# Rescue Agents Practical Handout 2

| COURSEWORK DETAILS | | | |
|---|---|---|---|
| **C/W NUMBER** | 2nd | **CONTRIBUTION** | 15 % of the module final mark |
| **C/W TITLE** | Cooperative Rescue Agent System | | |
| **C/W TYPE** | Individual Practical | | |

**LEARNING OUTCOMES**

Upon completion of this piece of coursework, a student will be able to:

- understand in depth the issues and the difficulties involved when building a multi-agent system, such as agent communication languages, interactions protocols, language used etc.
- to use an existing library to construct FIPA ACL like messages and implement an interaction protocol
- propose a suitable agent architecture to perform a problem solving task
- build a simple prototype of a multi agent system on a multi-agent simulation platform (NetLogo)
- evaluate the design choices made based on simulation results.

| ASSESSMENT CRITERIA | |
|---|---|
| ▪ Correctness, originality and justification of the proposed agent architectures | 25% |
| ▪ Correctness and justification of the cooperation protocols proposed | 30% |
| ▪ Implementation and code documentation | 30% |
| ▪ Analysis and presentation of experimental results | 5% |
| ▪ Presentation of the report (clarity, structure) | 10% |

## DETAILED DESCRIPTION

**Rescue Agents**

Rescuing civilians in disaster situations is a difficult problem that attracted a lot of attention in the recent years. Intelligent autonomous robotic agents can be a valuable tool in such situations, since they can help to locate and rescue victims of disasters quite efficiently. The rescue procedure usually is as follows. A rescue unit locates the civilian (victim) in need. Since in most of the cases, the victim requires immediate attention and help, the *rescue unit* provides oxygen, water or small rations of food to the victims, so that they can be sustained in life until transportation arrives. Additionally the rescue unit informs the *base* about the location of the (temporarily) rescued civilian. Upon the reception of such a message, the base informs an appropriate *ambulance* that will pick up the rescued civilian. Ambulances in this case are helicopters that can fly straight to the location of the civilian to be picked up, without encountering any obstacles.
Thus, such a rescue team is equipped with three types of agents:

- rescue-units that are autonomous vehicles that can move around the disaster area, locate (detect) any civilians in danger (victims) and temporarily rescue them.
- ambulances agents that are autonomous transportation units able to collect civilians rescued.
- a base agent, which is an agent installed in a central location coordinating the rescue activities.

An example disaster area with all agents operating is shown in the figure below (Figure 1). Note that agents listed above have different abilities, as those are described in the text that follows.

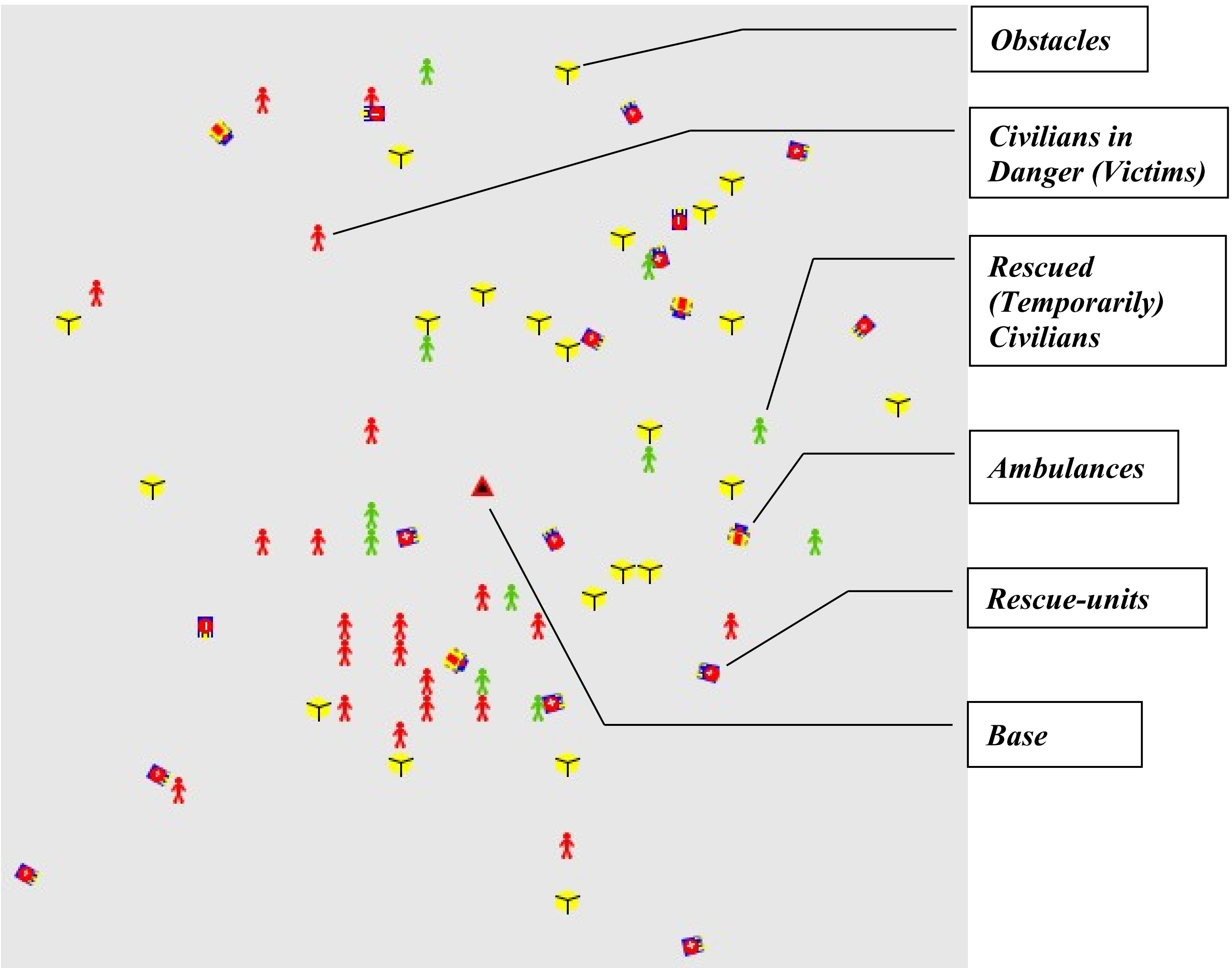


**Figure 1: Agents rescuing civilians in a disaster scenario**

Rescue units have a very limited set of sensors. The latter includes:

- a sensor for detecting civilians in danger, that operates at a very close range (*detect-civilian NetLogo reporter*)
- a sensor that notifies the agent that another agent or an obstacle is near by (*detect-obstacle NetLogo reporter*). Obviously this sensor is used to avoid collisions.

Rescue-unit agents:

- can move on the floor in order search for victims (civilians) (*move-randomly NetLogo procedure*)
- can avoid obstacles (*avoid-obstacle NetLogo procedure*)
- are able to provide immediate attention to victims, (*rescue-civilian NetLogo procedure*), and
- are able to send a simple message to the base reporting the position of a civilian to be collected by an ambulance (*inform-base NetLogo procedure*). This is an extremely limited form of communication. The rescue unit can only broadcast that its position to the base without being able to engage in any form of dialogue with the base or any other unit.

Rescue units are implemented as reactive agents in the NetLogo model provided.

On the other hand **ambulances** are more advanced than rescue units in terms of communication abilities, since they can exchange FIPA ACL messages with other agents. Ambulances are equipped with a GPS system that allows them to:

- report their location,
- compute their distance for a point,
- direct themselves towards a specific location in the disaster area given the corresponding coordinates (*move-towards-dest NetLogo procedure*) and
- pick up rescued civilians spotted by rescue-units (*pick-up-victim Netlogo procedure*).

It should be noted that ambulances are assumed to have a limited capacity of victims that they can carry, set by the corresponding *maximum_load* slider on the environment. When the load (rescued civilians) the agent is carrying exceeds the maximum load, the agent has to return to the base in order to leave the civilians in a safe location and continue its task.
Ambulance agents are modelled as hybrid agents were

- one layer avoids obstacles and takes the opportunity of rescuing civilians and goes back to the base whenever it carries more civilians than its capacity, and
- one proactive layer handles message passing and the other operations for rescuing civilians.

Finally the **base agent** is a stationary communicating agent that can exchange FIPA ACL messages with other agents.

All agents operate in the same environment (disaster area). The resulting system's efficiency is measured in terms of total distance travelled by the ambulance agents and time required to complete the task. Obviously a well designed system should minimise both measures.

Part of the agent system described above has been implemented as a model in the NetLogo platform, (as shown in Figure 1) and the code of the model is provided through the web page of the course. The model includes procedures and reporters that implementing sensors, communication facilities, appropriate code for building proactive agents in NetLogo and facilities for monitoring the time and cost (distance travelled by ambulances) of the performed experiment, as well as running the experiment with a varying number of agents. The code provided also positions all agents on the grid.

1) Briefly state why the hybrid architecture is appropriate for the ambulance agents, comparing it to the BDI or logic based architectures.

2) In the implementation provided, the information about the location of a victim is send by the rescue units to the base, which in turn forwards this information to all agents requesting the pick-up of the victim found. This is a rather naive approach since:
   - all agents rush to the same location at once
   - once the victim is rescued by one agent other agents are not aware of this fact, and thus still try to move to the specific rescue point.

   Propose and implement simple improvements to the above scheme that would allow the system to perform better. Your improvements should contain only minor changes to the architecture provided. Run a few experiments demonstrating the improvements compared to the previous version.

3) Having all communication redirected through the base unit causes a lot of problems and creates a single point of failure. Thus, the rescue unit agent architecture should be **extended with FIPA ACL communication facilities** in order to allow better cooperation between rescue units and ambulances. Propose and briefly describe a suitable cooperation protocol between rescue units and ambulances that could be employed to efficiently rescue civilians and justify your answer. Describe in detail all the FIPA ACL messages exchanged by the agents engaged in the interaction protocol that you proposed.
   Note: In this rather limited environment, you may assume that agents share a common ontology and language, so there is no need to specify them explicitly.

4) Taking into consideration all the above, provide detailed descriptions of the new, extended rescue units and ambulance agent architectures of the so that they can cooperate under the protocol you described in Q.3. In both cases your descriptions should be as detailed as possible, including the beliefs, intentions of the agent, etc.

5) Implement the ambulance and rescue unit agents in NetLogo and run a few experiments to demonstrate the correctness of your design choices.

| **SUBMISSION** |
|---|
| Students are expected to submit:<br>▪ A written report that contains<br>• Appropriate answers to all the questions that appear above, properly justified.<br>• Descriptions of the experiments carried out.<br>• A printout of the code **they have implemented**, **highlighting in bold characters the changes implemented to the original code given. STUDENTS SHOULD NOT SUBMIT A PRINTOUT OF THE WHOLE NETLOGO MODEL, BUT JUST INCLUDE ONLY THOSE PARTS TO WHICH CHANGES WERE MADE.**<br>• Students are expected to submit **two** NetLogo models and the corresponding printouts: the first model will implement the answer for Q2 and the second the answer for Q5.<br>• References<br>▪ A floppy disc containing the documented NetLogo code. In all cases the results obtained by the submitted code should match those appearing in the report.<br><br>*The submitted code should also be sent by e-mail to the e-mail address of the instructor. Please include the code in a zip file (compressed) attached to the e-mail.* |