RoboCupRescue 2016 - Rescue Simulation League Team Description Ri-one (Japan)

Akira Kinose, Futoshi Sato
Tatsuki Fukumori, Yukino Watanabe
Ryusei Nakatsu, Masataka Yamaguchi
Yuki Kobayashi, Hitoshi Nakamura, Kouki Hayashi

Ritsumeikan University, Japan
is0256hr@ed.ritsumei.ac.jp
http://rione.org/hp/

Abstract. This paper describes features of Ri-one, our team, for the RoboCup Rescue Simulation (hereinafter referred to as RCRS) in 2016. Last year, our team proposed several methods of conveying information through dynamic clustering, suited to each situation. Additionally, we worked on several individual implementations for the following: modifying the manner in which an ambulance team agent approaches a civilian, setting extinguishing positions for fire brigade agents, and producing an even distribution of police force agents around the map. In contrast, our team this year we implemented Search algorithm to use ADACHIalgorithm, Cooperating with some agents to use communication, suitable algorithm about Ambulance Team, Sharing building information to use communication, Effective methods for police force agents by completing another agents's purpose. We will explain each of these approach methods and the ways in which they are applied to the agents.

1 Introduction

The RoboCup Rescue Simulation is a multi-agent simulation for disaster relief. The RCRS server simulates various circumstances imitating a city after a disaster. The aim is to make use of the virtual agents in order to rescue buried victims from under blockades, and to extinguish fires causing buildings to go up in flames. Last year, our team implemented methods clustering of active information to use plural clustering, changing a solution of Ambulance Team to civilian, searching optimum areas for fire brigade agents to extinguish fires, and enhancing the efficiency of obstacle clearances by distributing police force agents evenly around the map. This year, we implemented a new search algorithm, new communication systems, and police force agents to satisfy with other agents request. Each of the chapters will describe the following contents:

- Chapter 2: Search algorithm to use*ADACHI*algorithm.
- Chapter 3: Cooperating with some agents to use communication.
- Chapter 4:
  - 4.1. Implementing suitable algorithm about Ambulance Team
  - 4.2. Sharing building information to use communication
  - 4.3. Effective methods for police force agents by completing another agent's purpose

2 Search

We have used the A* algorithm for agent's single moving. This algorithm we have used for a long time was excellent, however it has an problem. it is that when the agent goes to same destination where he has been to before, it was necessary to use the same algorithm repeatedly. To solve this problem, We use "*ADACHI*algorithm" as agent's moving algorithm instead of A* algorithm. "*ADACHI*algorithm" is one of the algorithm for path planning. It was developed for the Micromouse Contest, competition of autonomous moving robot that search in the maze.

Examples are shown below.(Fig.1)

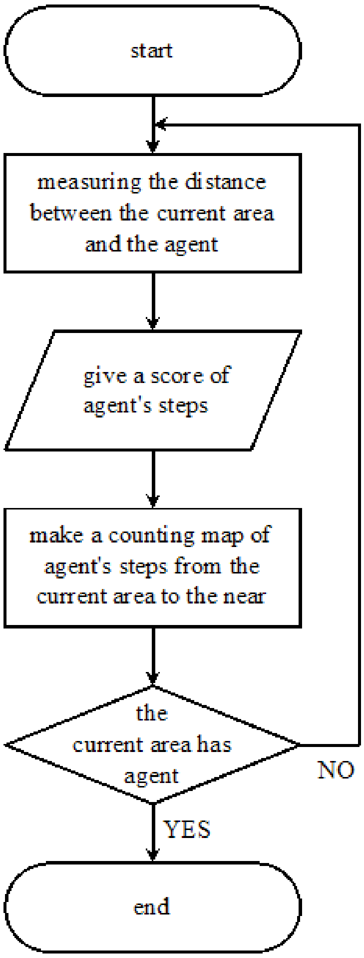

Fig. 1. frow

As a precondition to use this algorithm, following things are defined.

- At the initial state, agent knows only about the area.
- Agent is given the data of the goal area and the start area.
- Agent can measure the edge which is connected from existing area.
- Agent can move the graph along the edge.

The algorithm's processing flow are given below.

1. Make ideal graph (G') which hasn't any wall at the initial state.
2. Calculate the shortest distance to the goal area by using G'.
3. In the actual graph(G), allocate the cost to the adjacent area from area that agent exists.
4. Comparing G and route which is calculated at step2., and delete routes to the not existing area.
5. If the route was decided from step.4, the agent move along it. If there is nothing route age nt can move, return to step.2
6. Judge whether at step.5 area equal the goal area. if it's true, the agent finished searching. If it's false, the algorithm returns to step.3.

By using this algorithm, we can get the "Pedometer Map"(Fig.2) which shows how far the goal area. Therefore agents can get the shortest distance by moving along this "Pedometer Map".

Fig. 2. Caption:example of Pedometer Map on Rescue

However, these flows are ways of thinking about searching for a maze of the N*N grillwork. At the rescue, there are differences of the size among areas. In addition, some road can't through because of blockades. Therefore, in this implementation, we made pseudo pedometer map by using value of evaluation function which has been used before. Additionally, agent's moving algorithm we used before searched the routes by considering the agent's current position as start area and the target building as goal area. In this implementation, however we requires pedometer map. The agent's position always fluctuates, but buildings don't. Therefore we reversed the purpose area and start area before searching of the purpose area.

In this year, we use this algorithm only in the case of one-to-one. It is because that if we calculate the whole map, it'll happen combinatorial explosion. this algorithm have to memorize pedometer map, but it is difficult to make it of all buildings for lack of memory. That is why we use pedometer map for important area, are where agents go many times (for example refuge) . We use this algorithm only at these two conditions. We can show the decrease of calculate time, but we can't show the difference of score about before searched algorithm. we use value of evaluation function used last year, so it didn't appear the difference of route at optimum searching route. Therefore we have two subjects for after this. First, developing the value of evaluation function to operate this algorithm effectively. Second, at one purpose area-to-agents, it moves the agent which was found at first on pedometer map to the purpose area.

3 Communication

Communication has an important role in helping an agent to take efficient actions. The world model of each agent is updated by communication and the information it is obtaining from its view. In each world model, all information which is not obtained from the agents view is provided by communication. Consequently, this means that most information in an agent world model relies upon communication. An agent selects its actions based on the information from its own world model only, making them completely dependent on it. Given that there is a limit to the actions an agent can select from only the updated information that it sees, the additional information updated by communication becomes crucial. Thus, by improving the content of this communication, an agent can take more effective action.

In purpose of the former method, each of the agents unified much information and judged it. In this way, the agent is able to make a quick and correct decision. But, this method has an insufficient point. It is cooperation between agents using the communication. Conventionally, we implemented only the efficient action in the unit of agent by using communication. In this method, however, in order to satisfy agents‘ demands, we make another agents cover those demands. Specifically, we implemented sharing the buliding where a fire brigade agent can't extinguish. In addition, if fire brigade agents can't through the way when they go to hydrant, police forces clear it by using communication. We comment on these methods in Chapter 4.

4 Agent Skills

4.1 Ambulance Team

Optimize the matching algorithm The main role for the Ambulance Team agent(AT) is carrying buried victims under blockades to refuge and keeping them alive as much as possible. We have employed the Maximum Weight Perfect Matching Algorithm(so-called the Hungarian Algorithm) which the ambulance team agent has followed so far. This algorithm computes an optimum solution which civilian AT have to save when there are a number of victims. However, in this year, we decided to add the number of civilians in buildings to Hungarian-evaluation in order to improve it . In next paragraph, we are going to propose and discuss our post-Hungarian Algorithm.

Following is main points of the conventional Hungarian Algorithm:

- Following equation examines a degree of priority of whom to help first:

$$Degreeofpriority = \frac{1}{remainderoflifecycle} * \frac{1}{distancebetweenagentandvictim}$$

- The target duplication will occur among an ambulance team agent who is assigned a command and who is not because latter an ambulance team agent will not know the number of rescuer per survivors in need of help is already enough for them.
- Dispersing an ambulance team agent who is not given an order is needed not to cause target duplications.
- Ignoring the number of people who needs help brings about unbalanced assigning number of rescuer to them.
- Computing an exact solution is $O(n^3)$ and owing to this the process does not terminate.

Instead of the conventional Hungarian Algorithm‘s equation, we devise a new evaluation function as the post-Hungarian Algorithm. We introduce following items into this:

- The number of people in the building who are buried under blockades

– Distance between an ambulance team agent itself and survivor in building who are buried under blockades

This new function has an ambulance team agent go to a building where they can rescue more people if there are similar buildings where civilians who need help at the same degree remain. Assigning the required number of ambulance team agents to each buildings with this function also cancels the target duplication. Our post-Hungarian function improves the precision of priority evaluation. On the other hand, the order of computing still does not change. In the next paragraph, we will perform comparing our post-Hungarian function with another algorithm whose order is lower than it.

We introduce the Traveling Salesman Problem(TSP) as an algorithm which has low order than our post-Hungarian function. Although there are various solutions in TSP, we picked the Greedy Algorithm for comparison because we can take buildings with many people and a distance between an ambulance team agent and them into consideration. It is the one of the algorithm that an ambulance team agent visits the nearest buildings in a sequence from departure point. Faster computing speed and simple algorithm are significant features of the Greedy Algorithm. This algorhithm requires only two data: the distance between an ambulance team agent and each buildings, and scanning line that traces spots where the number of people there is highest. These information helps AT to prefer spots with large number of people and shortest path. By using plural maps in the same server that we used in 2014, we counted scores of rescued civilians in case of the Greedy Algorithm and our post-Hungarian method. From this experiment, we obtained better results from the Greedy Algorithm more than our post-Hungarian method as shown in Table 1.

Table 1. Comparison result

	Hangarian Algorithm	Greedy Algorithm
Eindhoven	217.57	221.59
Paris1	210.28	192.15
VC1	222.84	204.1
Istanbul1	107.75	101.33
Kobe1	99.03	69.35
Mexico1	158.69	137.1

4.2 Fire Brigade

Sharing information which burning buildings has not been extinguished completely by using communication The purpose of the fire brigade is to extinguish fire caused by disaster. It is important to prevent the fire from spreading and to limit the damage by which it causes. In addition, it is also important at Rescue Simulation to find and extinguish the fire as fast as possible. Our team had encountered some problems to resolve in previous years. One of them was that when a fire brigade agent runs out of water, it leaves the spreading fire in order to go to the water supply. This means that the fire carries on spreading rapidly while agents are gone, overall causing more damage than good. Last year, we devised and proposed the method how to decide whether a fire brigade agent can extinguish the fire completely. This year, we made the algorithm. Using communications about conditions of fire, fire brigade agents share the information about the building which hasn't been extinguished completely. In addition, we evaluated the fire brigade agent's behavior to go to the buildings which was decided from the information about the building which hasn't been extinguished completely. This method is sharing the target building's information with other fire brigade agents when the fire brigade agent find the fire and judge that it can't extinguish the fire by itself. The fire brigade agent who got the information go to the building which hasn't been extinguished completely when it judged that the building has high priority by comparing others after taking into account its work and the condition of its environment.

In this year, we implemented the communication of buildings to extinguish. The condition of communicating buildings is that This communication is performed when the scale of the fire is large and there are not enough numbers of fire brigade agent who extinguish the fire. The fire brigade agent which are not extinguishing the fire go to extinguishing the fire to the building which received by communication as priority where they received by communication according to the priority with it. As a result, this change enables us to utilize this method for the fire which fire brigade agent left because it went for water supply in the middle of fire extinguishing. Also it allows us to apply this for the building which fire brigade agent was not able to extinguish a fire completely. Therefore, we became able to often prevent the spread of a fire more. However, an agent might not send information well because of noises and error of the evaluation function. Taking these factors into account will be the next step to enhancing the efficiency in the future.

Sharing information which position of extinguish by using communication There is an problem that when fire brigade agents supply the water, they waste the time because it can't go to it's destination because of blockades on the road to go. This motivated us to suggest a new efficient method of extinguish fire by communicating the blockades fire brigade agent want to extinguish to the police force agent. In our team, we had faced the problem that if the way to refuge is obstructed by blockades when fire brigade agent goes to the water supply, it stops in front of the blockade, and it was taking place frequently. This problem has a significant influence. It leads to the cycle of the fire extinguishing being delayed that a cycle of the water supply is delayed. Thus, the fire spreads more and leads to the drop of the score. In order to solve this problem, we suggest the efficient method to clear blockade on the road from the fire extinguishing position to refuge by using communication. Summarized below are the anticipated benefits which we could receive by using this method: a way to refuge becomes able to go, the movement of a citizen and the agent becomes easy, water supply activity becomes smooth. Therefore, this method will lead to the improvement of the score.

4.3 Police Force

The main role of a police force agent is to clear blockades which are caused by disaster. The action of each police force agent influences all agents and civilians in the virtual city. All possible routes for every agent to take action, are determined by the range of passable roads. In order to efficiently perform the action which ambulance team and fire brigade agents execute, police force agents must clear the blockades efficiently. The police force agent of conventional Rione pursued that police force clear blockades on the whole map efficiently and introduced dispersion of the police force using the clustering in Team Description Paper of the last year. For this we use the K-means++ algorithm once again by dividing the map into the number of police force agents, so that each agent has one area to patrol. This ensures that we succeed to enhance the efficiency of clearing. On the other hand, agents were forced to take a useless step such as go in a roundabout way and don't get a motion since many roads which agent uses in relief activities are blocked up with blockades for a while after simulation began. To devise a countermeasure, we implemented the police force agent which receive the condition of other agents by communication and clear the blockade for other agents in this year. We suggest following two specific examples. The first suggestion is as follows. The content of this suggestion is given in the foregoing chapter. There is the problem that when fire brigade agents supply the water, they waste the time because it can't go to it's destination because of blockades is on the road to go. For this problem we suggest the communication of the extinguishing position of fire brigade agent. By using this communication, police force agent can clear the blockade on shortest courses from a fire to refuge during fire extinguishing of the fire brigade agent beforehand. Consequently, we can expect fire brigades to perform smooth fire extinguishing. The second one is to let saved civilians alone on the road temporarily so as to rescue other critically injured civilians. We had already proposed it in the chapter of the ambulance team agents of the Team Description Paper last year. Whereas, it is possible for ambulance team agents to leave rescued civilians in the lurch unintentionally if a route to refuge is too long even though it let them alone on the road where other ambulance team agent's frequently pass. To deal with this, we use the Ant Colony Optimazation that we proposed last year to clear shortest path from road to refuge whose priority is high. This allows them to pass the routes to the buildings with high priority earlier and to carry civilians easily. Therefore, it is well expected that it leads improving our score.

5 Result

Table 2. Comparison of Building damage on Preliminaries Day 1 and Day 2

	VC1	Eindhoven1	Berlin1	Paris1	Kobe2
Ri-one2014	0.280	0.122	0.942	0.976	0.923
Ri-one2015	0.347	0.127	0.892	0.972	0.900

Table 3. Comparison of Building damage on semifinals

	Berlin2	Paris2	Eindhoven2	Istanbul2	Kobe3
Ri-one2014	0.850	0.950	0.210	0.850	0.670
Ri-one2015	0.862	0.945	0.212	0.862	0.690

Table 4. Comparison of Building damage on finals

	Kobe4	Eindhoven3	Paris3	NY3	Istanbul3	Mexico2
Ri-one2014	0.963	0.220	0.990	0.577	0.180	0.813
Ri-one2015	0.960	0.230	0.990	0.540	0.180	0.790

Table 5. Comparison of Civilian Component on Preliminaries Day 1 and Day 2

	VC1	Eindhoven1	Berlin1	Paris1	Kobe2
Ri-one2014	206.680	221.335	131.366	200.672	82.777
Ri-one2015	198.742	216.902	127.188	191.622	81.360

Table 6. Comparison of Civilian Component on semifinals

	Berlin2	Paris2	Eindhoven2	Istanbul2	Kobe3
Ri-one2014	104.993	144.470	103.320	104.993	117.365
Ri-one2015	114.105	134.935	109.837	114.105	116.108

Table 7. Comparison of Civilian Component on finals

	Kobe4	Eindhoven3	Paris3	NY3	Istanbul3	Mexico2
Ri-one2014	159.810	29.235	83.687	61.280	27.047	205.973
Ri-one2015	158.215	29.573	74.060	56.765	28.060	197.615

6 Acknowledgment

We are deeply indebted to Chihiro Hatakeyama, Sho Okazaki, Ko Miyake and Tetsuaki Fujimoto for their advices.

References

1. K. Miyake, S. Oguri, M. Takashita, T. Fukui, T. Mori, Y. Takeuchi, and N. Sugi. RoboCupRescue 2014-Rescue, Simulation League. Team Description. Ri-one(Japan), 2014.

2. Ajith. Abraham, Crina. Grosan, Vitorino. Ramos, Satoshi. Kurihara and Kenichi. Fukui. Swarm intelligence in data mining, Japan, 2009. ISBN 9784501550905

3. 李志祥. (2011). Based on mobile agent Distributed Ant Colony Clustering Linearization method. http://www.cs.is.noda.tus.ac.jp/ mune/thesis/li.pdf