Kshitij Robocup Rescue Simulation Team Description

Ramachandra Kota, Prasant Gopal, Yasovardhan Reddy, Kamalakar Karlapalem

Centre for Data Engineering
International Institute of Information Technology
Hyderabad, India.
{ramachandra,prasant_a,yasovardhan}@students.iiit.ac.in, kamal@iiit.ac.in

Abstract. Kshitij Rescue Team participated in RoboCup Rescue competition for the first time in 2005 at Osaka, Japan and obtained the 3rd position. In this paper, we briefly discuss the high level strategies used by our team in the major aspects of rescue like fire extinguishing, civilian rescue, blockade removal and exploration. Some of our approaches are scalable and can be applied in very-large real-life situations.

1 Introduction

Robocup rescue simulation league [1] is modeled as an urban locality hit by an earth quake. The rescue simulation environment is similar to a real-life situation. The environment is dynamic and each agent has a partial observation capability. The agents are required to circulate the information they gather and make online decisions based on the available information. The agents are of three types- police forces, ambulance teams and fire brigades. These heterogeneous groups of agents need to be programmed to search the site, clear the blockades, rescue civilians and extinguish fires. Agents need to cooperate among themselves and co-ordinate their actions in order to increase the number of civilians and buildings that are saved. These methods could be applied for similar situations in real life.

In our work we try to develop good solutions for this problem. Specifically we develop methods that are scalable so that they can be applied in real-life situations. We are building upon our 2005 team [3] by improving it to tackle both the newer developments to the server environment and also developing new strategies and changing the existing strategies for better performance.

Rest of the paper is organized as follows: The exploration of our agents is explained in section 2. In section 3 we discuss the strategies followed by our fire brigade agents. The method of civilian rescue which uses civilian site information is explained in section 4. In section 5 we present our ideas related to fire brigades, path planning and communication model. We conclude our discussion in section 6. References are given in the last section.

2 Exploration

Knowledge of the positions of fires and civilians in the map is essential for agents to stop fires and rescue civilians. Exploration is mainly done by police agents and is supported by fire and ambulance agents. The police agents move through the unexplored areas and communicate the information of fires and civilians to the corresponding agents through the centers.

Each platoon of the same type of agents (i.e. ambulance, fire and police) explores the map independently. For each platoon, the map is divided into districts and each district is assigned to a single agent of the corresponding platoon. At each step the agent selects a target from its district such that the cost of travel is small and number of locations observable at that target location is large.

A priority is given to the unexplored regions according to the known importance of its surroundings. For example, exploring an unknown region with a couple of fires surrounding it is more important rather than exploring an unknown region with out any such threat. Similarly, searching for fires can be prioritized based on the civilian density in the vicinity.

2.1 Aftershocks

Aftershocks are the latest addition to RCRSS. They can be detected through observation of changes in the environment. For example, if an agent finds a blockade in a position which was earlier free of blockades, then it assumes that this was due to the collapse of a building nearby due to some aftershocks. Thus, the nearby region is now considered as unexplored and is included in the list of unexplored regions.

3 Fire Brigade Agents

The task of a fire brigade agent involves locating fires and extinguishing them. Exploration of the center of the map is important since fire in the center can spread in more directions. Similarly, choosing the best fire site to extinguish is of prime importance. Another challenge is positioning the agents around the fiery buildings so that they carry out the task optimally.

All the fiery buildings are grouped into fire sites on the basis of their neighbourhoods. The fire agents choose a fire site to extinguish and then choose a building in that site to extinguish. Selection of the fire site depends on two major factors, number of buildings that can be saved and the number of civilians that can be rescued. Damage due to a fire site in the future is also calculated. An interesting problem that was faced by our team last year was that sometimes, the fire brigades acted on a fire site which was too big to be extinguished and ignored the other smaller fire sites. Eventually the smaller sites would also grow large and cause extensive damage. So, we have tried to improve the agents by building in a mechanism in which the agents determine whether it is possible for a site to be totally extinguished or controlled from atleast one side. If they predict that the site is to big to be controlled, they would act and extinguish the other fire sites before acting on the large site.

The current fire simulator developed by ResQ Freiburg [4] which was first used in RoboCup 2005 is different from the previous one in many ways. First of all, it permits buildings which have been extinguished to reignite. Also, it calculates the fire intensity at a place not only on the basis of the fires of the adjacent buildings but also on the possible heat at the place due to the convection currents from surrounding areas. Thus a building nearer to a fire site has more probability of catching fire than a building farther away even if both of them do not have any fiery neighbours.

The 'Wall Strategy' which was widely used prior to 2005 is no more effective since the 'wall' itself can now catch fire. Also Buildings with highest fieryness which were not considered for extinguishing earlier now had to be taken into account. We have devised and heuristically developed the priorities that need to be given to fire sites with very few fiery buildings as such sites would be much easier and quicker to extinguish immediately. Also, the priority assigning mechanism takes into account the probable damage that can be caused by a particular building to both its clean neighbours and also the extinguished neighbours. Another factor that is to be considered is the ease by which the building may reignite once it is extinguished. We developed the priority assigning mechanism based on all the above parameters.

A different kind of challenge which is faced by the fire brigade agents is positioning themselves around the fiery building that they plan to extinguish. This problem didn't exist prior to 2005 because then more than one fire brigade could enter a building. Our agents try to pre-decide their destination building through communication even before reaching the region around the target building. This helps in cutting short the time wasted in the fire brigades arranging themselves around the target building.

4 Ambulance Agents

The rescue site has a number of civilians who are buried in the collapsed buildings. Civilians are the highest priority in the rescue environment. An ambulance agent is required to dig out a civilian from the debris. To maximize the number of civilians rescued, the ambulance agents should be utilized efficiently.

Our agent team selects a civilian site which has a large number of civilians with low health and low travel cost and will rescue all of them one after another. We use a max-cut algorithm to group the civilians to form civilian sites.

Selection of a specific site is based on estimation of the number of civilians that can be rescued in the future. For example, a group of civilians right next to a fire should be rescued before another group far away from any fire site. The cost involved in saving a group of civilians includes the travel cost of ambulance team to the civilian site.

We have also developed strategies for large scale maps as in real life situations. The whole region is to be divided into districts and each assigned to a set of ambulance agents. In case of large platoon size, we group the civilians who are near to each other by applying the max-cut algorithm. Now we find the best group to rescue and assign a sub-set of the total ambulance agents such that the average travel cost for that sub-set is lesser than the average rescue effort. In the next step we select the next best group of civilians and assign the ambulance agents similarly. We repeat this process until all the agents are alloted.

5 Future Work

In this section we present some specific areas in which we aim to improve our agents. We shall discuss possible improvements in the future especially regarding the strategy for water refilling of fire brigades and communication methodology.

5.1 Water Refilling

Generally, when a group of fire agents act on a single fire site, they start extinguishing the site at the same time which results in most of them exhausting their water tank also at the same time. When all of them return to the refuge to refill their tanks, the site is left uncontrolled and may spread its fieryness rapidly. Therefore, we plan to devise a mechanism by which the fire brigade agents control their water flow so that the water tanks of only a few of them are exhausted at one time. In this way, the fire site is always acted upon by some fire brigades and not left free and unmanned. The mechanism would involve the fire brigades at one site communicating their tank quantity to each other and some of them limiting their water flow so that their supply would last until the others refill their tanks. If the water content of a fire brigade is almost nil and the agent has a choice of two paths, it should choose a longer path with water facility (refuge) instead of a smaller path without one.

5.2 Communication

Communication is the most important part of a multi-agent team. It is required to pass the information about the world and to co-ordinate actions among the agents. In the rescue environment the communication bandwidth is limited and should be used effectively. A platoon agent can send and receive up to 4 messages of 256 bytes of information in each cycle, while the center agents have a bandwidth equal to the number of agents under it. A possible improvement to the communication model is piggybacking the previous messages that were sent, i.e. we attach the most important parts of the previous messages to the current message, so that the reliability of the communication channel is enhanced. We are also exploring the possibility of utilising the 'Say' method of communication when the fire brigades need to decide their positions around the target building.

6 Conclusion

In this paper we presented our robocup rescue team Kshitij. Our major work has been in the area of Exploration, Fire extinguishing and Civilian rescue. We have discussed the possible ways of handling aftershocks. We have developed heuristics which help our fire brigade agents to decide on the target fire site and building. We have also developed a strategy for deciding upon the position of the fire brigade agents around the target. Ambulance agents group the civilians into sites using a max-cut algorithm. Rescue of all the civilians at a single site will decrease the travel cost of the agents and also helps in faster decision making by decreasing the search space. These methods have helped in improving the performance of our team Kshitij.

References

1. *RoboCup Rescue League*. http://kaspar.informatik.uni-freiburg.de/%7Ercr2005/, 2004.
2. *How to Develop a RoboCupRescue Agent*, http://ne.cs.uec.ac.jp/ morimoto/rescue/manual 2002.
3. Ramachandra Kota et. al. Kshitij: Team description. In *Robocup Rescue Simulation League*, 2005.
4. Timo Nssle et. al. *Approaching Urban Disaster Reality: The ResQ Firesimulator*. http://kaspar.informatik.uni-freiburg.de/ nuessle/, 2004.
5. M. X. Goemans et. al. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. In *J. Assoc. Comput. Mach.*, 1995.