RoboCupRescue 2013 - Rescue Simulation League Team Description Ri-one (Japan)

Sho Okazaki, Takahiro Nakagawa
Ko Miyake, Shinya Oguri, Masahiro Takashita

Ritsumeikan University, Japan
is0105rs@ed.ritsumei.ac.jp
http://rione.org/hp/

Abstract. This paper describes the specialty which Ri-one developed for the RoboCup Rescue Simulation (RCRS) in 2013. Our team developed the route search applying point of visibility navigation graph, estimate the spread of fire and clustering of buildings on the map last year. Moreover, our team this year implemented a clustering algorithm by the use of K-means. This is implemented in combination with clustering of last year. In addition, we calculate the density of entities which have a high priority and estimate the time of death of a civilian. Therefore,our team is focused on cooperation between the agents than last year. This paper describes the algorithms and application to those agents.

1 Introduction

The RoboCup Rescue Simulation is a multi-agent simulation for the disaster relief. RCRS server simulates disasters which reproduces a city after an earthquake. The agents which work in this environment are called ambulance team, fire brigade and police force. They have to rescue the buried victims and extinguish the fires while they clear the obstacles.

Our team developed the route search applying point of visibility navigation graph, estimate the spread of fire and clustering of buildings on the map last year. However, this clustering cannot create all clusters equally. This year, we developed a new algorithm for clustering. This clustering algorithm have used in combination with the clustering algorithm in last year. Therefore, a new clustering create more equally clusters than the clustering in last year. In Section 2.1, we introduce the new algorithm. In addition, we have developed a efficiently method to use limited computational time of agents. In Section 2.2, we introduce the new method. Moreover, we calculate the density of entities which have a high priority. We have prevented the congestion of agent by giving priority to the same entity. In Section 2.3, we introduce the new calculation method. In Chapter 4, we introduce the use of these on the agents.

2 WorldModel

2.1 AreaPartition

It stands that grouping buildings by clustering proves to be useful, since it allows us to distribute agents evenly. By using cluster analysis with the linear measure of distance, we can obtain groups of buildings. One of the clustering methods uses length as a common characteristic in order to form clusters, and therefore is suitable as a means of dividing groups of buildings. Last year we placed buildings which were close together within a certain distance as groups, using this technique. However, the problem with this is that the numbers of clusters become dependant on the map. To solve this problem, this year we used the K-means technique as well as the area partition analysis used last year.

The K-means technique is one of many algorithms used in cluster analysis, and performs clustering by adjusting the number, k, of median points. For this year's RCRS, we took the map from last year which already had the area partition analysis applied to it, and took each cluster to be a node and further applied the K-means algorithm. Fig.2 shows the clustering with the K-means technique applied on top of last year's Area Partition analysis. Using this method of area partition, one can distribute the agents evenly, without them cliquing together in same areas. With this technique, police forces can now find obstacles efficiently in large areas, and can clear the debris effectively, since there will no longer be multiple police forces trying to clear obstacles which can be cleared by a single police force. Thus, by setting k without rein on the number, we have made it possible to adjust the number of clusters freely.

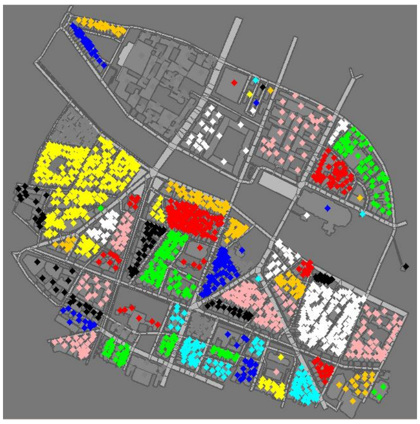

Fig. 1. Last year's clustering with area partition analysis

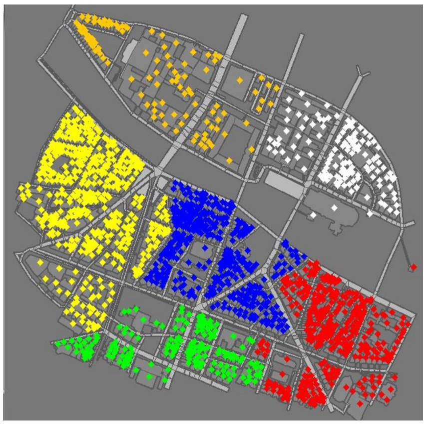

Fig. 2. The clustering with the K-means technique applied as well as last year's area partition analysis

2.2 Union-Find algorithm

Previously, we had used the breadth-first search to decide reachable. However, there is a possibility that execution of breadth-first search is inefficiently when computational time was limited. For example, if there are many reachable places by breadth-first search, this will be difficult to compute. Therefore, we improved the processing time by the Union-Find algorithm [2]. This algorithm unite two groups and decides that two elements in the set are a same group efficiently. The time complexities of the operations are O(a(n)). The a(n) is the inverse of ackermann function A(n, n). By using the operations, it is enough to unite pair becoming a new reachable. This algorithm does not concentrate computation in one cycle, because a new consolidated subgraph are already calculated in other cycles.

However, this algorithm cannot be the operations to dividing the group. Therefore, this algorithm need to build again the group. This algorithm increases the processing time by building again the group. As a result, our agents judge by using breadth-first search and Union-Find algorithm.

2.3 Prioritize

Previously, our agents decided a priorities of the entities by using information from view and communication. However, there is a possibility that they give priority to the same entity when they decided independently the entity. As a result, they will be crowded in a particular location. Therefore, they prioritized by using the location of the entities when we should give priority to entities.

Fig.3 shows the model of priority order. This method creates a circle by two specific points. In addition, that compute the density of the points on the map.

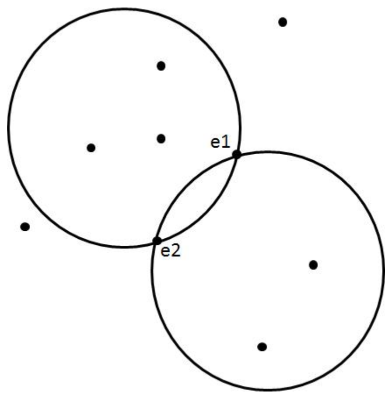

Fig. 3. A circle by two specific points

As a result, by computing the density of entities that are priority, agents judge independently the entity, as well as entities can judge crowding in a particular location. Therefore, agents prevented to give priority to the same entity. In addition, they can reduce that they are crowded in a particular location. After that, this method becomes a problem to compute the density for each cluster by Area Partition.

3 Communication

3.1 The control of action range of agent with center agent

In RCRS, it is the best if agents as a team be made to act so that they will obtain newer information.The behavior of each agent is determined entirely based on the information that it holds.Therefore, making the information than an agent holds more recent will result in behavior that is more accurate and efficient. The entirely new information that an agent can obtain during each cycle is limited to the information within the agent's field of view.

Information obtained from another agent via communication will be past information from the perspective of the agent that sent the information, and known information from the perspective of the team as a whole.Consequently, the information held by the agents as a whole will be more recent if each agent behaves so as to obtain unknown information. Accordingly, acquiring unknown information should be prioritized over the updating of known information.This is because it is extremely difficult to predict completely unknown information, but it is easier to extrapolate the future of known information. For example, if an agent has ascertained the conditions within a given building in the past, the possible conditions within the building in the present will be limited. Conversely, if the conditions within a building have never been ascertained, any of the potential conditions within the simulation could be predicted.Therefore, information that is unknown from the perspective of the team as a whole should be maximized.

Problems such as the following may interfere with the collection of unknown information by agents:

1. When the distance between agents is small
   An agent's field of view determines the range within which new information can be collected. However, when multiple agents are close together, the information within their fields of view will contain the same details, causing redundancy.This causes a reduction in the new information that can be obtained per cycle. There will be redundant content in information within areas where circles overlap.
2. When agents are unevenly distributed
   When agents are unevenly distributed around the map, information in areas where there are no agents cannot be collected. Areas without agents could be considered areas with much unknown information.

In order to handle such situations, behavior such as the following will be considered, using area partitions and a center agent. The reason for using a center agent is that the center agent is specialized for communication. Based on the information sent in from each agent, the center agent will send communications to specific agents instructing them to perform information collection in areas with unknown information that have been created by area partitions.Agents that have received these instructions will take action based on this information.Using communication in this way to control agent behavior will make it possible to collect a greater amount of unknown information.

4 Agent Skills

4.1 Ambulance Team

Ambulance teams rescue victims in a building, and carry them refuge. In RCRS, it is important that as much victims as possible live. Until last year, an ambulance team carries a victim refuge soon if it is possible to load a victim. However, this method is inefficient when an ambulance team rescue some victims in same location. A damage of buried victims continues increasing. Ambulance teams should prevent the increase in a damage. To solve this problem, ambulance teams rescue buried victims when ambulance teams can carry a victim and buried victims exist in same location.

The following figures show a process that one ambulance team rescue two buried victims. Fig. 4 shows a method that ambulance teams carry victims refuge soon. This method was used until last year. Fig. 5 shows a method that ambulance teams rescue victim who have a high priority. We made this method in this year. As a result, ambulance teams can rescue as much victims as possible and the increase in a damage of victims can be prevented.

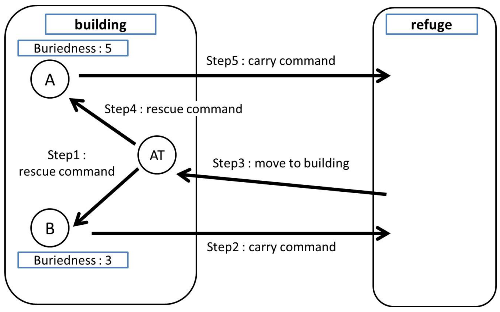

Fig. 4. The rescue method until last year

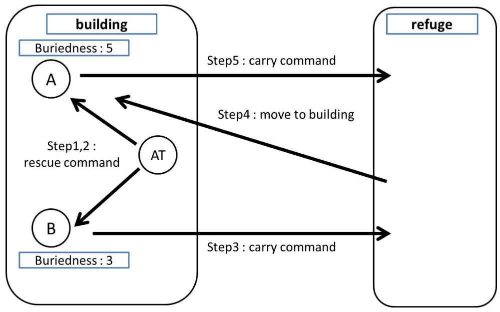

Fig. 5. The rescue method in this year

Fig. 6 and Fig. 7 shows graphs about damage of victims in Fig. 4 and Fig 5. In those graphs, an ambulance team takes 10 cycles from the building to the refuge, and damage of a buried victim increases per cycle x points. The size of those graphs are the total amount of damage. Therefore, this method decreases the total damage.

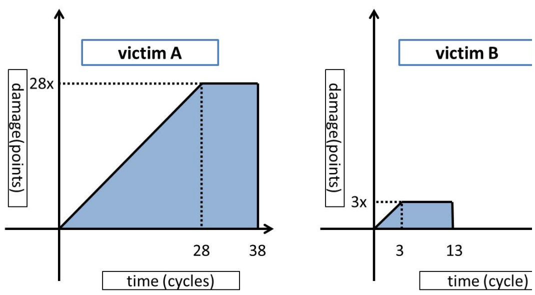

Fig. 6. The damage of victims in Fig. 4

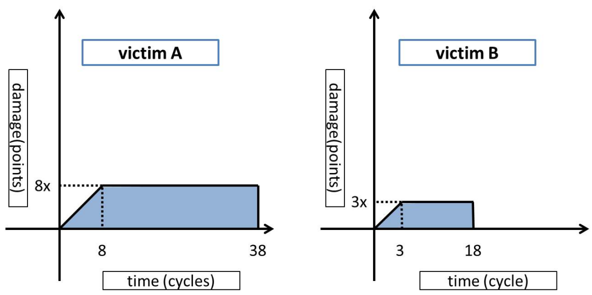

Fig. 7. The damage of victims in Fig. 5

4.2 Fire Brigade

Fire brigades extinguish fire caused by disaster. It is important to prevent the spread of fire. A burning building raise the temperature of the surrounding buildings. Therefore, fire brigades extinguish an outside of burning buildings and prevent spread of fire. Then, it is necessary to find burning buildings and outside of them.

We solve this problem by using Area Partition in Section 2.1 and top of the convex hull. Burning buildings are got by using Area Partition which made the node of burning building. An outside of them is got by top of the convex hull. The convex hull is the smallest convex set includes a given set. To get top of the convex hull, first reference point is made into a node of the smallest x-coordinate in burning buildings. Then, fire brigades compute an angle of the last reference point and current reference point on the basis of x-axis. The first angle is treated as 0 degrees. On the basis of this angle, fire brigades compute next reference point which is point of the smallest angle with the current point. Continuing this method, it is made a convex set of reference points. This set is an outside of burning buildings. Fire brigades choose this set as target to extinguish.Therefore, fire brigades can extinguish efficiently and prevent spread of fire.

4.3 Police Force

Police forces clear the obstacles caused by the disaster on the road. The police forces can make roads to pass other agents. When the obstacles block a road, agents except police forces cannot go along the road and are prevented doing their action. In order to do the ambulance team and fire brigade's action efficiently, they must clear the obstacle efficiently. Based on information from communication, a police force clears the obstacle preventing the agent's action preferentially. The police forces of Ri-one in 2012 set priority to entity, and move to entity of the highest priority with clearing the obstacles.

They of Ri-one in 2013 did the method in 2012 and set priority to crowding entity by the Prioritize in Section 2.3. Therefore, they became to be able to clear the obstacles more efficiently. Specifically, using the Prioritize computing a density of crowding entity based the information from communication, we set high priority to the obstacle that is not reachable with the high density of agents or civilians around and make police forces go to it. Therefore, police forces can make roads that have high frequency in use traversable preferentially.

5 Result

Table 1-3 are the result of comparison with the agents in Ri-one 2012 at RoboCup 2012 Rescue Simulation League competition. The scores in some maps are increasing. This score is caused by Area Partition and Prioritize. However, the some scores are decreasing almost slightly.

Table 1. Comparison of Preliminaries Day 1

	Berlin1	Eindvoven1	Kobe1	Mexico1	Paris1	VC1
Ri-one2012	37.489	25.976	64.969	74.706	109.664	121.074
Ri-one2013	42.513	17.923	75.326	81.23	109.03	114.062

Table 2. Comparison of semifinals

	Berlin3	Berlin4	Eindhoven3	Istanbul2	Kobe3	Mexico2	Paris3	VC3
Ri-one2012	15.418	44.778	5.025	38.524	85.330	34.072	39.473	154.570
Ri-one2013	16.33	38.87	5.89	28.51	84.82	34.501	35.17	140.646

Table 3. Comparison of finals

	Berlin5	Eindhoven4	Eindhoven5	Kobe4	Mexico3	Paris4
Ri-one2012	36.499	4.138	39.155	47.350	33.021	9.614
Ri-one2013	39.15	4.474	18.12	52.76	33.025	11.348

6 Acknowledgment

We are deeply indebted to Masatoshi Tsushima and Keisuke Yasuda for their advices.

References

1. Toby Segaran. Programming collective intelligence : building smart web 2.0 applications. O'Reilly. 2007. ISBN 9780596529321.

2. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. Introduction to Algorithms, Second Edition. MIT Press and McGraw-Hill, 2001. ISBN 0-262-03293-7.

3. M. Tsushima, K. Yasuda,S. Okazaki, T. Nakagawa, and Y. Kurose. RoboCupRescue 2012-Rescue Simulation League. Team Description. Ri-one(Japan), 2012.