A Consideration for Cooperative Behavior with Heterogeneous Agents' Communications

Ken'ichi Honji, Seiichi Furuta, Nobuhiro Ito

Department of Electrical and Computer Engineering,
Nagoya Institute of Technology
Gokiso-cho,Showa-ku,Nagoya 466-8555,JAPAN
agent@phaser.elcom.nitech.ac.jp

Abstract. We have designed and implemented rescue agents based on a cooperative agent model by forming groups. By using this model, we could improved ability of rescue agents through many simulations. In this paper, we design and implement a simple cooperative agent model with communications. Then, we consider cooperative behavior of heterogeneous agents on agents' communications on our agent model. And we also think of our challenge of RoboCup2004.

1 Introduction

We have designed and implemented rescue agents based on a cooperative agent model by forming groups[1, 2]. By using this model, we could improved ability of rescue agents through many simulations.

In this paper, we will consider cooperative behavior of heterogeneous agents on agents' communications.

There is a previous rescue agents, YowAI2003, which adopt a cooperative agent model with communication. This agents' team won the second prize on RoboCup2003. So, we adopt such a cooperative model for our agents. In this paper, we design and implement the cooperative model, at first. Second, we consider the effectiveness of the model.

As the results, we confirmed that the model improve ability of agents through some simulations.

However, our agents didn't work very well in RoboCup2004. Therefore, we discuss the problem of RoboCup2004 version's agents.

2 Cooperative Agent Model with Communications

2.1 Previous Works

In previous works, YowAI2003 uses a cooperative model with communications. This model has the following features.

(1) Agents gather surrounding environmental information.

(2) The gathered information is broadcasted to all agents via centers(which is the term defined in subsection 2.2).
(3) The agents which receive the information help other agents which need support.

However, in this approach, there is a serious problem that agents need vast amount of communications, because all information is told to all agents. Moreover, the number of received messages is restricted in RoboCupRescue. So, it makes the problem difficult further. Then, in this paper, we deal with the number of sending / receiving messages and discuss a model as agents can receive required information by using it.

2.2 Agent Model

First, we define agents as follows.

Agents : agents consist of six kinds of agents, which are fire brigades, fire stations, ambulance teams, ambulance centers, police forces, and police offices.

Mobile agents : mobile agents consist of three kinds of agents such as fire brigade, ambulance team, and police force. These are RoboCupRescue agents.

Center agents : center agents consists of three kinds of agents such as fire station, ambulance center, and police office. One kind of mobile agents can communicate with other kinds of mobile agents via center agent. In other words, One kind of mobile agents can't communicate with other kinds of agents, directly.

Here, we define some terms applied to two kinds of agents—the mobile agent and the center agent, as following for ease of explanation.

- **The same type of agents**: The same type of agents consist of one kind of mobile and center agents. A combination of these agents is restricted as follows:
  (1) fire brigade and fire station
  (2) ambulance team and ambulance center
  (3) police force and police office
- **Belonged center agents**: when one mobile agent belongs to one center agent, we call the set of center agents which include such a center agent as belonged center agents.
- **Belonging mobile agents**: when one mobile agent belongs to one center agent, we call the set of mobile agents which include such a mobile agent as belonging mobile agents (to the center agents).

Next, we will define agents' behaviors and communications.

Synchronous behaviors : an agent synchronously behaves according to global clock, which is shared clock among all agents. "one step" consists of (1) receiving messages, (2) deciding actions and (3) sending messages. An agent behaves by repeating the step. We don't explain how to decide action, in this paper.

Direct and indirect communications : Agents have ability of indirect communications (TELL) and direct communications (SAY).

- **TELL**: The "TELL" sends a message to all agents. A mobile agent can communicate with the same type of agents. A center agent can communicate with the other center agents and belonged mobile agents. It shows in Fig. 1. TELL realizes the radio communications in RoboCupRescue.
- **SAY**: The "SAY" sends a message to agents within a certain distance from an agent, which sends message. SAY is used by only mobile agents. By using SAY, a mobile agent can communicate with the all kinds of mobile agents. It shows in Fig. 2. SAY realizes the direct communications in RoboCupRescue.

Agents can distinguish a TELL message from a SAY message when they receive it.

A restriction of communications : Mobile agents can respectively send and receive only four messages for each step.
The number of messages which center agents can send and receive is both the twice number of belonged mobile agents for each step.
Mobile agents and center agents must select important messages when their sending and receiving messages go over the limit. However, these agent must select messages only by using the information of message sender.

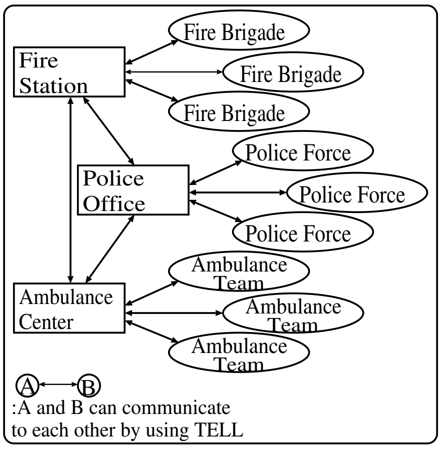

Fig. 1. agent communications with TELL

2.3 A Communications Algorithm

In this section, we discuss a simple communication algorithm based on above our agent model. From the viewpoint of a message flow, a rough algorithm is as following.

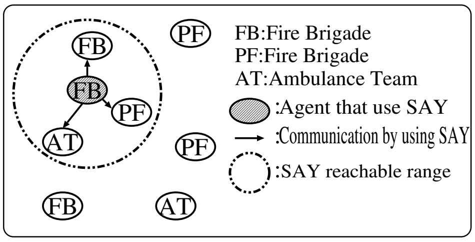

Fig. 2. agent communications with SAY

(1) A mobile agent A sends a TELL message containing their demand to a belonging center agent.
(2) The belonging center agent receives the message sent in (1).
(3) The belonging center agent sends the message received in (2) to the other center agents.
(4) A belonging center agent for a mobile agent B receives the message sent in (3).
(5) The center agent sends the message received in (4) to the mobile agent B.
(6) The mobile agent B receives the message sent in (5) and starts to help the mobile agent A according to the demand which include the message from the mobile agent A.

We use a simple algorithm where one step includes only one message passing. We need to adjust the number of sending / receiving messages on the above restrictions. So we must find out the effective number of messages which an agent sends to the other agents, in order not to lose messages from the other agents.

3 Experiments

In this section, we will find out the effective number of messages through some experiments.

We adopt an approach to adjust the number of sending message. The approach causes all agents to receive messages effectively. There are following four cases in the viewpoint of sending messages.

(1) Mobile agents → the other mobile agents (by SAY)
(2) Mobile agents → the belonging center agents (by TELL)
(3) Center agents → the other center agents (by TELL)
(4) Center agents → all of the belonged mobile agents (by TELL)

So we will discuss parameters for the number of messages related to the above four conditions as following.

(1) Mobile agents → the other mobile agents: 2 .. 3

(2) Mobile agents → the belonging center agents: 1 .. 2

The number of messages which center agents can receive is the twice number of belonged mobile agents for each step.

So,we fix the range of parameters for 2 to 1 .. 2 then we also fix the range of parameter for 1 to 2 .. 3 because of the restriction of messages.

(3) Center agents → the other center agents
(4) Center agents → all of the belonged mobile agents } (3) + (4) : 1 .. 4

We can classify the above 3 and 4 into the following three cases.

(a) Mobile agents → the belonging center agent
→ the other center agents (above (3))
(b) Mobile agents → the belonging center agent
→ all of the belonged mobile agents (above (4))
(c) Center agents → the belonging center agent
→ all of the belonged mobile agents (above (4))

A center agent doesn't sort messages according to destination of messages and broadcast to all connected agents. In other words, a center agent works such as an internet hub.

Based on the above-mentioned and the restriction of messages,we fix the range of parameters for 1 .. 4.

4 Results

4.1 The optimal number of sending messages

We omit a detail result of experiments (simulations), although we experimented on various maps such as Kobe, VC and Foligno.

Here, we show a consideration of our experiments. It is showed as follows.

result of 1 : Each mobile agent sends two messages to the belonging center agents.

result of 2 : Each mobile agent sends one or two messages to the same kind of mobile agents.

result of 3 and 4 : Each center agent sends one message to the different kinds of center agent and one message to the belonged mobile agent (total two messages are sent).

When we use the above parameters related to the number of sending messages, we understood the following related to receiving messages.

- Each mobile agent receives all messages from the belonging center agent and the messages from the same kind of mobile agent as many as possible.
- Each center agent receives all messages from the different kinds of center agents and the messages from the belonged mobile agent as many as possible.

4.2 Comparison with YowAI

We applied the above conditions to our agent and tried to compare with YowAI2003.

We confirmed that our agent can behave more effectively than YowAI2003.

Then, the better effect of the communication support between the different kinds of agents has been confirmed in respect of the following.

- A ambulance team rescued buried fire brigades/police forces more quickly.
- A police force cleared blocked roads, and it affects activity of a ambulance teams/fire brigades more quickly.

5 Consideration in the convention

We challenged to RoboCup2004 by using our model and algorithm. But our rescue agents didn't work very well. So, we considered two reasons for this problem.

reason 1 : Our routines, such as fire fighting, clearing roads and so on, are so complicated that our agents cannot decide action in time.

reason 2 : Our agents couldn't send / receive enough messages to behave exactly.

We thought of the above two reasons as follows.

discussion for reason 1 : We checked whether our agents behaves correctly or not, as changing the number of client machine for running the agents. However, the problem wasn't solved. Therefore, it shows that the problem is independent of the routines' complexity.

discussion 1 for reason 2 : We investigated the agents' behavior in the case of following (1) and (2).

(1) We used only one PC for RoboCupRescue server and our (client) agent program.

(2) We used one PC and some PCs for server and client program, respectively.

From results of (1) and (2), the agents worked correctly only if agents ran on server machine.

discussion 2 for reason 2 : We modified routines to decrease the number of sending messages, and we compared this modified agents with original agents. As a result, the agents worked correctly.

discussion 3 for reason 2 : We modified routines to decrease the number of sending messages, and we compared this modified agents with original agents. As a result, the agents didn't worked correctly.

From the above result, we found that the problem depends on the network throughput and the number of sending messages. Moreover, we found that increasing the number of sending messages makes network throughput low.

6 Conclusion

We have designed and implemented rescue agents based on a cooperative agent model by forming groups till last year. In this paper, we considered cooperative behavior of heterogeneous agents on agents' communications.

At first, we defined our agent model and communication algorithm. When we applied this algorithm to RoboCupRescue agents, we need to fix the number of messages for an agent to communicate with the other agents. Because the number of messages is restricted in RoboCupRescue. Through some experiments, we could find some results related to the number of messages.

We challenged to RoboCup2004 by using this model. But our rescue agents didn't work very well. As the result of the competition, we found that our rescue agent model and algorithm depended on network throughput and the number of sending messages.

In future work, we need to improve the above problems.

References

1. Nobuhiro Ito, Tetsuya Esaki, Naohiro Ishii, "A cooperative agent model by forming a group," 2002 IEEE International Conference on Industrial Technology Proceedings, Vol.II, pp.1260–1265(2002).
2. Nobuhiro Ito, Yoshiki Asai, Nobuhiro Inuzuka, Koichi Wada, "Cooperative behaviors by a group-forming algorithm," Proceedings of First International Workshop on Synthetic Simulation and Robotics to Mitigate Earthquake Disaster (Associated to RoboCup 2003), pp.53–58(2003).

Team Composition

Ken'ichi Honji(Team leader) : 2nd-year master's degree student at Graduate School of Engineering Nagoya Institute of Technology. Development and the design of the whole rescue agent.

Nobuhiro Ito(Team members) : Research associate of Graduate School of Engineering Nagoya Institute of Technology, Japan. Instruction and editorial supervision of the whole development, support, etc.

Seiichi Furuta(Team members) : 2nd-year master's degree student at Graduate School of Engineering Nagoya Institute of Technology. Development of the debugging tool for agent developing environment.