RoboCup 2005 Rescue Simulation League Rules

May 25, 2005

Simulation Environment

1. **Software environments:**
   (a) **Simulation Environment:** A Linux package will be used. (Now we are going to prepare FedoraCore 3)
   (b) **Simulation Package:** At the competition a package of the latest version will be used. This package are maintained by Cameron Skinner and Arash Rahimi.
   Anyone can download the package from **http://sourceforge.net/projects/roborescue/**
      i. Aftershocks will be implemented. These may cause buildings to collapse, possibly trapping civilians. Agents will be immune to damage/buriedness from aftershocks.
      ii. The message protocol will be extended to include a "*length*" field.
      iii. Communication and perception code will be moved out of the kernel into separate modules.
      iv. Message loss will be implemented in a controlled fashion (c.f. 2004 competition where messages were lost due to UDP packets disappearing). So, we will use TCP communication to control message loss.
      v. The kernel will not send the IDs of all objects when an agent connects. All fixed objects (roads, nodes, buildings etc.) will be sent, but only visible moving objects (i.e. civilians) will be sent. When an agent observes a new civilian during the simulation the appropriate information will be sent.
   (c) **Traffic simulator:** In accordance with the community voting for Robocup '05 the traffic simulator was changed to allow only one fire brigade per building (but which may still extinguish other buildings from inside).
   (d) **Firesimulator:** The ResQ Firesimulator will be used at 2005 competition. In particular this means that buildings may reignite after having been extinguished if they are not completely burnt down, and that buildings can be preemptively watered. The Firesimulator is maintained by Timo Müssle. Anyone can download it from **http://kaspar.informatik.uni-freiburg.de/~nuessle/**
   To model the new possibilities for states of building the "fieryness" property was given the following new interpretation both in the fire simulator and the 2D and 3D viewers:

0	not burnt, no water damage
1	burning, slightly damaged
2	burning, more damaged
3	burning, severely damaged
4	not burnt, but watered-damaged
5	extinguished, slightly damaged
6	extinguished, more damaged
7	extinguished, severely damaged
8	completely burnt down

   Note that that the previously unused level 4 is used for describing buildings that have not yet burnt, but have been watered.

2. **Maps:** Five maps can be used. The maps to be used are determined by TC.
   (a) Maps known to all teams
      i. center of Foligno (Italy)
      ii. Kobe (1/10, 1/4)(Japan)
      iii. Virtual City
   (b) Random map prepared at the competition site by the Technical Committee.

3. **Agents:** Agents shall be implemented by Teams.
   (a) **Numbers:** The numbers of agents and ignition points specified in gisinit.txt are in the following range. The numbers are open to contestants before games.

Entity	Min.	Max.
Fire Brigade	0	15
Police Force	0	15
Ambulance	0	8
Civilian	70	90
Fire Brigade Center	0	1
Police Force Center	0	1
Ambulance Center	0	1
Refuge	0	5
Ignition points	2	8*

∗ The maximum number of initial burning buildings will increase to **30**. However these buildings will be grouped into no more than **8** different areas - i.e. there will be up to 8 initial fires, but they might begin with several burning buildings each.

(b) Agents (and civilians) will suffer damage if they are near a burning or collapsing building, not just if they are inside the building.

(c) `AK_MOVE` will be extended to include an optional "*positionExtra*" specification.

(d) Perception will be based on line-of-sight calculations rather than a simple radius of vision/hearing. This will calculate perception based on what the agents can actually see.

(e) Agents will receive notification of other agent's actions that they can observe. For example, a police force agent might see a fire brigade extinguishing a building.

(f) The space inside each building will be limited. The exact limit has not yet been decided. So the number of agents which can enter a building is limited.

4. **Civilian Agents:**

(a) Shinoda and Noda's civilian with behavior rule-sets shall be used.

(b) Civilian agents shall be automatically generated using Nishimura and Takeuchisimplementation.

(c) As a mention above, civilians (and agents) will suffer damage if they are near a burning or collapsing building, not just if they are inside the building.

5. **Messages:**

(a) **Communication conditions:** Messages used by rescue agents for communication within a team shall satisfy the following conditions. In each cycle,

i. For platoon agents, maximum 4 messages sent and 4 messages received,

ii. For center agents, maximum 2*n messages sent and 2*n messages received, where n is the number of platoon agents of the same kind as center. For example, there are 10 fire brigades so a fire station can send and receive at most 20 messages per cycle.

iii. Each agent must completely ignore and discard the messages it decides not to hear.

(b) **Format:** A message shall consist of three parts: `selfID senderID data_part`. There is no other restriction.

(c) **Size:** The size of the `data_part` shall not exceed 256 bytes. `selfID` and `senderID` are 32-bit ingegers set by the simulation kernel.

(d) **Message Reading Procedure:** Standard `message_read` procedure for platoon agents:

i. Store all messages into two buffers, `buffer_1` and `buffer_2 :` `buffer_1` for `selfID` and `senderID` and `buffer_2` for data_part.

ii. Select 4 messages using ONLY `buffer_1`.

iii. Read the corresponding data-parts from **buffer_2**.
You can see an example of recommended implementation in Agent.java of YabAPI code.

```
KaHear hear = (KaHear) data;
RealObject sender = world.get(hear.senderId);
if (sender != self()
&& m_numHearing ++ < hearingLimit())
hear(sender, hear.message);
```

(e) **Additional hearning:**

i. **KA_HEAR_TELL** and **KA_HEAR_SAY** are added to the RoboCup Rescue protocol.

`KA_HEAR`**:** Used in 2003

`KA_HEAR_TELL`**:** `KA_HEAR` sent to kernel by `AK_HTELL` command

`KA_HEAR_SAY` : `KA_HEAR` sent to kernel by `AK_SAY` command
The kernel sends two packets as follows, when kernel receives a communication command (`AK_TELL` or `AK_SAY`):

- When kernel receives an `AK_TELL` communication command, then kernel sends two packets, `KA_HEAR` and `KA_HEAR_TELL`, which have the same message body.
- When kernel receives an `AK_SAY` communication command, then kernel sends two packets, `KA_HEAR` and `KA_HEAR_SAY`, which have the same message body.

ii. Agent developers can also use the old `KA_HEAR` header. The agents developed according to 2003 rules can run without any changes, because the kernel sends two packets as mentioned in (a) above. Agent developers can use either or both of the present `KA_HEAR` header and the headers added in this proposal, `KA_HEAR_TELL` and `KA_HEAR_SAY`. Agent developers can choose command to receive a communication message.

iii. The number of messages sent and received is not changed. Agents should receive a message via only one of `KA_HEAR`, `KA_HEAR_TELL` and `KA_HEAR_SAY`, because these three commands have the same body.

6. **Parameters uses in games:**

(a) parameters in config.txt[kernel]:
Default parameters are used.

```
Default parameters are used.
################################################
# New options for the 2004 competition
#These will be available if they are implemented by Jan 31, 2004
# The sum of the amount of water in an agent's nozzles in a cycle is equal to the limit for
a single nozzle.
#There is no limit on the number of nozzles.
# not new, but the subject of the discussion
say_max_bytes : 256
################################################
# options for the 2003 competition.
# An agent can discover a fire within a distance D[m] roughly
# proportional to its fieryness, where
# D = K * (cycles from the start of burnup)
# If fire_cognition_spredding_speed >= 0, K is given by
# fire_cognition_spredding_speed. Otherwise, K is infinite.
fire_cognition_spredding_speed : 10000
# If simulate_tank_quantity is true, tank quantity properties are
# decreased by extinguishments. If tank is empty, extinguishments
# should be failed.
simulate_tank_quantity : true
# Tank quantity properties are increased by tank_quantity_recovery_on_refuge
# if the agent is in a refuge.
tank_quantity_recovery_on_refuge : 1000
# same as max_extinguish_power
# Gives the maximum quantity of a tank.
```

```
tank_quantity_maximum : 15000
# not new, but the subject of the discussion
# 2003 limit
# say_max_bytes : 80
################################################
# new options for the 2002 competition.
ignore_nozzle_position : true
area_per_repair_cost : 20000000
round_down_quantity : false
accept_multiple_nozzles : true
RoboCup 2004 Rescue Simulation League Rules V1.01 4 January 20, 2004
near_agents_rescuable : true
steps_far_fire_invisible : 5
steps_agents_freezed : 3
notify_initial_position : false
notify_position_history : false
miscsimulator_supports_load : false
notify_only_fire_for_far_buildings : true
notify_unchangeable_informaion : true
# true for compatibility
 :
```

(b) Other parameters:
   - simulation time: 300 steps,
   - range of eyesight is 10m,
   - range of voice is 30m,
   - power of earthquake: shindopolydata.dat, galpolydata.dat

Competition:

1. There shall be two competitions: Agent competition and infrastructure competition:

   (a) **Agent Competition:**

   i. The Technical Committee shall design the disaster situations to be used during the competitions. Disaster situations chosen for a competition shall be able to test efficiency of agents' task in various conditions.

   ii. Simulations held in each day of competitions shall be divided into Sessions. A 'Session' is a group of Simulations held on a specific disaster situation in a round.

   iii. Teams shall submit their code/binary before the start of session.

   iv. Teams shall NOT know the disaster situation (map, gisini, *polydata) of the session before it starts.

   v. Teams can either modify their code or submit new code between sessions.

   vi. Teams shall have all kinds of agents and must connect the correct number of agents as specified for the current session.

   vii. *Precomputation* defined as an agents loading and using MAP-SPECIFIC data which has been created and processed by another program and saved in a file is allowed under the following conditions:

   A. The data should have been generated by a computer program with no human interaction.

   B. Information for all known maps should have been generated by a single computer program.

   C. The computer program used for computing data for known maps should work properly if it is given a new map.

   D. An agent should choose the data file to be used itself.

   E. Agents should be able to work if no precomputed data is present for a map.

   F. The source of the precomputation program shall be open after the competition.

viii. **Number of simulations in each round:** In each round there shall be more than one simulation on the same map. The number shall be determined and announced on competition site by TC. The score for that round shall be determined by finding the average of the points for the runs.

ix. **Evaluation Rule:** We will use a new scoring rule of German Open 2005. The scoring was intended to take the new aspects of the fire simulation into account.
The higher V value for a map, the better rescue operation.
V=(P + S/Sint) * sqrt(B/Bint)
P: number of living agents,
Sint: total HP of all agents at start,
S: remaing HP of all agents,
Bint: total area at start,
B: area of houses that are undestroyed,
the following rule was used to compute the value B:

0	no penalty
1,5	1/3 of area counted as destroyed
4	water damage, also 1/3 area counted as destroyed
2,6	2/3 of area considered as destroyed
3,7,8	whole building considered as destroyed

x. **Valid games:** Some simulators may fail during a game, although they are debugged. In that case, only ONE more simulation is permitted. When the same problem occurs at the simulation, the simulation shall continue to the end. The point at the game is treated on the principle that the continuation should not be in the team's favor. Namely,

A. in a case of the traffic simulator fails, the continuation is against the team's score. So, the game is valid and the score created is an effective one.

B. in cases of the other simulators (including kernel) fail, the continuation is for the team's score. For example, the fire simulator stops then there are no more fires. In that case, the team gets 0 points for that simulation in the preliminary games, and loses the match in the final games.

xi. **Others:** All the conditions for a particular disaster situation, including random numbers, etc shall be identical for all the teams.

(b) **Infrastructure competition:**
Improvements in simulation components, such as disaster simulators, GIS, kernel, viewers, civilian agents and so on, are essential in rescue simulations. This competition tests the performance of these components. The awarded team will be requested to provide the component for next year's competition. Teams will present their tools in front of all teams. Ranking will be decided with votes from TC members and teams.

2. **Open source policy:**

(a) Source code files must be open immediately after games when other teams request to guarantee fair play and to encourage community activity after competition.

(b) Log files and related parameter files will be open.