Reducing the communication for multiagent coordination in the RoboCupRescue Simulator

S.B.M. Post, M.L. Fassaert, and A.Visser

Computer Science Department, University of Amsterdam, Kruislaan 403,
1098 SJ Amsterdam, the Netherlands
Tel: ++ 31 20 525 7532 - Fax: ++31 20 525 7490
{sbmpost, mfassaer, arnoud}@science.uva.nl
http://www.science.uva.nl/~arnoud/research/roboresc/

Abstract. In this article we will concentrate on two problems that are important for the design of a multi-agent system, operating within the 'RoboCupRescue' Simulator system. To cope with the limited communication between the center and the agents in the field, we introduce teams with an information manager. The team will build a common knowledge model. The manager will update the center with summaries, enriching the model with information from other teams. Based on the common knowledge, the agents are able to predict the behavior of their teammates, which enables the possibility to cooperate without explicit communication ...

1 Introduction

On a regular basis the world is shocked by a catastrophic event. This may range from natural occurrences, such as earthquakes or floods, to urban riots or terrorist attacks. Crisis management is essential under such circumstances. Unfortunately, real large catastrophes lead not only to thousands of deaths or injured people, but also hit the communication and civil protection infrastructure as well, disabling all carefully prepared plans. It is obvious that something needs to be done in order to minimize such damage in the future. It is possible that the solution may be aided by the use of advanced technologies such as robotic rescue operators and artificially intelligent expert systems.

In this article we will investigate multi-agent coordination within the 'Robo CupRescue Simulator System'. The RCRSS simulates a small piece of a real world environment, in which a disaster takes place. The simulation starts with an earthquake, after which several buildings in the disaster map collapse. This causes buildings to catch fire, roads to be blocked and civilians to get buried under the debris. These events affect the agents on the map and their goal is to respond in the most appropriate way, in order to minimize damage. In section 2 we will determine what kind of problems need to be solved in order to accomplish this goal. In section 3 we will then attempt to tackle these problems. Finally in section 4 we will explain how to prove that our design does indeed improve the performance in the way we expect.

2 Analysis

2.1 Agent Objectives

There are three different agent categories: civilian agents, platoon agents, and center agents. Civilian agents are just ordinary people who can observe, but most of the time they will just run to refuge buildings. The main objective of platoon agents and center agents is to save as many agents as possible and to minimize damage due to fires. The platoon agents and center agents are separated in three operational types: the fire brigade to put out fires, ambulance units to rescue civilians from under the debris and to transport them to refuges, and police units to clear the roads.

The platoon agents perform the actual rescue tasks. This requires a fair amount of localized intelligence. There are many practical problems to overcome like getting to the location of the next task, making sure a fire brigade has enough water to put out the fire and making sure there is room in an ambulance to pick up more wounded. With the very limited communication between the platoons and the centers it is vital that not every detail of a task needs to be communicated to the planning agents. Even some coordination with homogenous and heterogeneous units in the neighborhood should be done autonomously, especially since communication through the center agents clogs up the communication line between them.

We aim to make the localized intelligence in the agents so advanced that the multi-agent system can perform decently with minimal top-level coordination.

2.2 Situation Awareness

The coordinating part relies heavily on information gathered in the field. The platoon agents are not only the hands of the centers but also their eyes and ears. Since the communication line from the centers to the platoons is very limited it will also be necessary for some agents to act as the 'mouth' of the centers. The more the agents can do so autonomously, the more communication can be used for information gathering.

Because of the limited visual range of agents, information gathering will be an active task. A part of the available platoons should be patrolling the disaster space looking for problems that are not known to the control centers. After a risk area is identified, the center will need a risk assessment to base decisions on. This will require the measuring of the extends of a fire, the number of collapsed buildings, the number of buried civilians and their survival chance and finally the number of blocked roads and the finding of a route to the problem area.

On the basis of the constructed world model the centers can decide on a course of action. This decision consists of a distribution of platoons to tasks. The decision making process can take many forms from a simple heuristic function to a game-tree based prediction algorithm or a learning algorithm. Which one of these solutions is best is subject of future research. Our current goal is to translate the information gathered in the field to a world model that is usable by any form of a decision making process and is as good a mapping of the actual situation as possible. The world model is based on summaries of what agents encounter in the field so it will not be a perfectly detailed model, nor should it be. Cutting down on the number of variables in the world will make the job of the decision making process easier. It is also a more realistic simulation of what goes on in a real disaster control center.

3 Solution

3.1 Information teams

The platoon agents will be organized in teams that perform a task together. These groups will have a leader that is in contact with the appropriate center. This agent is called the communication manager. The team can consist of any combination of units. The group composition is a decision of the center based on the advice of the manager. The coordination in the field is done between the members of the group. Seeing platoons act as a team and have to respond fast to a situation, their coordination problems in the field actually resemble those of a RoboCup Soccer team. The difference is that the opponent is not another team but a disaster. Even so certain coordination solutions found for the soccer team, as role taking, are applicable to our problem [1].

3.2 Agent Architecture

Figure 1 contains the structure of an platoon agent for which the Agent Development Kit [3] will be our starting point. This kit hides the details of the RCRSS by providing a number of event handlers. We have worked these handlers further out for certain events. During a cycle multiple communication events can occur. As already mentioned in the introduction, the number of communication events is limited. We can distinguish two kinds of communication handlers that are used during such cycle:

select hear: a hear event-handler decides whether a message is accepted or not. The handler needs group information to determine from which agent to accept messages. The object information in accepted messages is stored in its private object pool.

select send: a send event-handler broadcasts the world information that an agent wants to send to other agents. The handler needs group information to select the appropriate means (say/tell) and priority for relays of new sensory information to other agents.

In case the sending agent is an information manager, it will need an additional buffer where the hear-handler can collect world information that was send by the other agents. When messages of all team members have arrived, the send-handler can produce a summary for its center.

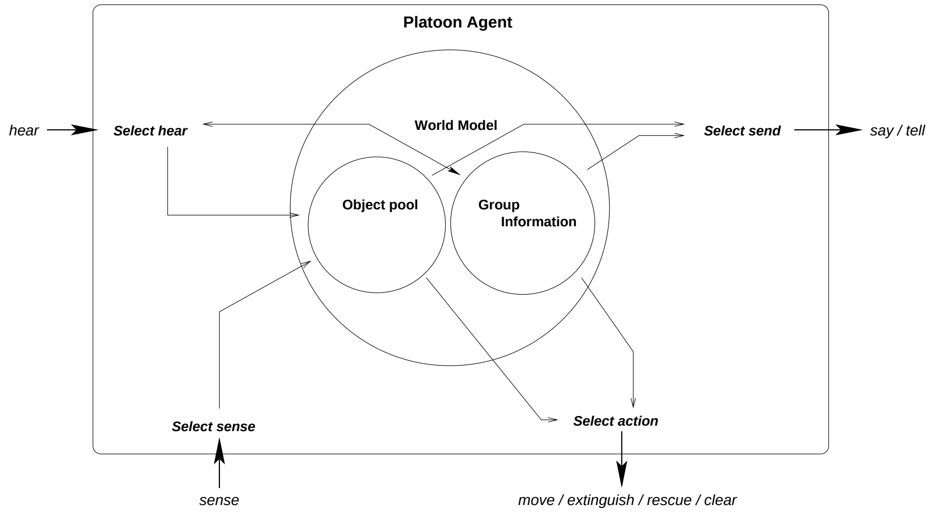

Fig. 1. Agent architecture

The private object pool in the agent's world model is primary maintained by handling its own sense-events. Some care has to be taken to correlate direct observations with observations distributed over the communication channel (**select sense**). Local intelligence is in the selection of the appropriate act- event. We have defined a number of behaviors [2], that define a number of logical sequences of actions over time (for instance: move close to the fire, than extinguish). The selection of the appropriate behavior is largely dependent on the chance on success. The chance on success is for some actions (for instance: extinguish) a function of the number of cooperating agents, which explains why the group information is also used during the **select action** process.

Notice that we foresee that the Group Information is quite static. We will start with predefined teams, and membership of a group can only be changed by an explicit command from a center. This means that for an individual agent the Group Information can only be updated via a hear-event.

3.3 Communication Model

The goal in our communication model is to maximize the worldview that centers have, thereby solving the situation awareness problem. Because sending and receiving messages is restricted, we cannot distribute the worldview easily. The notion of a 'communication tree' with 'communication groups' and 'communication managers', enables us to get what we want. This is illustrated by figure 2.

To be able to tell the difference between meaningful and meaningless messages we introduce communication groups. All members of a communication group will give priority to messages of group members. For those messages to be meaningful, the group members should be in the same region of the map,

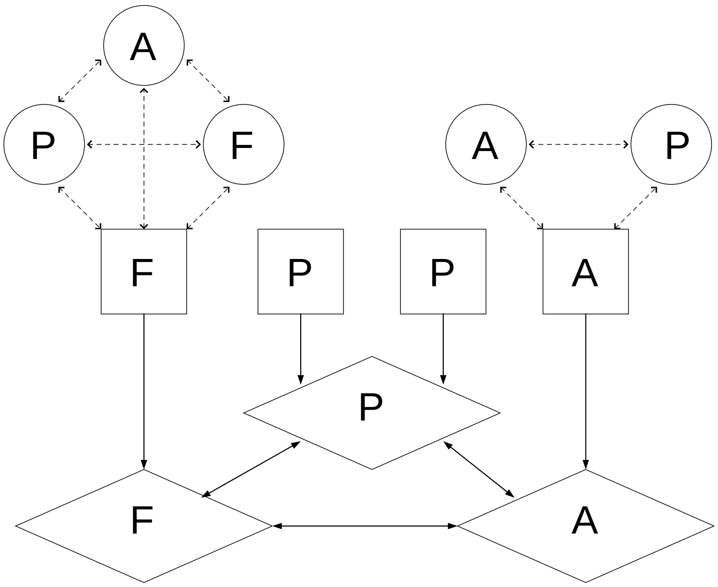

Fig. 2. Communication model: diamond=Center; circle=Agent; square=Communication manager; A=Ambulance; P=Police; F=Fire brigade; dotted line=spoken information; solid line=radiocommunication;

so they can exchange local information, which all members can use to think at a local level. To facilitate local information sharing, we have introduced the concept sectors [2]. In order to maintain information on a global level, we use intermediate information managers. Managers should tell their center of the same type what they see in the world. Because the information that is communicated is a summary it is not needed to communicate with each related agent every turn. This means it is possible for a manager to control a large amount of team members and for a central to control a large amount of teams.

3.4 Common Knowledge

The nice thing about our communication model is that we use communication teams. As a consequence, the agents in the same team tend to know the same things about the world, which means some form of common knowledge is introduced. Because of this they can cooperate without communication. We will make this clear with an example of two ambulance agents. Assume that somewhere on the map there is a badly injured civilian that needs to be rescued as soon as possible. Say, this civilian is discovered and there are several ambulance agents nearby. Because the agents are nearby to a mutual point, chances are that they belong to the same communication team. Agents belonging to the same team will largely have the same information of the local situation; we will call this a problem model. On the basis of these (almost) equivalent problem models they are able to make a reliable prediction of the behavior of the other agents in their team. The coordination is based on common knowledge and social convention. This knowledge can not only be useful for avoiding redundant actions, but also allow the agents to predict whether it is useful to help nearby agents with their task. Because in general it holds that different agents in the same area tend to have the same problem model, this will also work to some extend for heterogeneous agents via the exchange of information between the different centers. A police agent can predict whether or not a nearby fire agent will be traveling a certain road that possibly needs to be deblocked.

4 Research Methods

4.1 Proving the Design Goals

More important than our ranking in the RoboCup Rescue competition is proving that we reached our two design goals presented in this document?Summ. The first goal was creating operational agents that could operate decently without coordination by the centers. It will not be hard to sever the link between the centers and the finished agents to test this claim. The hard part will be to define what decent behavior is. This definition is not directly related to the overall score of the system. We will define uncoordinated behavior as the frequency of failed tasks by the platoon agents. A police unit that arrives at a road only to find that it was cleared by someone else, a fire brigade that can not reach the fire, an ambulance that tries to rescue a civilian agent that has already died, those are all signs of disorganization. We expect that a team that operates on a problem area alone will perform almost the same with or without coordination from the centers, because that is how we designed the system, but serious problems might arise when two teams operate in the same place. It will be interesting to compare our approach (predicting what the teammates are doing based on the shared world model) with an approach where the agents inform their teammates of their current objective. This former approach will suffer from wrong predictions due to an incomplete world model; the latter approach will suffer from a delay of at least one cycle before the objective is distributed over the team.

Our second goal was making sure that the information provided by the proposed communication system is precise and fast enough to base decisions on. To measure this we have to compare the combined world models of all agents in the field with the summarized world model in the centers and count the number of differences at every cycle. It is expected that the error due to communication lag will be large in the first few iterations of the simulation because the difference between the global world model of the center and the local world model of the field-agents will be large. When this difference becomes less the communication lines will become more readily available and the error in the center's world model will asymptotically decrease.

5 Conclusion

We have analyzed problem areas in the design of a multi-agent system for the RoboCupRescue simulator environment. We have extended the formation of teams by assigning the role of information manager to one of the team members, who is responsible for the interface to the center. Inside a team a more detailed common knowledge model is shared, which is used to predict the behavior of the teammates. In this article we have already indicated the measures to performance of this solution. With this research we will cooperate in this year competition on this area, the RoboCup Rescue Simulation League.

Acknowledgements

This paper is based on preliminary research on RoboRescue Simulation done by: S.Post, K.Luikx, J.Uijlings, R.Geertzema and V.Mahabir

References

1. J. Kok, M.Spaan, N.Vlassis. Multi-robot decision making using coordination graphs, Department of Autonomous systems, University of Amsterdam.
2. M.L. Fassaert, S.B.M. Post, A. Visser "The common knowledge model of a team of rescue agents", submitted to the 1th International Workshop on Synthetic Simulation and Robotics to Mitigate Earthquake Disaster, Padova, Italy, 6 July 2003
3. Michael Bowling's Agent development kit `http://www-2.cs.cmu.edu/~mhb/research/rescue/`
4. J.Kok, R.deBoer, and N.Vlassis. Towards an optimal scoring policy for simulated soccer agents, in M.Gini, W.Shen, C.Torras, and H.Yuasa, editors, Proc. 7th Int. Conf. on Intelligent Autonomous Systems, pages 195-198, Marina del Rey, California, March 2002. IOS Press.