Kosar 2006 Team Description

Mohsen Zamani Tabar
Seyed Mohammad Hossein Falsafy
Hadi Asghari Vahid Valizadeh
Computer Engineering Department Shamsipour University
Iran, Tehran.

www.KosarTeam.com
Rescue@KosarTeam.com

Abstract

Robocup rescue simulation intends to propose new ways for decision making in difficult situations after an earthquake. During the past few years participating teams have presented many promising solutions some of which have been unfortunately forgotten due to lack of an informative system.

To accomplish this, Kosar team has been attempting to create a database of techniques and methods used by other teams, to facilitate the analysis of these methods in order to improve on those methods or propose new techniques. In addition, our team is developing an optimized messaging system and knowledge base for each agent, so that each agent will have a powerful world model to consult upon while making proper decisions.

1 Introduction

Agents are regarded as general tools for understanding concepts; and multi-agent-systems are used as a powerful tool for simulating real world scenarios. For instance, agents can be used to simulate different societies, in which different results are gained by altering the environment.

The robocup rescue simulation aims at exploring different techniques for minimizing civilian loss and civil damaged after an earthquake. Rescue agents must make decisions based on the initial city data and new information collected during each simulation cycle (by the sensing process or by communicating with other agents). The decisions must be made in a coordinated manner with the objective of saving civilian lives.

The simulation time can be divided into three phases, according to the information agents have about the disaster stricken city:

Phase I: *The beginning of the simulation.* At this time the agents receive information on the city before the quake, and have no information of what has happened after the quake. The purpose of this phase is to analyze the initial data properly with the help of each agent's *previous* knowledge.

Phase II: *The period in which the agents know little about the disaster space.* During this period, the agents would make predictions about what has happened in the city based on incoming information. The purpose of this part is performing fairly accurate predictions upon which the agents can update their current knowledge and make decisions.

Phase III: *The time after agents have extensive information about the disaster space.* At this point our main purpose is to plan a coordinated course of action to accelerate the rescue operation.

Kosar team wishes to create coordination and cooperation between the agents in order to reduce the time spent in the second phase and increase the accuracy of the predictions used in this phase. We have defined the following steps:

i. Reviewing and evaluating the techniques and functionality used by participants in the previous years and creating a powerful database of all used techniques and methods (section 2).
ii. Analysis of the above database and various maps in order to create a knowledge base for agents to use in phase I and phase II (section 3).
iii. Creating an extremely powerful messaging system to facilitate communication between agents and obtain a *near precise world model* (section 4).
iv. Using statistical methods, data-mining and offline learning to create a *fuzzy world model* for use in phase II (sections 5 and 6).
v. Picking the best techniques presented so far and making some enhancements to these techniques for use in phase III (sections 7 and 8).

2 Database Creation

As already stated, the aim of robocup rescue simulation is minimizing civilian loss and city damage from a quake. To achieve this aim the following issues must be considered:

a. Coordination and collective work between agents.
b. Map segmentation (into district zones).
c. Target discovery and prioritization. (e.g., burning buildings for fire brigades, injured civilians for ambulance teams, and blocked roads for the police force).
d. Path finding and selection.
e. Traffic jams avoidance.
f. Evaluating the spreading of fire and the possibility of reigniting.
g. Civilian life prediction during next cycles.

Each issue includes many sub-issues. For instance, during path selection we need to know which roads are blocked and which are not, and also the probability of some unknown road being blocked. So Kosar team decided to look at the solutions proposed and implemented for each issue and evaluate the effectiveness in different environments, and finally categorize the approaches if possible.

For this purpose, we have gathered data in the following fields:

i. Extracting data on various cities.
ii. Extracting statistics on functionality of various teams.
iii. Classifying mentioned and implemented ideas in certain fields.
iv. Classifying parameters for each presented method.

To obtain this information we have used competition log files and maps and have reviewed the source code of *MRL, Persia, Caspian* and other teams. We have also reviewed papers submitted by each team, and on the whole extracted the underlying ideas.

To facilitate the evaluation process we have designed a dedicated viewer, and have also developed a program to import data from log files into a database. With these tools and using SQL for data-mining on this database, we have studied the 2005 championships. We currently have around 15 million records in this database.

3 Extracting and analyzing the data

After creating the database we have attempted to extract and evaluate data on the techniques used by participating teams, and find the pros and cons of each approach and have used the outcome as guidance for our team.

A few examples are as follows:

3-1. Examining the number of lanes that are usually blocked in a road, and determining the number of lanes that the police agents should clear in these roads. This number has a direct effect on both the agility of the police force, and on the other hand on avoiding traffic jams.

Passable Lanes / Road Lanes	0	1	2	3	∑ Sum
1	**504**	**665**	**-**	**-**	**1169**
2	**44**	**318**	**603**	**-**	**965**
3	**29**	**16**	**28**	**70**	**143**
∑ Sum	**577**	**999**	**631**	**70**	**2277**

Table 1 - Kobe 4 Semifinal

3-2. Evaluating the number of times per cycle that ambulance or fire brigade agents contact with blocked roads. This number is a good indicator of the agent's world model and police force efficiency. We have calculated the average number of block road contacts for each team in various cities.

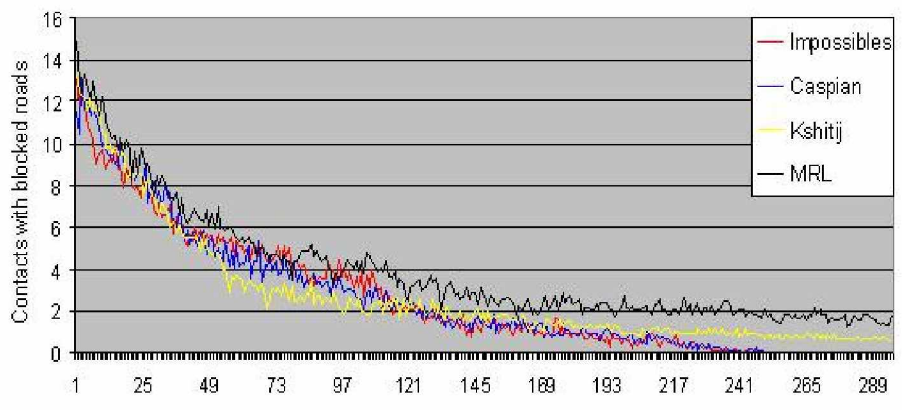

Figure 1

3-3. Evaluating actions performed by team while extinguishing a fire site. The figure shows the average percentage of burned buildings in various cities.

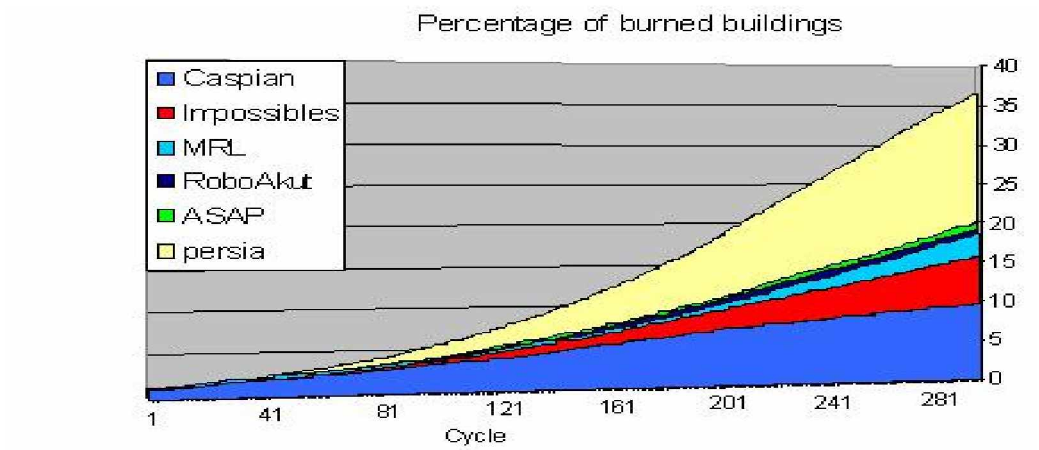

Figure 2

3-4. Evaluating the civilian rescue procedure used by each team, including ambulance collectivity work**.** The figure shows the average civilian death in various cities.

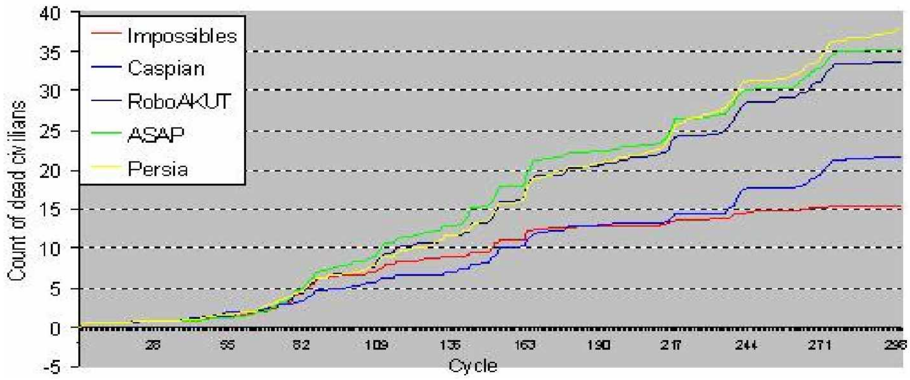

Figure 3

By examining the above charts and the functionality of the *Caspian* and the *Impossibles* teams we can conclude that although both teams are mostly identical but Caspian is better at extinguishing fires and Impossibles at rescuing civilians. A closer examination reveals that Impossibles' ambulances don't act as one group (unlike Caspian) but seemingly use the *contract-net* technique for coordination. *"A contract-net is a classic agent resource allocation technique"* and also *"a coordination strategy that enables agents to allocate tasks and self organize"*.

3-5. Evaluating team performance in various cities.

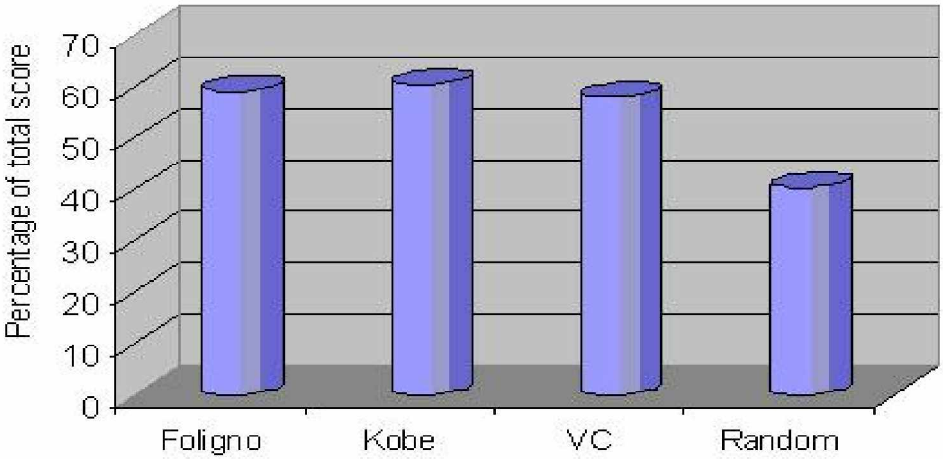

Figure 4 - MRL team performance

This chart shows that the MRL team has a normal score in VC, Foligno and Kobe, indicating that the agents have appropriate previous knowledge about these cities. But the low score in the random city reveals the fact that the MRL agents have not been trained sufficiently for unknown cities.

Although we have just started our analysis and in this course have encountered numerous problems, it is already emerging that all teams participating in robocup rescue simulation will benefit from this analysis and knowledge. We shall give an in depth analysis of the results obtained from our database in a separate paper.

4 Communication Structure

The key criteria influencing the acts of an agent are the presence of a powerful world model. An agent can make better decisions if its world model reflects the real world more accurately.

Our findings state that in order to have a powerful world model a powerful messaging system is needed. Although agents do have restrictions on sending messages, but with aid of compression and standardization, agents can send both sensatory data and some additional data.

Some proposed standardization rules are as follows:

1. Messages should be made up of bit structures.
2. Agents should send the least possible number of messages but receive the most possible.
3. If two or more agents are in the same area, only one of them should send their overlapping information.
4. No information should be sent twice.
5. All objects present in an agent's world model should be mapped to some unique code.
6. The length of object id codes should be reduced dynamically based on the number of objects present in the map.

By following the observing the above rules, sensatory data takes up only 15% of a message, so in addition to sending sensatory data in each message, we can also send information about an agent's acts, location, and targets, and additionally predictions of events happening in the next few cycles. This additional data can help us optimize certain issues. For instance:

a. We have observed that because of the latency present in the messaging system, each agent could predicate events in its surroundings more rapidly than other agents. For instance, a police agent can send information about roads that will be cleared during the next few cycles beforehand, instead of sending this information afterwards. This would influence other agent's path finding process greatly.

b. As another example, consider the fact that the distance between ambulance rescue teams and a specific civilian casualty has a direct effect on each ambulance team's target selection. Thus sending agent location information will be of great benefit.

c. As a final example, agents sense a fire differently based on their distance from the fire site, and sending the exact location and fieriness of the burning building can tremendously help the fire brigades in the first few cycles of the simulation.

Centers should group, condense and prioritize messages received from their agents and resend this information to other centers (and their own platoons) based on this priority. With the help of this messaging system and the use of some common algorithms, target selection and prioritization can be easily enhanced. In summary, we intend to make each agent act as the eye of other agents so that they will perceive their world better.

5 The Police Force

Police functionality in clearing blocked roads plays an important role in optimizing the rescue operation. Clearing the main roads in the first few simulation cycles would boost other platoons' performance.

Police agents should predicate the probability of road blockages and the importance of each road with the help of their built-in knowledge as well as information received from other agents, and start clearing the roads based on this priority.

We attempt to compute the probability of a road being blocked and the priority of clearing that road with the following steps:

a. With the aid of our database, we calculate the static probability of road being blocked as $P_s(r)$, and the static probability of a building being ruined as $P_s(b)$.

b. With the results obtained from the previous step, we can figure a relation between the probability of roads adjacent to a ruined building being blocked (including the number of lanes and length of the blockage.) and the building's properties such as building damage, height and structure. For this, we will define two sets in each city, one with all the buildings and another with all the roads, and define an adjacency relation between the two sets:

$$Roads = \{r_1, r_2, r_3, \ldots\}$$
$$Buildings = \{b_1, b_2, b_3, \ldots\}$$
$$R = \{(b, r) \mid \text{contact}(b, r) > M\}$$
$$\text{contact}() \sim \text{distance between road and building, building height, building area}$$

Because of the uncertainty of the *contact* function, we have produced a *fuzzy dependency coefficient* α between members of *Roads* and *Buildings* sets which improves the precision of the above relation. α is stored in the agents' knowledge base, and during the simulation the probability of a road being blocked is calculated with the following formula:

$$P(b) = \begin{cases} 0 & \text{Building is known to be erect} \\ P_S(\text{b}) & \text{Building situtation is unknown} \\ 1 & \text{Building is known to have collapsed} \end{cases}$$

$$P(r) = \begin{cases} 0 & \text{Road is known to be cleared} \\ \sum(\alpha . P(b_i)) + P_S(r) & \text{Road situtaion is unknown} \\ 1 & \text{Road is known to blocked} \end{cases}$$

The computed probabilities will be stored in each agent's knowledge base so that $P_s(r)$ can be calculated more accurately in the next runs.

c. The final step is prioritizing the roads that need to be unblocked. At start of simulation time, the city is divided into a number of same-size districts, and a police agent is dedicated to each district. (Actually a police agent can work in other districts but it will prioritize its own district's blocked roads by some multiplier.) A static priority R_S is assigned to each road with the Floyd algorithm, and during the simulation this priority is dynamically changed based on the requesting agent and the total number of requests for unblocking each road.

   If a is the number of requests made by the fire agents and b by ambulance agents, the following formula gives the dynamic priority to clear each road:

   $$R_D = a * G(t) + b * F(t)$$

   The police agents will choose their next target based on R_S, R_D, and $P(r)$.

6 Path Finding

A critical task for rescue agents is path finding. Our approach is to create a *fuzzy graph* based on $P(r)$ and with the following weight for unseen edges:

$$W_f = \frac{1}{(1 - P(r))^k} * W_e$$

Using this fuzzy graph and a modified version of Caspian team's routing algorithm we can find the optimized path between an agent and its target.

7 Civilian Exploration

Before rescuing a civilian we need to pinpoint his location. In order to achieve this, all agents will perform the following tasks:

a. If agents *see* a civilian they will notify the ambulance team with the location and attributes of the civilian.

b. If agents *hear* a civilian they will notify all other agents by sending their location along with the civilian's id. Upon receiving this data, every agent will mark all buildings in the 30m radius of the sender. If the same civilian id is heard by another agent, only the overlapping buildings are marked. With the aid of the probability *Ps(b)* (previously calculated), the search priority for each building is determined. As this procedure reiterates, the agents' world model becomes more accurate and the chance of finding survivors increase. This approach is vital if a crowded simulator is implemented

8 The Ambulance Team

Analyzing competition data has led us to part the ambulances into a dynamic number of teams. The advantages of this approach are:

a. Rapid recognition of the disaster space.
b. Starting the rescue operations sooner.
c. Concurrent rescue operations.
d. Lowering the possibility of a traffic jam.

In summary, this approach causes the *rescue-to-move ratio* to increase, and therefore raises our ambulance teams' performance.

In addition, our ambulance agents rescue civilians in an order based by the civilian's *human power, buriedness, fieriness and damage,* which further increases the rescue success.

9 Conclusion And Future Work

We have accomplished in creating a database of techniques and methods used by other teams. In addition to analyzing some of these various techniques and choosing the best ones available, we have also made enhancements to some of the techniques.

In the future, we intend to accomplish these tasks:

i. Improve our data analyzing tools.
ii. Using the available data, extract a pattern for the spread of fire, fire site prioritizing and fire brigade grouping.
iii. Use a slightly modified version of the *contract net* technique for coordination and distribution of ambulances.
iv. Increase the validity and accuracy of each agent's built-in knowledge base.

10 Acknowledgements

We would like to thank Caspian team 2005 for their contributed source code; also Mr. Hossein Heydari (a member of Persia team 2005) for his generous help. Finally we wish to thank Dr Sadeghi and Dr Keyvanpour for their useful insights.

11 References

1. OVUM Report, Intelligent Agents: The New Revolution in Software, 1994.
2. B. Masoumi *et al*, An introduction to distributed artificial intelligence, 2005.
3. M. Huhns, M Singh, Conversational Agents, (IEEE Internet Computing), 1997.
4. T. Morimoto, How to Develop a RoboCup Rescue Agent, 2002.
5. A. Kliner *et al*, ResQ Freiburgb: Team Description and Evaluation, 2004.
6. S. H. Hamraz *et al*, Caspian RoboCup Rescue Simulation Agent, 2005.