

# RoboCup 2004 Rescue Simulation League Rules V1.01

## January 20, 2004

### Simulation Environment

#### 1) Software environments:

a) **Simulation Environment:** A Linux package will be used. The package will be downloaded [here](#). At the competition a package of the latest version will be used.

- i) Aftershocks can occur during the simulation. For each trial during the competition the same “random” sequence of aftershocks will apply to each of the contestants. (will be effective if implemented by Jan 31, 2004).
- ii) Kernel should generate random IDs for units in each simulation. This means that the ID generated for a building in each simulation should be different from the next one.

b) **Viewer:** [Morimoto viewer](#) will be used at 2004 competition. It also means that scores displayed by Morimoto viewer are official ones.

#### c) **Open Source Project (Will be effective if implemented by January 31, 2004):**

i) The following simulation environment components are open source:

- kernel,
- gis,
- traffic simulator,
- viewer,
- blockadessimulator,
- collapsesimulator,
- civilian and
- firesimulator.

ii) The open source project is administered by members of the Technical Committee.

iii) Anyone can download the source code and make changes, but only the administrators can commit those changes to the source repository. Ordinary users can submit patches which can be committed by an administrator.

iv) The changes to the environment are fixed 4 months prior to the competition.

2) **Maps:** Four 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 (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
Refuges	0	5
Ignition points	2	8

b) Agents can specify exactly to which part of a road they want to move, i.e., agents can set *PositionExtra* property for a target road.

- c) **Civilian Agents:**
- i) [Shinoda and Noda's civilian](#) with behavior rule-sets shall be used.
  - ii) Civilian agents shall be automatically generated using Nishimura and Takeuchi's implementation.
- d) **Messages:**
- i) **Communication conditions:** Messages used by rescue agents for communication within a team shall satisfy the following conditions. In each cycle,
    - (1) for platoon agents, maximum 4 messages sent and 4 messages received,
    - (2) 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.
    - (3) Each agent must completely ignore and discard the messages it decides not to hear.
  - ii) **Format:** A message shall consist of three parts: `selfID senderID data_part`. There is no other restriction.
  - iii) **Size:** The size of the `data_part` shall not exceed 256 bytes. `selfID` and `senderID` are 32-bit integers set by the simulation kernel.
  - iv) **Message Reading Procedure:** Standard `message_read` procedure for platoon agents:
    - (1) Store all messages into two buffers, `buffer_1` and `buffer_2` : `buffer_1` for `selfID` and `senderID` and `buffer_2` for `data_part`.
    - (2) Select 4 messages using ONLY `buffer_1`.
    - (3) Read the corresponding data-parts from `buffer_2`.

You can see an example of recommended implementation in Agent.java of [YabAPI code](#).

```

66 KaHear hear = (KaHear) data;
67 RealObject sender = world.get(hear.senderId);
68 if (sender != self())
69   && m_numHearing ++ < hearingLimit())
70   hear(sender, hear.message);

```
  - v) **Additional hearing:**
    - (1) `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_TELL` 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.
    - (2) 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.

(3) 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.

4) **Parameters uses in games:**

a) parameters in config.txt[kernel]:

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  
#  $D = K * (\text{cycles from the start of burnup})$   
# If fire_cognition_spredding_speed  $\geq 0$ , 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
```

```

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 agent's 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:
  1. The data should have been generated by a computer program with no human interaction.
  2. Information for all known maps should have been generated by a single computer program.
  3. The computer program used for computing data for known maps should work properly if it is given a new map.
  4. An agent should choose the data file to be used itself.
  5. Agents should be able to work if no precomputed data is present for a map.
  6. 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:** The higher V value for a map, the better rescue operation.  

$$V = (P + S/S_{int}) * \sqrt{B/B_{int}}$$
P: number of living agents,

S: remaining HP of all agents,  
Sint: total HP of all agents at start,  
B: area of houses that are not burnt,  
Bint: total area at start.

- 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,
    - 1. 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.
    - 2. 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.