RoboCup Rescue 2013
LTI Agent Rescue Team Description

Alan D. Barroso, Felipe de C. Santana, Victor Lassance,
Annibal B. M. da Silva, Luis G. Nardin,
Anarosa A. F. Brandão, Jaime S. Sichman

Laboratório de Técnicas Inteligentes - EP/USP
Av. Prof. Luciano Gualberto, 158 – trav. 3
05508-970 – São Paulo – SP – Brasil
{alan.barroso, fesantana, victor.lassance.silva,
annibal.silva, luis.nardin}@usp.br,
{anarosa.brandao, jaime.sichman}@poli.usp.br

Abstract. In this paper, we describe the strategy implemented by the LTI Agent Rescue team for the RoboCup Rescue Agent Simulation 2013 competition. We mainly focus in some aspects identified as the bottleneck of the team's performance in the 2011 competition: the map exploration and the coordination strategy. In order to improve the map exploration, we use a map partitioning technique to allocate Police Forces to specific regions of the map, while changing the task allocation approach from a hybrid to a distributed one in order to improve the coordination strategy. Moreover, we implemented some techniques to avoid communication channel overload such as data compression and interleaved communication.

1 Introduction

Natural disasters are a source of concern for virtually every country in the world as they usually are difficult to predict and cause huge human and material losses. Altough some disasters may be predicted with a considerable advance, most of them cannot, which leaves a short period of time for any preventive action. Thus, it requires the pre-existence of efficient policies to help handling and responding to such situations, which makes disaster management an important discipline.

According to [1], disaster management is comprised of four phases: mitigation, preparedness, response and recovery. In order to promote research on techniques focused on the response phase of disaster management, the RoboCup Rescue Agent Simulation League [4] was founded in 2000. The competition approaches the problem from a multiagent system perspective, in which teams of heterogeneous agents acts on a simulated post-earthquake scenario. This kind of scenario imposes great challenges to the development of teams of agents since the environment is dynamic, the information and communication are limited and unreliable, and the time to make a decision is limited.

The LTI Agent Rescue team is applying for its second participation on the competition. For RoboCup 2013, we focus in some aspects identified as the bottleneck of the team's performance on RoboCup 2011: the map exploration and the coordination strategy.

Last year, the map exploration was performed based on the agents' random walking, however it caused a concentration of the agents in specific regions of the map leaving many parts of the map unexplored, victims undetected and fires unnoticed. Besides, the channel communication characteristics were not explored since the team used a strategy that concentrated all the communication and information on the *center* agents.

Therefore, some changes and improvements have being made for the RoboCup 2013 competition. In order to mitigate the map exploration problem, we developed a method of partitioning, used by the Police Forces to divide the map into sectors and reduce the patrolling area of each one of them (see Section 3). Regarding the coordination approach problem, we changed how the agents exchange information among themselves in order to allow them to communicate directly without requiring the *center* intermediation. Thus, it changed the task allocation approach from a hybrid to a completely distributed one. Moreover, some techniques to avoid channel overload, such as the Adaptive Huffman encoding [3,6] and interleaved information exchange were implemented (see Section 4).

The rest of the document is structured as follows. Section 2 describes the distributed strategy employed in the development of each kind of agent, as well as their specific behaviors. Next, the map partitioning technique developed to subdivide the map for Police Forces' surveillance are described in Section 3. In Section 4, we present the communication protocol used to enable the information exchange among the agents. Finally, we present some conclusions in Section 5.

2 Strategy

In RoboCup 2011, we implemented a strategy that followed a hybrid task allocation approach [5] to control and coordinate agents in the RoboCup Rescue Simulator (RCRS) platform. Such approach was inspired on the *Partial Global Planning* (PGP) approach [2] in which the agents exchange information in order to reach common conclusions about the problem to be solved. It is considered *partial* because agents have limited information of the environment available in order to make the best decision; on the other hand, it is *global* because the agents can exchange and obtain relevant missing information from each other when necessary.

The proposed hybrid approach intended to exploit the advantages of the *centralized* and *distributed* approaches, considering and handling the existence of both local and global information, making the *platoon* agents not completely dependent on communication and on the *center* agents. Despite being inspired on PGP approach, the proposed approach had four key differences to the PGP:

- The agents exchanged tasks instead of plans;

- There was no direct communication among *platoon* agents, which required the performance of such tasks by the *center* agents;
- The global information was stored in the *center* agents;
- The global information was only requested and used when the agent's had no local tasks to act upon.

Based on the experience obtained in RoboCup 2011, we have decided to adopt a new strategy for RoboCup 2013, in which the information storage and decision making are completely local, and the agents exchange their sensed information directly among themselves. Therefore, the new strategy follows a distributed approach.

The main reason for such change is that, since the release of the new RCRS platform, in 2010, the *platoon* agents can communicate through channels directly among themselves, canceling any imperative need of the *center* agents to exchange information. In fact, using center-intermediation would generate a delay of one cycle in the process of information exchange, worsening the situation instead of bringing any real advantage in the case of a distributed approach. Hence, the *center* agents play no role in this new strategy.

Moreover, using a distributed approach maintains some advantages of the previous one, such as the nonexistence of a single point of failure and the elimination of a possible bottleneck. Besides, task allocation can still be performed even if the agents have little or no interaction with other agents, which is interesting for an environment like the RCRS, where the communication is subject to failure or not available at all.

The disadvantage of this approach is that the communication channel may be overloaded, for example, in the case when all the agents are connected to the same channel. Thus, in order to overcome this problem, a set of techniques such as interleaved information exchange, data compression, and transmission of only required data is applied and detailed in Section 4.

In order to implement the new strategy, each agent maintains two databases, one of *tasks* and another of *world information*. In our context, *world information* are data about the state of the environment's objects; *tasks* are world information upon which the agents may act, such as buildings on fire for Fire Brigades, blocked roads for Police Forces, and victims for Ambulance Teams. The task database will also indicate which agents are currently acting on each task.

Using these information, the following general agent behavior was designed:

1. The agent percepts the environment and receives messages from other agents;
2. The agent updates its world information database;
3. The agent generates a list of new identified tasks from the world information database;
4. The agent adds the new identified tasks to its tasks database; it also updates the list of the agents that are acting on each task;
5. The agent calculates the payoff of each task, taking into account the possibility of cooperation with other agents, and chooses the task with the highest payoff for act upon; if the agent has no task it can act upon, it moves randomly on the map;

6. The agent sends information to the other agents, comprised of:
   - the world information whose state has changed;
   - the task it has chosen to work on (if any);

Next, the specific behavior of each kind of agent is described:

2.1 Police Force

The Police Forces have an important role in the new strategy developed in order to overcome the problem of map exploration faced in RoboCup 2011. Each one is given the task of performing the surveillance of a specific partition of the map. Other teams such as MRL, HfutEngineRescue, IAMRescue, and RMAS_ArtSapience have already reported the use of mapping partitioning and we think that it will help us too. Hence, our intents in partitioning the map were (i) to be able to better explore the map and identify faster possible tasks to act upon, and (ii) reduce the number of entities that each Police Force has to check fastening the exploration process.

Besides, since Fire Brigades need to get to refuges to refill their water tank and Ambulance Teams need to have fast ways to reach the refuge when transporting civilians, clearing paths to refuges is considered crucial in our strategy. Therefore, in order to make refuges accessible to the other kinds of agents, the first task of the Police Forces are to move to the closest refuge located in its sector, if there is any, clearing the blockades in the path.

Once it reaches the refuge, the Police Force starts patrolling its sector by visiting each road and building contained in it. During such activity, it may find new blockades or receive blockades information from other agents, requiring it to choose a blockade to act upon: for this, it first rules out the blockades that have other agent acting upon, since there is no advantage of more than one Police Force acting in the same blockade, and calculates the payoff of the remaining blockades. The payoff calculus takes into account the blockade repair cost and its distance, giving higher payoffs to blockades with small repair cost and that are closer to the agent.

In order to clear more roads and cover a broader area, the Police Force clears the blockades in a way it can transpose it. In some cases, this does not require the complete clearing of the blockade, which saves time improving the overall Police Force results.

2.2 Ambulance Team

The aim of the Ambulance Team is to rescue the largest number of civilians trapped inside buildings, called victims. In order to succeed in its goal, if the Ambulance Team has no task assigned to act upon, it checks its tasks database and calculates the payoff value related to each task in it. The payoff calculus is the difference between the expected time to rescue the victim and the time of life the victim still have. Then, the Ambulance Team chooses the task with the highest value to act upon. Once a task is selected, the Ambulance Team moves to the victims location, performs the rescue procedure, loads the victim, transports it to the closest unblocked refuge, and unloads the victim.

Usually, the Ambulance Team sticks with the task until the rescue is complete (victim in the refuge), however, some situations may cause the Ambulance Team to give up on the victim and look for other task to act upon, for example if the Ambulance Team cannot reach the victim because all the possible paths are blocked or if the Ambulance Team identifies that the building where the victim is trapped in is on fire.

2.3 Fire Brigade

The Fire Brigade has the most difficult coordination of the three kinds of agents, since as opposed to victims and blockades, fires are highly dynamic, appearing randomly and spreading during the course of the simulation; they also may reemerge after being extinguished.

The goal of the Fire Brigades is to find fires quickly and extinguish them, preventing it from spreading. Map exploration is important for that, so Police Forces play an important role in the fire fighting activity. In case the Fire Brigade needs to choose which burning building it will try to extinguish the fire, it first calculates the payoff (Eq. 1) of all the fire it knows and chooses the one with the highest payoff value

$$\left(\frac{1}{buildingArea} * fieryness\right) + \frac{1}{\sqrt{distance}} + numberOfUnburningNeighbors \tag{1}$$

Using Eq. 1, it is expected the Fire Brigade to choose the building with the smallest area, highest fire intensity, closest to it and with the largest number of neighbors buildings that are not burning. Therefore, it should choose a small building that is closer to it and it is located at the border of a fire cluster.

3 Map Partitioning

The solution we propose to the problem of map exploration is the partitioning of the map into sectors, with each Police Force being assigned to patrol one of the sectors. Police Forces are the obvious choice for explore the map because they have the ability to remove blockades granting access to any place in the map. Additionally, separation of Police Forces in sectors ceases their encounters reducing the number of idle agents.

The maps of the RCRS are graphs, such as the neighbors entities shown in the viewer of the simulator. Then, the goal becomes obtaining sectors in which the set of entities (buildings and roads) of each one forms a connected subgraph. Each entity will belong to an unique sector, so that Police Forces that patrol different sectors do not visit the same location.

The algorithm divides the map into n sectors, where n is the number of Police Forces in the map. The number of sectors was defined arbitrarily and may be changed according to the map size and configurations.

The proposed map partitioning method divides the map into sectors comprised of:

- An index that identifies it;
- The two sets of X and Y coordinates that delimit its area;
- The set of entities it contains forming a single connected subgraph.

The pseudo-algorithm that performs the map partitioning is presented in Pseudo-Algorithm 1.

Pseudo-Algorithm 1 Map partitioning

```
Require: $n \Leftarrow numberOfSectors$

  $mapLength \Leftarrow$ Length of the map
  $mapHeight \Leftarrow$ Height of the map
  $term[2] = Factorization(n)$ {Factorize $n$ into 2 numbers}
  if $mapLength > mapHeight$ then
    $length\ =\ mapLength\ /\ term[2]$
    $height\ =\ mapHeight\ /\ term[1]$
  else
    $length\ =\ mapLength\ /\ term[1]$
    $height\ =\ mapHeight\ /\ term[2]$
  end if
  $sectors \Leftarrow createSectors(length, height)$
  $sectors \Leftarrow allocateEntities(sectors)$
  return $sectors$
```

First, it obtains the length ($mapLength$) and height ($mapHeight$) dimensions of the map in pixels. Then, it factorizes (Pseudo-Algorithm 2) the number of sector (n) passed as parameter into two terms, where those terms are the closest factors of n, e.g. if $n = 20$ then $term[1] = 4$ and $term[2] = 5$.

Pseudo-Algorithm 2 Factorization

```
Require: $n \Leftarrow numberOfSectors$

  $difference \Leftarrow \infty$
  for $i = 1$ to $\sqrt{n}$ do
    if $(n \bmod i) = 0$ then
      if $|n - (n/i)| < difference$ then
        $result \Leftarrow (i, n/i)$
        $difference \Leftarrow |n - (n/i)|$
      end if
    end if
  end for
  return $result$
```

After this, the largest dimension is divided by the largest term and the smallest dimension is divided by the smallest term. Based on those values, the sectors are created and their set of X and Y coordinates are adjusted.

In the sequence, for each sector, the entities geographically contained in it are identified and grouped into connected subgraphs (Pseudo-Algorithm 3). Note that there may be more than one connected subgraph in each sector. Thus, the largest connected subgraph of each sector is chosen as the main connected subgraph of that sector and the remaining subgraphs are relinked to one of the main subgraphs. The relink is performed between the subgraph entities closest to the largest main subgraph that may form a new connected subgraph.

Pseudo-Algorithm 3 Allocates Entities

```
Require: $sectors \Leftarrow$ Created Sectors

  Identify the entities within each sector
  Identify the largest connected subgraph within each sector
  Allocate the remaining subgraphs of each sector to a sector's main graph
```

4 Communication

Communication is usually a limited resource in a disaster situation, nevertheless such resource is extremely useful for coordination purposes and sharing of information. Those characteristics are very well reflected in the RCRS platform with the limited bandwidth, probability of drop a message, and probability of channel failure.

In order to overcome those limitations, we implemented several techniques that allow a better use of the resources available and circumvent any problem it may be experienced during the simulation.

The first improvement incorporated is the use of Adaptive Huffman coding [6], which allows the data compression and transmission of more data over the same amount of bandwidth. The Adaptive Huffman coding is an adaptive technique based on Huffman coding [3] used for data compression in real time since permits building the code as the symbols are being transmitted, having no initial knowledge of source distribution, that allows one-pass encoding and adaptation to changing conditions in data.

Since the use of any adaptive coding technique makes the data transmission much more sensitive to transmission errors, and it may happen in RCRS, we added a 1 byte Cyclic Redundancy Check (CRC) field in every message transmitted. The CRC value is the lowest byte of the sum of all the message bytes.

Another improvement is the interleaved used of the channel by different agents depending on their IDs, where agents with even IDs transmit on even cycle, while the others transmit on odd cycles. Despite of the benefit of not overload the channel, such technique may cause a delay on information transmission, because of that we apply such technique only when the channel bandwidth is considered insufficient for the data transmission, which is still in the analysis phase.

5 Conclusion

This paper presented a description of the strategy implemented in the LTI Agent Rescue team for the RoboCup Rescue 2013. The new strategy has some differences from the one implemented for RoboCup 2011 mainly related to the map exploration, the coordination approach, and the communication efficiency. The map exploration proposed is based in a map partitioning technique used to avoid that specific regions of the map stays unexplored. The coordination approach was changed from a hybrid to a distributed one, making the agents completely autonomous on their decisions and actions. The increase on the communication efficiency was obtained by the application of several techniques to avoid channel overload and to allow a better use of the resources available.

Therefore, we expect to have an increase in the teams performance by applying those changes. However, we are still implementing some of those changes and we could not perform comparison analysis with RoboCup 2011 performance. Nevertheless, we surely will have all those changes in place for RoboCup 2013.

Acknowledgments

Annibal B. M. da Silva is supported by AEP/Poli/USP. Jaime S. Sichman is partially supported by CNPq/Brazil.

References

1. W. Blanchard, L. G. Canton, C. L. Cwiak, K. C. Goss, D. A. McEntire, L. Newsome, M. D. Selvers, E. A. Sorchik, K. Stenson, J. E. Turner, III, W. L. Waugh, Jr., and D. West. Principles of emergency management supplement. Technical report, International Association of Emergency Managers, 2007.
2. E. H. Durfee and V. R. Lesser. Partial global planning: A coordination framework for distributed hypothesis formation. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(5):1167–1183, sep/oct 1991.
3. D. A. Huffman. A method for the construction of Minimum-Redundancy codes. *Proceedings of the IRE*, 40(9):1098–1101, 1952.
4. H. Kitano, S. Tadokoro, I. Noda, H. Matsubara, T. Takahashi, A. Shinjou, and S. Shimada. Robocup rescue: Search and rescue in large-scale disasters as a domain for autonomous agents research. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, volume 6, pages 739–743, Tokyo, Japan, October 1999. IEEE.
5. A. H. Pereira, L. G. Nardin, A. A. F. Brandão, and J. S. Sichman. LTI Agent Rescue team: A BDI-based approach for Robocup Rescue. In *Proceedings of the RoboCup 2011*, Istanbul, Turkey, 2011.
6. J. S. Vitter. Design and analysis of dynamic huffman codes. *J. ACM*, 34:825–845, 1987.