

POSEIDON TEAM DESCRIPTION PAPER

ROBOCUP 2013, EINDHOVEN

Afsoon Afzal², Parand Alizadeh¹, Mozhdeh Arian Nezhad³, Kimia Ghaffari¹, Ghazal Hasanpour¹, Kiana Jahedi¹, Pooria Kaviani², Giti Omidvar¹

1 Farzanegan High School, Robotics Research Group, No. 56, Shahid Sarparast St., Taleghani Ave.,
Tehran, Iran

2 Sharif University of Technology, Azadi Ave., Tehran, Iran

3Tehran University, Enghelab Ave, Tehran, Iran

afzal@ce.sharif.edu, parand1997@gmail.com, m.ariannezhad@ut.ac.ir, {ghaffari.kimia,
ghazalhasanpour, kiana.j97, pooria.kaviani, giti.omidvar} @gmail.com

Abstract. The objective of this manuscript is to technically explain Poseidon team's contributions to rescue agent development. Each section describes our algorithms and methods that are implemented in an efficient way. We use artificial intelligence algorithms and probabilistic methods to solve the existing problems in the rescue simulation league. This version of the Poseidon team is based on the sample code of the server with extensive changes in structure of agents.

1 Introduction

The purpose of Rescue Simulation League is to decrease life and financial losses caused by natural disasters such as earthquakes, floods and etc. In order to achieve this goal, a large urban disaster is simulated and indicates agents' actions in this situation. This simulation matches real world limits and problems as accurately as possible. Rescue simulation agents include ambulance team agent, police force agent and fire fighter. The main task of the police force is opening the closed roads, the main duty of the ambulance team is saving lives and fire extinguishing is the main duty of the fire fighters. In addition, all the agents are responsible to facilitate the other team's tasks. The Poseidon team is trying to find solutions and improve available algorithms to solve agents' problems.

Our contributions include changing police force agents' zoning algorithm, which has been explained on Poseidon 2012 TDP[1]. This approach improves compartment of the map to determine tasks for each police agent justly. Since the main duty of the ambulance team agent is to rescue the maximum number of civilians, selecting injured civilians is noteworthy. Thus we implemented a new idea to choose and rescue civilians. In addition, a new algorithm for the fire brigade agents' search and centralizing the fire brigade agent and ambulance team agent is implemented.

2 Ambulance Team Agent

Due to the importance of civilians' life in real world, one of the most important purposes of Rescue Simulation is to reduce the number of victims. This is team agents' job to do.

In the process of decision making the ambulances may decide autonomously (distributed) or the commanding center may decide for them (centralized). Also the ambulances can be sent to rescue the civilians in groups or individually. The decision-making process is a complicated task. This is due to the fact that there are some issues such as estimating the time each civilian takes to die, the time that is needed for the agents' movement in the city, and the time that the fire takes to spread and burn the civilians.

In Poseidon 2012 [1] the algorithm of ambulance agents was designed in order that each agent decides and selects its target individually. In order to make the best decision, ambulance agents need to be informed of others' decisions. Although limited communication and radar noises would make this difficult, using a centralized algorithm is more efficient. According to Poseidon 2012's [1] radar system, the ambulance center has more accurate information about the position of ambulance agents, the position of the injured civilians, the estimated death time for each civilian, the damage of each agent and civilian, etc. As a result center based ambulance agent is implemented in Poseidon 2013.

2.1 Selecting the injured civilians

In the ambulance problem there are a number of civilians that can only be rescued in a specific time. Each civilian, in order to stay alive needs a certain number of ambulances, while the number of ambulances is limited. Planning for rescuing civilians and determining the number of ambulances they need is similar to a Knapsack [4] problem. In Knapsack algorithm we are given a set of items each with a cost and a value. We try to choose the items with most value but due to being limited in charge we must choose the items with minimum cost, therefore, the goal is to determine items to include in a collection so that the total doesn't exceed some given cost, *capacity*, and the total value is as large as possible.

In this case maximizing the value is equal to maximizing the number of saved civilians. Since the number of ambulances and the game length is constant, the number of civilians that can be rescued is limited. This indicates *capacity* in Knapsack algorithm. In ambulance agent's problem each civilian has a value and a cost that is determined regarding to multiple factors:

- Estimated death-time: We changed Poseidon 2012's [1] death-time estimation algorithm to ZJU 2008's [2].
- The civilian's position: In this factor we consider its distance from fire points, refuges and available ambulances.
- The number of needed ambulances: It is set according to civilian's health point and damage.
- The time is needed to save the civilian: It is set according to the time is needed to the ambulances' moving to the civilian's position, rescue the civilian and moving to the nearest refuge.

Each civilian's value is determined as follows:

- If the civilian is alive its value is equal to 1.
- Else the civilian's value is equal to 0.

Since we only consider the living civilians in ambulance problem and all the living civilians have the same value, and in this situation in knapsack solution the greedy algorithm is considered, we used the greedy algorithm for the solution of our problem.

Greedy algorithm for the solution of the knapsack problem is as follows:

- Compute value/cost ratio for each item.
- While knapsack is not filled, take the next one with greatest value/cost ratio.

Due to all civilians' same value this algorithm in ambulance problem is as follows:

- Compute cost for each civilian.
- While the game is not ended and all ambulances are not busy, save the next civilian with minimum cost.

2.2 Decision-making process

Based on the Knapsack algorithm the ambulance center sets a task for each ambulance agent. The tasks are sent through radar system, as a message containing civilians' positions. After receiving the tasks sent by the ambulance center, each ambulance agent chooses to act based on center's given task or makes a decision considering its own information. Comparing the center's given information and its own, each ambulance agent selects the target that is seen most recently.

In defined position sent by center might be more than one civilian. Regarding to civilians' conditions, such as death time and available ambulances in mentioned position, each agent would make the best possible decision.

2.3 Poseidon ambulance result

The result of the comparison of Poseidon 2012 and Poseidon 2013 is shown in Fig 1. Each point in the figure represents the score of team in defined map.

There are no fire brigade agents, police force agents and blockades in these maps. We only compared ambulance agents.

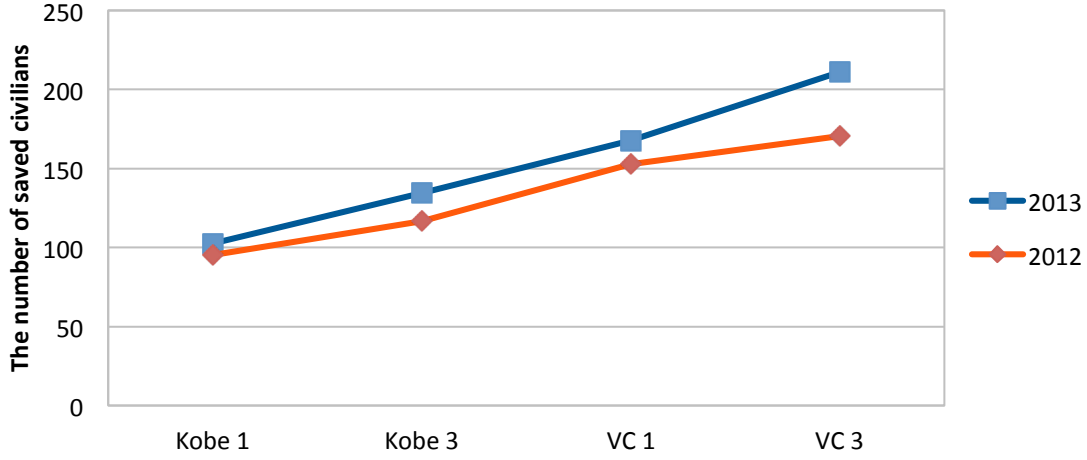


Fig 1. Comparison of Poseidon 2012 and Poseidon 2013

3 Police force agent

The most important duty of the police force agents is to open the roads, which has noticeable influence on other agents' operations. Thus, if police force agents don't do their responsibilities as good as possible it may cause large number of other agents stuck in their ways and as a result they won't be able to do their tasks.

To determine where each police should acquit his clearing obligation, we divide the map into a number of zones. In Poseidon 2012 [1] the map was divided into some sectors and each zone contained a number of buildings, as some zones might be too big to be seen entirely during the game. So we implement a new method of zoning to solve this problem. Since police force agents' main duty is clearing the roads, it's better to divide the roads instead of buildings. To do this, in the first step, main intersections, which is the place of connection of some roads becomes specified. After that the roads between any two of them are connected to each other using BFS algorithm. This approach uses less process comparing with similar algorithms as a result of being implemented in the post-connect. In this case some paths will be formed (Fig 2). We add the roads, which are connected to buildings (entrances) in the end of each path as extra roads and they have less priority for being cleared comparing other roads unless there is a civilian in their buildings. These paths consist, not only roads but also buildings for the occasion that police agents want to search civilians as we explained in Poseidon's 2012 TDP [1]. After making the paths, we assign a number of them to each agent using "Simulated Annealing" algorithm explained in Poseidon's 2012 TDP [1] (Fig 4).

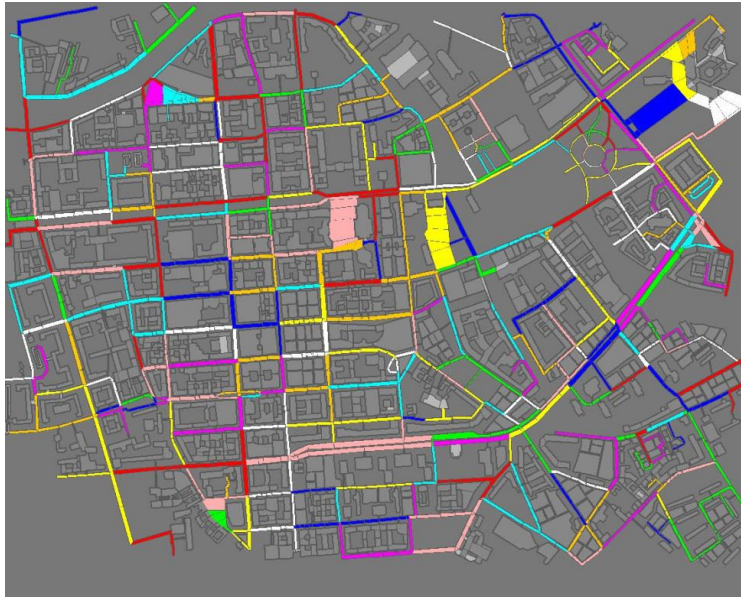


Fig 2.Path Generating



Fig 3. New zoning algorithm

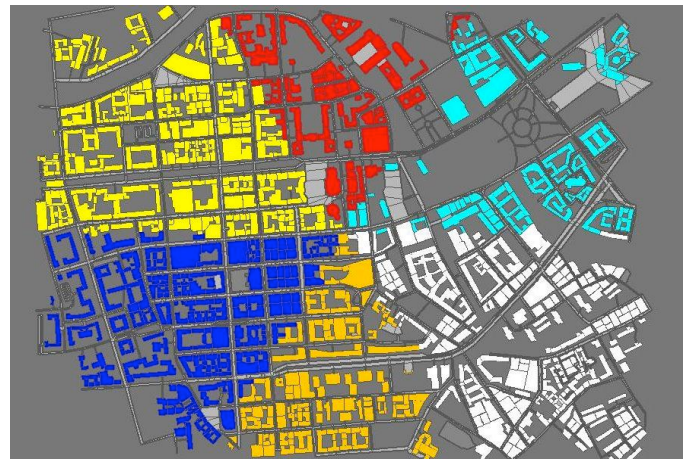


Fig 4. The Poseidon 2012 zoning algorithm

4 Fire Brigade Agent

Coordination between fire brigades is very important but when they decide alone due to limited communication, coordination is difficult. As a result of creating harmony between fire brigades, we use the fire station to make decisions.

Due to the lack of the requirement of making decisions independently by any fire brigades, being Centre base will save time. But sometimes because of limited or shortage of communication, they can't hear the fire station decisions. When fire brigades do not receive the message from fire station, decide on their own terms by Poseidon 2012 [1] algorithm.

The Poseidon divides city into some sections named “Zone” by the Poseidon’s 2012 [1] zoning algorithm. Each zone is a special area in the city, which has a number of ignited buildings.

Connection status among entities should be informed to center. Having the information about the connections between roads and buildings is very effective and can help making better decisions. Therefore we make fire groups using BFS algorithm. Each fire group has buildings, roads and fire brigades that connect with each other. Fire station sets tasks for any fire brigade, which is available for them by using fire groups. In other words, fire groups will show the reachable areas for each fire brigade agent.

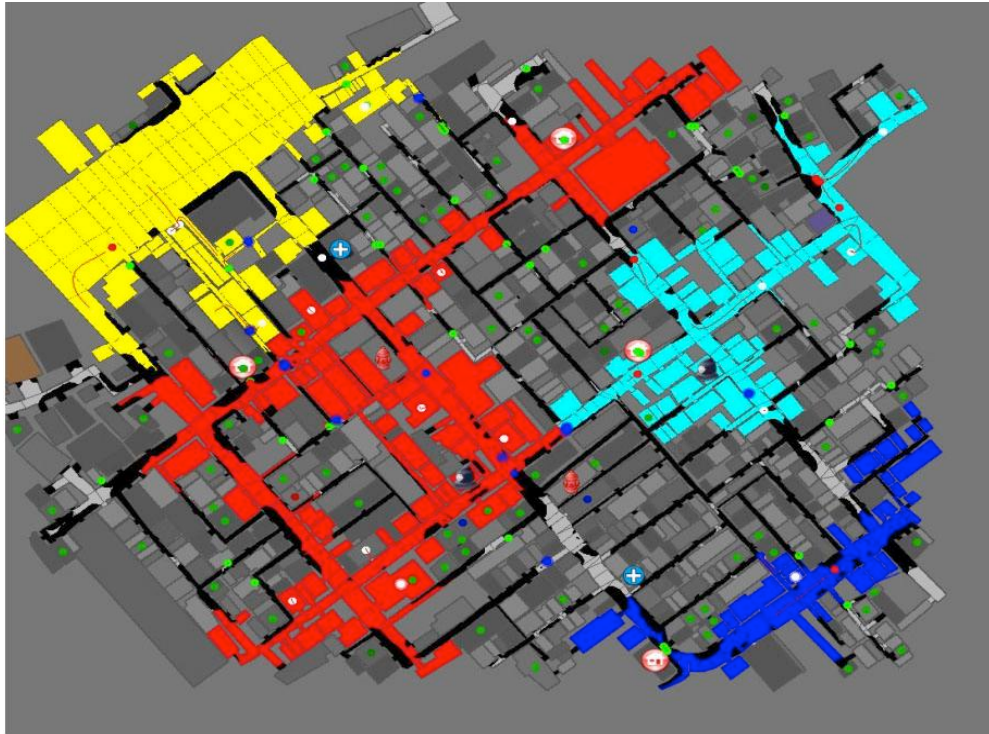


Fig 5. Fire Groups in Kobe

After that fire groups are formed, the fire station allocates some fire brigades to each zone. The fire station examines every building of the fire zones and finds the points that are in different roads. A fire brigade on that point can extinguish some buildings of that fire zone. The fire station determines fire groups that have access to those points. Thus, each fire brigade of a fire group can extinguish some buildings of accessible zones. Consequently, the fire station assigns a fire zone to each fire brigade. After that, the fire station defines fire brigades and zones as vertices of the graph, then connects them using “Perfect Matching Algorithm”. To connect the vertices by the edges, it is important to consider that the total length of the edges must be the minimum. Then, the fire station assigns a number to each zone. This number shows that how many fire brigades require for each zone to extinguish that fire zone. Based on using this numbers and “perfect matching algorithm” [3] at finally every fire brigade is assigned to the most proper zone as its duty.

After choosing a zone for each fire brigade, that brigade is obliged to extinguish one of its zone's buildings. The most important buildings in each zone are the buildings that are located on the boundary of the zone. The fire station defines a value for each important building. To define that value the fire station must consider some factors: 1. The fire station checks which side of the chosen fire zone will cause more spread of the fire in the next cycles (by using simulator explained in the Poseidon's 2012 TDP [1]). 2. The time that is needed to extinguish a building is estimated according to ground area, fieriness, total area, temperature and floors. The Fire station assigns fire brigades to the important buildings of their fire zones with regards to their distances from their positions to the buildings and the values of the buildings. Consequently, the aims are determined for each fire brigade.

4.1 Search

We use a new search algorithm to find the fire instantly and preventing the spread of it .In order to do this, the entire map is divided into smaller parts and each part is assigned to one of the fire brigades randomly. Therefore all fire brigades are dissipated on the map and after seeing all buildings of his part; agent continues to search by the Poseidon 2012 algorithm [1].

4.2 Search by temperature

Since each ignited building impact on the temperature of it's around buildings we have decided to implement a new search algorithm based on the temperature. If a fire brigade agent sees a building with the temperature above zero, it will search around that building more accurately. If an ignited building exists around, it will be extinguished. Otherwise, he will return to usual search that has been explained above. Due to the fact that fire brigade agents mostly cannot see all the buildings of a zone from their position, after extinguishing the zone, they cannot be sure that there isn't any ignited building in that zone. But according to the search zone by the temperature and the fact that right after being extinguished, buildings still have nonzero temperature, fire brigade agents will see the entire buildings of the zone and look for not extinguished buildings.

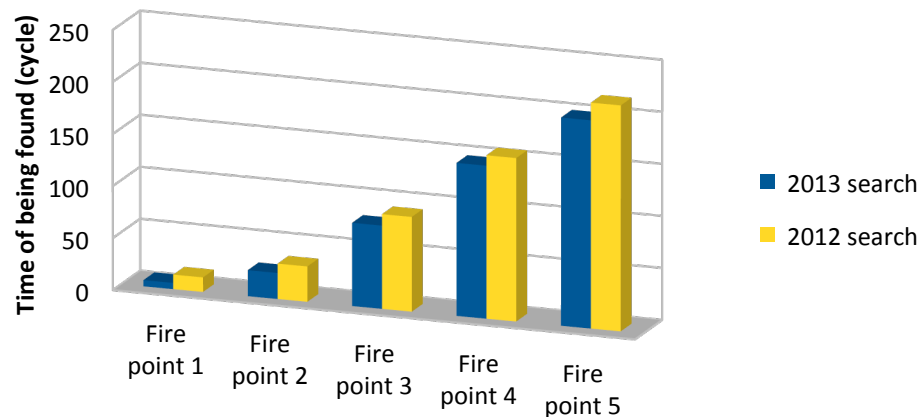


Fig 6. The comparison between search techniques

References

1. Afzal, A., Arian Nezhad, M., Eslami, M., Farzan, M., Ghasemi Zavie Sadat, S. F., Jafarian Yazdi, R., Moradi Dadkhah, B., Pourmoghaddam, R., Sanjabi, S., Seyed Majidi, N., Shah Hosseini, M., Shirvani-Mahdavi, N., Kaviani, P., : Poseidon Team Description Paper, 2012.
2. Rong Xiong, Yikun Tao, Yuankai Wang, Shunhong Zhong, Jie Zhang, Wentong Li Baixin Wang, Ye Tian: ZJUBase RoboCup 2008 Team Description.
3. S.Even, G.Even: Graph Algorithms. Cambridge University Press, second edition 2012.
4. R. Neapolitan, K. Naimpour: “Foundation of Algorithms”, Jones & Bartlett learning, 2010.