procedures, etc[2]. In the early 1990s, we put forward a micro organizational structure (MOS) framework with the agents constrained by the organizations[4], and ACPS was also studied experimentally. Based on the MOS framework, the BOS model introduced in this paper is designed according to the cooperative problem solving within the current LAN architecture and hierarchy organization. So it is a cooperative knowledge representation framework, which is mainly used to support the cooperative problem solving among coarse-grained, loosely-coupled, and semiautonomous intelligent groups of agents. Our current research work is to apply the BOS model and the ACPS methods to the Distributed Traveling Information Management System and to establish a practical cooperative information system.

BASIC ORGANIZATIONAL STRUCTURE (BOS) MODEL

Introduction to BOS

A social organization is generally composed of several smaller basic organizations. So, the cooperations exist both among these basic organizations and within each of them. For example, a university consists of many departments, and a department is divided into several teaching and researching sections or administrative sections. If a section is regarded as a basic organization, the organization is then composed of chief of the section, several staff members, and the public facilities. The basic organizations are used to model the smallest groups of agents in a LAN. By this way, the interrelations among several agents and among groups of agents can be controlled more effectively; thus the system will run effectively and cooperatively as a whole and implement the corresponding global and local goals.

The main advantages are as follows.

(1) The complexities of designing the agents are decreased: In a multi-agent system, one agent usually cooperates with several other agents. And these cooperative actions are uncertain and interwound. Therefore, to get cooperation, each agent must have a large knowledge representation framework so as to illustrate various intentions, skills, knowledge, resources, goals of the other agents concerned, and so on. In the BOS model, the problem solving actions of the intelligent agent are decomposed further in detail, which makes each agent in the BOS decreases its cooperative relations, and makes the knowledge representations shorter and clearer, and the problem solving actions more specified as well. Meanwhile, the knowledge about the skills, goals, intentions, resources of the related agents are represented and managed by the BOS in a unified way. Therefore, the cooperations within a BOS help to make simple agent structures realize complicated cooperative actions.

(2) The complicated cooperative relations are sequenced: By adding a managing layer to a BOS, the decomposition, allocation, supervision and cooperation mechanisms can be unified and carried out with the head agent. So, the cooperation can be done on two layers, i.e. among agents and among BOSs, which simplifies the cooperative relations and strengthens the cooperative efficiency.

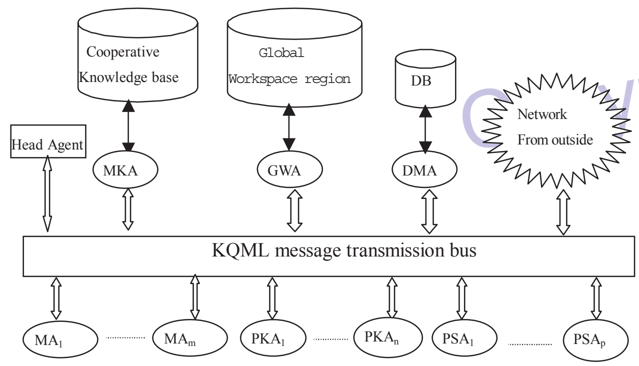
(3) The hierarchy structure combines organically with the parallel structure: The organizational structures in a cooperative problem solving system usually are classified in three types: hierarchy structure, parallel structure and combined structure. If a certain agent in a BOS is a BOS or an abstract representative of the next layer, then by means of global commitment, BOS can model the organizational relations in an information system with any complexities.

(4) Openness: When new agents or new functions are added in a BOS, they can be registered to the Head Agent and are broadcast to other basic organizations by the Head Agent so as to increase new coopera-

This paper appears in *Issues and Trends of Information Technology Management in Contemporary Organizations,* the proceedings of the Information Resources Management Association International Conference.

tive functions. When a certain agent wants to solve problems but faces unfamiliar tasks, the tasks can also be handed over to the corresponding Head Agent in order to seek new cooperative partners. In fact, the Head Agent is playing the role of an intelligent facilitator here. This performance makes a multi-agent system be developed gradually and incrementally.

Figure 1: The basic framework of a BOS



Knowledge Representation in a Basic Organizational Structure

In a BOS, cooperative information processing system and its subsystems are modeled by organization and basic organization respectively. The system organization is an organic community, which is linked by the BOS according to the organizational structures and regulations defined by the organizational model. The BOS has independent computing and cooperating abilities, and they are distributed in a LAN at different locations and interact on each other through high-speed communication nets.

The BOS is a community, which is composed of agents. The agents here are semiautonomous intelligent agents constrained by the organizational roles (i.e. commitments); and in the system implementations, they are defined as programmable computing entity with their data, knowledge and operation sets encapsulated together. The cooperative actions within the BOS are cooperated and completed by various types of agents.

Five types of agents are generally defined within BOS. They are as follows.

(1) Head Agent (HA): HA is the representative of a BOS. It is responsible for cooperating with other BOSs and scheduling the operations of various types of agents within the BOS. HA also takes charge of increasing new agents in a system or adding new functions to an agent and supervising the work state and progress within the BOS. HA is unique. It usually includes many modules, such as planner, task-distributor, scheduler, coordinator, monitor and configurator, in order to realize the functions mentioned above.

(2) Maintenance and management Agent (MA): The main task of MA is to maintain the normal run of the BOS and undertake the routine transaction management. The definition of MA depends on the organizing roles that the BOS undertake. MA usually includes User Aid Agent (UAA) which manages the man-machine interactions, InterAction Agent (IAA) which is in charge of the interactions with other BOSs, and the real time Data Input Agent (DIA) which is responsible for updating information, and so on.

(3) Problem Solving Agent (PSA): PSA is mainly used to realize problem solving and create cooperative tasks by using the cooperative strategies, expertise and domain knowledge. Besides the commitment knowledge such as the organizational structures and roles the BOS has, PSA also has cooperative knowledge such as heuristic model, cooperative strategies that the related acquaintances have and its own knowledge base. The definition of PSA shows the special skills of the BOS.

(4) Procedural Knowledge integrating Agent (PKA): PKA is mainly used to integrate the structured knowledge, which has been tested and proved correct. When running, PKA need not interact with other BOSs or agents, because its problem solving process is fixed. The definition of PKA represents the functional serving abilities of the BOS.

(5) Common Facility Agent (CFA): CFA is used to manage the public data, knowledge and information within the BOS. It is often on the servers and linked with bases. It provides other agents with access operations to the public information. The CFAs of a BOS must include the MetaKnowledge management Agent (MKA) and Global Workspace Agent (GWA), etc. The generally used CFA also includes the Database Management Agent (DMA) and the Modelbase Management Agent (MMA), etc.

For example, in our prototype of Distributed Traveling Information Management System, a BOS has ten agents as follows.

(1) HA: Head Agent (HA).
(2) MA: User Aid Agent (UAA), Visualizing Agent (VA), and distributed Assumption-based Truth Maintenance Agent (ATMA).
(3) PSA: the real time Data Inputting Agent (DIA), User's Information inputting Agent (UIA), Information Fusing Agent (IFA), and Situation Assessment Agent (SAA).
(4) PKA: Situation Identifying Agent (SIA)
(5) CFA: the Meta-Knowledge management Agent (MKA) and Sharing Information Management Agent (SIMA).

THE COMPUTING MODEL OF BOS

In this section, the principal ideas of the BOS computing model are introduced briefly. In the BOS, the communications between agents use the Knowledge Query and Manipulation Language (KQML). Two sorts of important message transmissions lead to the computing continuity of the BOS. They are as follows.

(1) HA sends to various agents in BOS controlling messages, such as task decomposition, allocation message, schedule message, cooperation message and monitor message, and controls the feedback information to activate and coordinate the run of the agents.
(2) The cooperative messages sent by all the agents (including within or among BOS).

Tasks and Events

"Tasks" means that problem-solving orders are input by users or issued by higher-level BOS, and cooperative tasks are from other cooperators. When a system runs, each task is concretized into a problem-solving goal in BOS. Complicated problem-solving goal can be decomposed into smaller ones, and several intelligent agents can handle the undecomposed basic problem-solving goal cooperatively.

In BOS, we define the concept of "BOS event" (called "event" briefly). The computations are solicited by events. An event results in a new computation or makes an unfinished computation keep on.

An event is defined as one of the following cases: (a) sending an outer message (i.e. KQML message from or to other BOS); (b) accepting an outer message; (c) inputting a user command; (d) inputting a real time data information; (e) displaying a result on a console.

For example, in our prototype of Distributed Traveling Information Management System, a BOS has following events.

(a) HA sends a cooperative message to the other BOS;
(b) HA accepts a cooperative message from the other BOS;
(c) UAA inputs a user command;
(d) DIA inputs the data information from the sensors, or UIA inputs the data information from the user;
(e) UAA displays a result to the user.

Task Decomposition and Allocation

User commands are input into BOS by UAA. UAA interprets the commands and takes charge of completing the problem-solving goals (or problem-solving tasks) corresponding to the user command event. UAA has heuristic knowledge about the goals, intentions, and abilities of various agents in BOS. The planner of UAA decomposes and allocates the simple tasks according to the knowledge. The task implementation procedures are scheduled and monitored by its scheduler and monitor. Complicated tasks are handed over to HA.

KQML messages from outside of a BOS are processed by IAA (or HA) in accordance with the cooperative knowledge. The problem-solving tasks committed by other BOS are handed over to HA for solutions.

A real time data input is processed by DIA. According to different types of data, DIA stores the newly input data in the corresponding CFA and generates corresponding problem-solving tasks in accordance with the rude data for processing and then hands them over to HA.

According to the scheduling strategy, HA first sequences all the problem-solving tasks or goals. Then, it works out problem-solving plan for each task. The plan can either match one in the original plan case base or be worked out according to the metaknowledge provided by MKA. After the plan has been worked out, HA stores the structure information about the related task plans in SIMA so that the cooperative problem-solving agents can share this cooperative information. HA sends control messages about task allocation to all cooperative BOSs or problem-solving agents and supervises the implementation of the problem-solving procedure.

After HA receives the feedback information from other agents or BOS, HA synthesizes these results if they are result information. If they are failure information or other cases, HA will modify the original problem-solving plans.

The basic control algorithm of HA is as follows (in approximate Common Lisp language):

```
( loop
( setq L L+1 )   /* to a new control circle.
( setq E_L [ event-area value in HA] )    /* read all events to be
  processed from event area.
( When [ E_L no empty ] do
    [ scheduler generates the processing sequence SPQ of E_L ] /*
     SPQ is event scheduling sequence.
( dolist TR in SPQ   /* TR is an event to be processed in SPQ. TR may
be a new problem-      /*solving goal g, cooperative problem-solving
result r and the factor q resulting in plan modifications.
 ( case TR
    g: [ planner generates the scheduling sequence of the target g;
      task-distri distributes task for g and adds them to TPQ ];
      /* TPQ is the task planning sequence.
     r: [ r is added to the corresponding task result information, the
results are synthesized ];
     q: [ modify TPQ according to q ] ) )  /* case and dolist ended.
 ( dolist H_L (T_i) in TPQ   /* each task in the task planning sequence
 ( case H_L (T_i)
     PSA_L (T_i): [sends KQML messages, makes PSA_L (T_i) finish the
               corresponding subtasks;  the monitor monitors the run-
ning results; ]
     PKA_L (T_i): [sends KQML messages and makes PKA_L (T_i) finish
the corresponding
               subtasks; the monitor monitors the running results; ]
     BOS_L (T_i): [results in outer message sending event, and the moni-
tor monitors the running results; ]
     MA_L (T_i): [ sends cooperative information to other inner agents
]
          /* calls for knowledge, data or function services ) )
         /* case and dolist ended
( if [ the ending conditions are satisfied ] then
    ( GO TO END ) ) )  /* loop is ended
```

Cooperative Computing

After PKA receives KQML messages, it completes the corresponding problem-solving goals and returns to the solving results.

All the tasks accepted by PSA are sequenced according to the scheduling rules and then carried out by PSA one by one. The completion of each task must all cooperate with other agents in accordance with the cooperative knowledge. And the cooperative knowledge come from its own intention base, knowledge base and task information or from BOS's sharing information such as MKA, GWA and DMA. During the cooperations, PSA can communicate with agents in the BOS. It can also communicate directly with agents in other BOS (i.e. the simple cooperative information) or with other BOS by means of IAA (i.e. the complicated heterogeneous information transfer).

IAA is in charge of the metaknowledge communications between BOSs, the complicated information exchanges, and the task-level interactions. According to the knowledge in MKA and the current status in GWA, IAA interprets and processes the received outer messages and then sends the related information or tasks to HA or the corresponding agents.

MKA, GWA and DMA manage the sharing cooperative knowledge in BOS, the global task structure information, and the data information respectively. The common information management agent in the different BOS can cooperate and provide the users with the transparent global information services.

Synthesis and Output of the Results

If the results are synthesized by UAA, they are output to users in accordance with the corresponding forms.

If the results are synthesized by HA, they are returned to the committed agents or other BOS according to the committed forms.

Now we use the Distributed Traveling Information Management System prototype as an example to explain cooperative computing. When DIA inputs the real time data information from the sensors, DIA checks the form of new information, records the new information, assesses the new information according to expertise, and sends the new information to the Information Fusing Agent (IFA). In the process of assessment, if DIA found the new information useful to the other BOS, it sends the new information to HA; if DIA found the new information important, it sends the new information to Situation Assessment Agent (SAA) to analyze the information thoroughly.

When HA accepts the new information from DIA, it inquires of MKA about the cooperative knowledge. According to the knowledge from MKA, HA sends the new information to the related BOS.

When IFA accepts the new information from DIA, it fuses the new information with historical information and environment information. IFA sends the new fusion results to the UAA, SAA and SIMA.

When SAA accepts the new information from DIA and the new fusion results from IFA, it assesses the new situation according to the expertise, and sends the new situation assessment results to the UAA and SIMA.

When UAA accepts the new fusion results from IFA or the new situation assessment results from SAA, it displays the results to the user on a travel map.

When SIMA accepts the new fusion results from IFA or the new situation assessment results from SAA, it records the results to the relevant databases.

CONCLUSION

It is of great significance to study the organizational structure of the multi-agent system for the distributed cooperative information processing, which can greatly quicken the development in many application systems. The examples are distributed sensor network, distributed network diagnosis, distributed information retrieving and collecting, distributed electronic bookstore management, coordinated robotics /no man driving vehicles, distributed perception processing and distributed cooperative situation assessing tasks, etc.

The problem solving in BOS is neither centralized nor all localized but distributed dynamically according to the solving tasks. So this method is suitable for the cooperative problem solving which is real-time, dynamical and distributed. The theory behind BOS was tested and evaluated in a series of experiments in the context of the Distributed Travelling Information Management System. The main result of the experiments is that the distributed cooperative information is processed efficiently and the hierarchical knowledge management is in perfect order, too.

ACKNOWLEDGMENTS

This research was partly supported by a project from NSFC, whose Grant No. was 79800007.

REFERENCES

[1] Yao Li. Assumption-based Distributed Cooperative Problem Solving. Journal of softwareÿ1997ÿ8ÿ12 ÿÿ914-919. (in Chinese)
[2] G.M.P. O'Hare and N.R.Jennings Eds. Foundations of Distributed Artificial Intelligence. John Wiley & Sons , Inc.1996ÿ505-526.
[3] Keith Decker, Victor Lesser, et al. MACRON: Architecture for Multi-agent Cooperative Information Gathering. CS Technical Report, 95-11, University of Massachusetts, 1995.
[4] Yao Li. Distributed Cooperative Knowledge Model and Its Application. A Doctoral Dissertation, National University of Defence Technology, 1995.
[5] Yao Li. Modeling Organization of DAI System. Computer Engineering, 1997, 23(3), 15-19.

Related Content