# RoboCup Asia Pacific 2018 – TDP Rescue Agent Simulation M.C.S. (Iran)

Armin Kazemi, Hamid Ziari, Zahra Golpayegani, Ava Abdrezaei, Amir Kiani, Fatemeh Babaei, Mona Mehrabizade Honarmand, Mohammad Hassani Saadi, AmirMohammad Noori, Saeed Shiri Gheydari

Amirkabir University of Technology, Iran

arminkz@live.com , aut.ziari@gmail.com

**Abstract.** Problems such as fire expansion, having some agents stuck in the blockades and very limited access to different parts of the map are some of the major difficulties which rescue agents need to face after an earthquake. In this paper, our team's some new solutions for these problems are discussed. Including strategies like fire expansion controlling by determining buildings which have the greatest impact on the fire spread, and enhanced methods for clearing paths to important parts of the map, considering the priority of them. Also Ambulance Agents Priority to rescue civilians discussed.

## 1 Introduction

RoboCup Rescue Simulation is a Large Multi-Agent System which its aim is to manage the disaster when an earthquake happens. Usually after an earthquake, destruction of buildings and roads results in blocked roads and building entrances. So ability of Ambulance teams and Fire brigades to search becomes very limited and they may lose access to important places like refuges and hydrants which clearly decreases their performance. In such situations, Police Forces play the major role. As a result, MCS team's focus for RoboCup-2018 competitions, was on finding strategies to accelerate PF's operation in the initial cycles of simulation. Such as finding a sequence of targets which minimizes the total time spent on clearing these targets and an enhanced method for connecting clusters to each other. One of the advantages of these strategies is that we can set higher priority to some of the targets which may be more important than others in different situations.

We also worked on a strategy with the aim of controlling fire spread in a more effective way. FBs use convex hall to find buildings on the border of the fire zone and use our new method for determining buildings which have the greatest impact on the fire spread, in order to prioritize buildings on the border. In the following, all FB's , AT's and PF's strategies are described in detail.

# 2 Strategies

## 2.1 Police Force

Police forces have the ability to clear paths, open the entrance of buildings and free stuck agents. So in spite of the fact that Police forces operation does not effect on the score directly, the performance of other agents is highly dependent on them. In this section S.O.S team's methods are described that are used to handle each of Police forces major tasks:

### 2.1.1 Making most parts of the map reachable

Each police is given one cluster in order to prevent them from wasting time on traversing paths that are already cleared by other PFs and clear more paths in less time totally. According to that, the need for agents to have access to most parts of the map is met by connecting center of police clusters to each other by an "minimum spanning tree" on center of clusters.
Each PF is responsible for clearing one edge of tree and the allocation is done in precomputation phase (Fig.1). After that PFs start connecting important points of their cluster to this MST in order to expand the accessible area.

Fig.1: MST and share of each Police force

### 2.1.2 Freeing stuck agents and opening building entrances

In some situations there are some agents stuck in the blockades that need to get free by police agents. So non police agents must get checked by PFs, to find out if they are stuck or not. In addition to FBs and ATs, refuges and buildings need to get checked too. So there are some points in each cluster that the PF assigned to it should traverse and take appropriate action if needed.
Traversing should take place fast and by considering the importance of the targets. For example in some situations freeing FB agents is more important than opening entrance of a refuge.

According to these conditions, the problem is similar to ”Traveler sales man problem” which finding a sequence of the points that minimizes the total traveled distance is desired. But here we need to minimize the total time that it takes for PF to visit each point. This way we can give some points a coefficient for their approximated delay in order to force the algorithm to choose these points as initial points of sequence.

$$Error = \sum_{i=1}^{n-1} \alpha_{i+1} \Big( Distance(1, i) + Distance(i, i+1) \Big) \tag{1}$$

We use ”Hill climbing” algorithm to minimize the total delay and a greedy algorithm similar to ”Nearest Neighbor” to find the initial solution. After finding initial solution, algorithm attempts to find better solutions within its share of time in each cycle. This way, at the end of cycle, the final solution is better than or equal to the solution found by greedy algorithm.

Connecting the clusters to each other is the first thing that PFs do after simulation is started. The amount of time that it takes for an agent to complete this task depends on size of map and number of PF agents. So it may take too much time before they start freeing agents and it's not appropriate for many reasons like the ones mentioned below:

- Buried and damaged civilians have a limited life time and less free ambulances means more dead civilians in initial cycles.

- Ignition in a building can take place in any time and location. chance for a building to be seen by agents in initial phase of burning decreases if some of agents are blocked. so the fire spreads more and controlling it becomes harder.

So unblocking agents shouldn't be postponed until the PFs finish their first task. This issue can be handled by doing these two tasks at the same time. That means PFs choose some stuck agents while clearing their allocated edge of MST and recalculate their path in order to free them. Choosing among stuck agents is performed by back tracing algorithm which chooses most number of agents with a path that is not longer than 130 percent of initial allocated edge length.

#### 2.1.3 Make Fire Brigade Reachable to Fire zone

One of the other changes in Police Force is adding an Interrupt State for connecting a Fire Brigade to its fire. This works in the way that, if a Police Force sees or senses a fire, the agent starts estimating the fire's size. If it guesses that the fire is rather small in size and the Fire Brigade that will be assigned to it is in this agent's clear list, the Police Force will drop its task and first free a path between the Fire Brigade agent and the fire, so the fire will be extinguished sooner and with less waste of energy.

#### 2.1.4 Check Fire Probability

Another feature added to the Police Force is the Check Fire Probability State. In this state, if the police agent senses some buildings with high temperature, the agent starts estimating the probable place of fire. If the estimated pace of fire is near the agent, the agent clears the road to it, so there would be the possibility to stop fire in its beginning of ignition.

### 2.2 Fire Brigade

It is Fire Brigades responsibility to prevent fire from expansion and save the city from the heavy damages that it causes to buildings and the civilians inside them. FBs have a limited time to control fire before it reaches an uncontrollable state and their water quantity is finite. They need up to date information about the shape of the fire zones and the importance of different buildings in order to compare them to each other and use the mentioned resources effectively. In the following we define what makes a building important why it is important to specify important buildings.

#### 2.2.1 Detecting Neighbors

Heat of a burning building can transfer by conduction and radiation. So the neighbors of a burning building have the highest chance to become fiery.
We've used a simple Ray Casting system to determine neighbors for each building in the precomputation time. This is done by setting a ray source on each building and then casting a reasonable number of rays in a reasonable range to determine neighbors for that specific building.
This data then can be converted to an adjacency matrix to save adjacency data for the entire map. Because this matrix is usually sparse we've used adjacency list to store it in a memory efficient way.

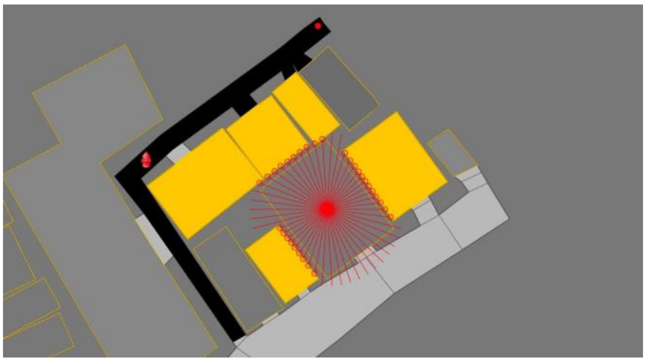

Fig.2: Ray Casting on a sample building

### 2.2.2 Scoring Buildings Importance

There is an obvious need for extinguishing buildings that are more important. What defines importance of a building includes the fieriness and number of civilians inside it, which we call it ”Base score”, plus a proportion of importance of nearby neighbors in the adjacency matrix.

$$BuildingScore = BaseScore + \alpha \times NeighborBaseScore_1 + \alpha^2 \times NeighborBaseScore_2$$

Here $NeighborBaseScore_1$ and $NeighborBaseScore_2$ are the base score of first and second level neighbors of building. Also $\alpha$ is a number between zero and one.
Fire on a building can rapidly expand to nearby buildings and the speed of expansion is dependent on number and ignition potential of neighbors of that building. Therefor extinguishing the building that has important neighbors (or neighbors of neighbors) impacts more on controlling fire spread.
This is done in a two-pass algorithm, one for calculating base score and one for final score.
After calculating score for each building, we can have a total score for each cluster. Number of agents assigned to each cluster is related to importance of that cluster. For example, a cluster with score of 200 may have agents twice as the cluster with score of 100.

### 2.2.3 Determining the Fire Zone

The Buildings on fire which are near each other create a fire zone. Information about the shape of fire zone is useful specially when fire is too strong and extinguishing is very hard.
Sometimes it’s better to leave the buildings inside the fire zone and try to prevent the expansion of the fire instead. To achieve this, agent tries to cool down the buildings that are on the border of the fire zone. To find buildings on the border of

fire zone we used Convex-Hull algorithm for fiery buildings inside each cluster. Agents are distributed evenly on the border of fire zone in order to control the fire. Agents choose among the buildings on the border of fire zone based on their score. Since the higher score indicates that the building will cause more destruction and faster spread of fire, if it continues burning.

### 2.3 Ambulance Team

Decision making has been changed to centralize deciding, based on each AT's current Task, the cost of rescuing each civilian by each AT and etc. In our algorithm, each will have two targets to decide between them. One of them is chosen by the Ambulance center. The agent also chooses his target and then decides between these 2. The reason that we find 2 targets is to only rely on center deciding but to also consider problems caused by reachability errors and communication faults.

We used a particular filter to determine estimated civilian dying time but the results of past competition showed some mistakes in evaluating dead time. So we decided about utilizing new learning methods like using decision tree to determine the dead time of civilians. Last year task assigning was individual and every civilian had only one ambulance agent to rescue, but in last cycles of simulation in one map it was seen that most of civilians were not chosen because of low remaining cycles. So we lost the score relating to them. In this year's strategy we want to define new situation that ambulance agents will work simultaneously to rescue civilians with low HP.