# Hurricane Team Description

Amir Reza Zareian[1], Foruhar Ali Shiva[2], Ehsan Tabari[3], Reza Samimi[1]

[1] Shahid Beheshti University [2] AmirKabir University of Technology
[3] University of Tehran

**Abstract.** This is a team description paper for the rescue simulation league of the RoboCup competition. In this paper the behaviors of the various agents are described and an outline is provided for the protocols used for the communication between each rescue agent and it's respective center.

## 1 Introduction

The Rescue Simulation Domain, provides a real time environment for cooperation between different types of agents. In this domain if the situation is left uninterrupted, it will undergo constant deterioration. Hence, coordination of agents, and finding optimum solutions are of the utmost importance. In this paper we discuss the algorithm used for each of the rescue agents, and provide and overview of the communication methods employed.

## 2 Communication Model

At the initial stage of the disaster, the Fire Station identifies the regions where the initial ignitions have occurred, and based on the average fiery ness of each region and the density of buildings in each region, assigns fire brigades to each region. The station also assigns a refugee to each region.
The Fire Station keeps a centralized world model base on the perceptions it receives from the brigades. These perceptions include the status of the buildings across the disaster space. Other than the status of the buildings each brigade also reports its current position and destination. Each time the fire station receives a status report from a fire brigade its world model is updated.
When the world model is updated, the fire station reevaluates each region, and assigns fire brigades to them accordingly.

The operation of the police Station and ambulance Center are basically similar to the Fire Station. The main difference is that, in the case of the police station and ambulance center no regions are defined, only the nodes which contain a blockade or civilian are considered according to each blockades toughness, or each civilian's buried-

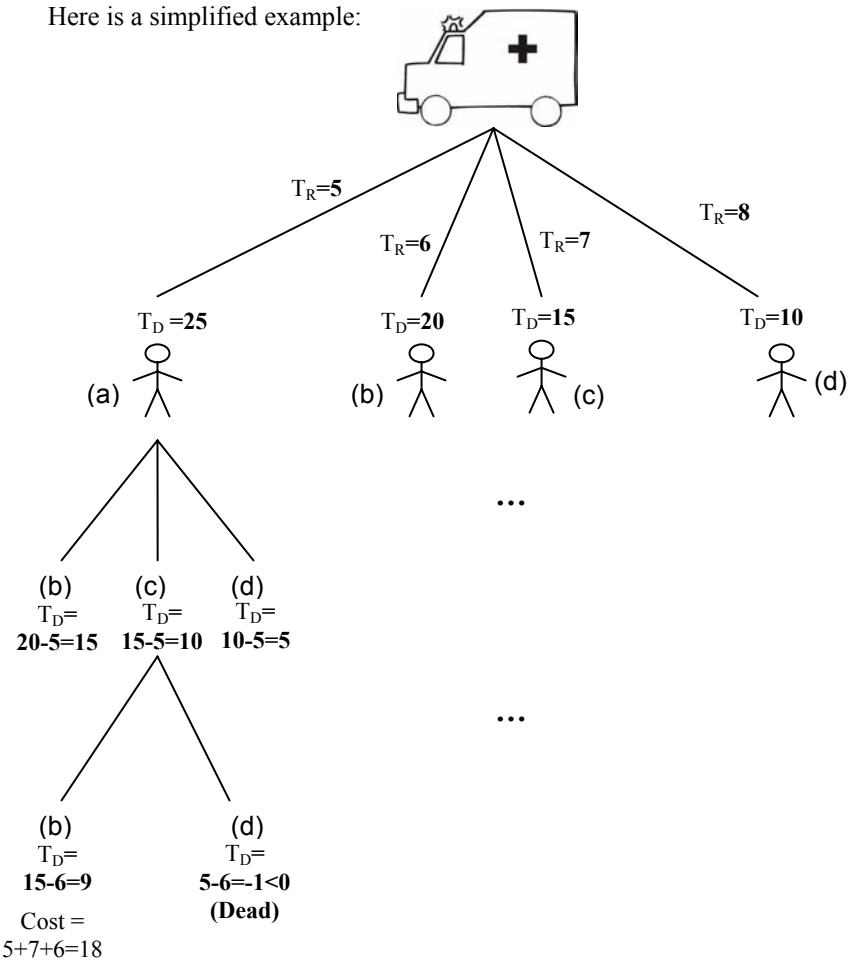
ness. Each center assigns a set of Civilians/blockades to it's teams. The Ambulance Center also assigns a destination refugee for each civilian.
The other difference is in the items which they store in their world model. The police station keeps tracks of the blockades and whether they are removed are not. The ambulance center keeps track of the civilians, their health and buried-ness.

Additionally each of the agents (regardless of type) can send a building status, civilian status, blockade status to its respective station. These reports will be forwarded by each station to the appropriate station.

## 3 Ambulance Team

After a set of civilians is assigned to an ambulance, it must decide on an order to rescue them. The algorithm for this decision considers the buried-ness and the total time required to reach each civilian to the assigned refugee. The algorithm uses a decision tree as follows: The time required to reach and rescue a civilian ($T_R$) are assigned to the edges and the nodes contain the time remained until a civilian dies ($T_D$).

Here is a simplified example:



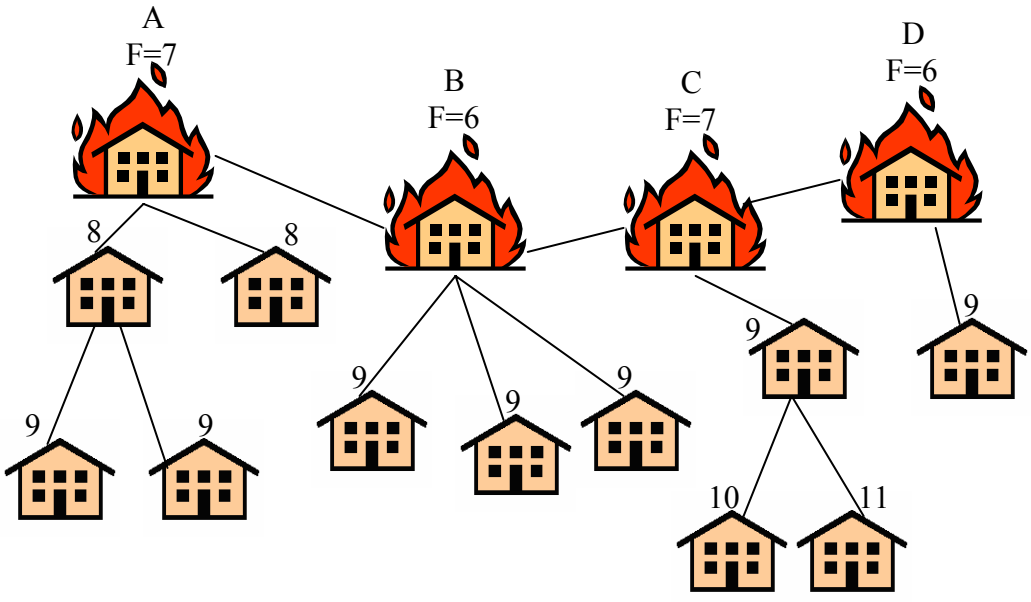
The path with minimum casualties and least cost will be selected. In the above example only part of the tree is expanded and one of the solutions, which might not be the optimum solution, is displayed. It has resulted in a cost of 18 and a single causality.

The above three will be very large for, higher number of civilians involved. The number of nodes can be calculated like this: If there are n civilians, the number of nodes will be:
$n + n*(n-1) + n*(n-1)*(n-2) + \ldots + n*(n-1)*(n-2)*...*2 =$
$n!*(1/(n-1)! + 1/(n-2)! + 1/(n-3)! + 1/(n-4)! + \ldots + 1) \cong$
$n!*(e-1)$

For example for 10 civilians there will be about 6'060'096 nodes, and for 7 civilians there will be about 8417 nodes. So using this method for smaller number of civilians is greatly beneficial.

## 4 Fire Brigades

When a set of Fire Brigades are assigned to put out the fire of a region, the algorithm used to select the buildings considers their fiery-ness and the material used in the adjacent buildings. Using this data the amount of time to put out a fire, and the amount of time for a new building to catch fire are calculated. Based on these times priorities are assigned to buildings, and agents are distributed among the ignited building based on their priority. Now we explain how priorities are assigned to buildings:



The above image shows a simple example of the spreading of the fore among neighbors. The nodes which are connected are neighbors. The F factor shows the fiery-ness of the buildings which have already caught fire. The number on top of the other buildings shows the value that, if reached, will make the building catch fire. In that case the number will be incremented each cycle.

| Cycles passed | A | B | C | D |
|---|---|---|---|---|
| 1 | 2 | 0 | 0 | 0 |
| 2 | 2 | 0 | 1 | 1 |
| 3 | | 3 | 1 | |
| 4 | | | 1 | |

The above Table shows that, if the fire is left unattended, after each cycle, how many of the neighbors of each of the ignited buildings will catch fire. For example after 2 cycles 4 of A's adjacent buildings, 1 of C's and 1 of D's will have caught fire. Now if it is assumed that putting out A, B, C and D takes 3, 2, 2 and 1 cycle each, respectively, the number of houses which will catch fire as a result of putting out each of these building is shown in the following table:

| Building which is selected to be put out | Number of buildings which fire is spread to by each ignited building | Total number of buildings which catch fire |
|---|---|---|
| A | B-3,C-2,D-1 | 6 |
| B | A-4,C-2,D-1 | 7 |
| C | A-4,B-0,D-1 | 5 |
| D | A-4,B-0,C-0 | 6 |

The fire brigades are assigned based on the priority: e.g. if there are 2 brigades available they are assigned to C and the other to A or D. These operations are performed using a Depth First Search method.

## References


1. Takeshi Morimoto, How To Develop Robocup Rescue Agent for RobocupRescue Simulation System version 0 1st edition
2. T.Moritmoto, K.Kono, I.Takeuchi, YabAI the first Rescue Simulation League Champion