

Rules 3D Simulation League RoboCup 2005

Osaka

May 23, 2005

Abstract

This document contains official rules of the 3D soccer simulation competition at RoboCup 2005. While we will try to cover all cases, if unexpected events do occur, the rule committee will seek input from the participants and then make a decision. However, once the committee has made a decision, that decision is final and will not be open to further discussion. More information about this competition can be found at

<http://www.science.uva.nl/~jellekok/robocup/rc05/>

1 Simulation Environment

For the technical information, please refer to the file `3dconfiguration.pdf` available from the *Regulations and Rules* section at the website mentioned above.

2 Team Player Requirements

A valid player agent must satisfy the following conditions:

- Any attempts to subvert the timing and process tracking mechanisms of SPADES are expressly forbidden. Your responsibilities include, but are not limited to:
 - All calls to fork, clone, etc. must be dynamic calls. You can NOT use a fully statically linked executables UNLESS each of your agents runs in exactly one process and thread.
 - Your code should make no calls to change the process monitoring code.
 - Any interface with the SysV IPC or Unix domain sockets used by SPADES is expressly forbidden.

- Agents may not communicate with any process/thread not tracked by SPADES. In particular, your agent can not do any computation in any process/thread that is not spawned by the main agent thread or its descendants.
- Agents are not allowed to communicate with each other except through SPADES/rcssserver3D.
- Everyone will be expected to provide a SPADES agent database entry of type `< agent_type_external >` for your agents. The `< timer >` entry must either not exist or be `default`.

If a team is under suspicion of violating the mentioned rules, the committee has the right to ask for source code inspection.

3 Tournament Structure

3.1 General Points

- All matches will be started manually by a human referee.
- The games normally consist of two halves, each of which lasts 5 minutes. In some situations extra halves will be added, further information is provided in the sections below.
- You can either modify your code or submit new code during the competition. Any consequent problems that might arise are at your own risk, of course. Make sure that you see the section about Agents Failure. Note that upload of your code will only be possible between rounds in the morning before the games start !
- There will be three points allocated for a win and one point for a draw. A forfeit will record a score of either 3:0 or the score of the forfeiting teams other game in that round with the largest goal differential, if it is larger than 3.
- You are expected to provide a SPADES agent database for your agents. Please have a look at the sample agentdb.xml file provided with the simulator package. Place it in top of your home directory (e.g. `/home/[teamdir]/agentdb.xml`).
- The agent database file will be used by a different user in your user group. Your file and your team binary have to be given at least group read and executable permissions.
- If your team does not work we will not fix it.

3.2 Tournament Procedure

The tournament procedure will be as follows:

- It will be mandatory for every team to test their binaries *before* the first day of the competition. If a team's binary cannot be started within 5 minutes of the scheduled time, this will lead to disqualification, unless there are obvious problems with the server !
- The competition will consist of a total of 4 rounds plus the finals. For the preliminary round, seeding will be made by the committee according to ranking of the RoboCup 2004 competition. The teams are randomly assigned to the groups.
- In the preliminary round teams will be distributed to 8 groups each of 4 teams, where each team plays against all the other teams of the same group *twice*. The 3 top teams of each group proceed to the first round. For tie breakers, see the corresponding section.
- In the first round teams will be distributed to 4 groups of 6 teams. Each team plays against all the other teams of the same group. The 4 top teams of each group will advance to the second round.
- In the second round teams will be distributed to 4 groups of 4 teams, with each team playing against all the others in the same group. The 2 top teams of each group advance to the third round.
- In the third round teams will be distributed to 2 groups of 4 teams, with each team playing against all the others in the group *twice*. The 2 top teams will advance to the semi-finals.
- In the final round, each game must have a winner. If the game has no winners, three extra half-times will be played with the golden goal rule. If there is still no result, the coin toss by the human referee will determine the winner!

4 Tiebreakers

Tiebreakers between $n \geq 2$ teams in the first two rounds will apply in following priorities:

1. points
2. head-to-head results
for $n = 2$, this breaks the tie if and only if the head-to-head match had a winner
for $n > 2$, this breaks the tie if and only if one teams point is higher than the other teams only in the games between tied teams.

3. overall goal difference for this round
4. the goal difference in the games between the tied teams
5. if ($n > 2$), overall number of goals scored including only games with the tied teams
6. overall number of goals scored in this round
7. if $n = 2$, three extra halves with golden goal rule will be used to determine the winner of the game. if $n > 2$, first the teams are ranked using the game statistic values (numbers 3 – 6). Any remaining teams are broken uniformly randomly. The ranking is used for a standard single elimination bracket where each pair plays three extra halves with golden goal rule. If the game has no winner the coin toss by the human referee will determine the winner.
8. coin toss!
The exception to the above list: If teams are tied and it is the case that some team(s) in the tie will advance to the next round and some team(s) will not, then game statistic values (numbers 3 – 6) will NOT be used to break the tie.

5 Drop Ball

In certain situations, like kick-ins, the game is stopped. If a team fails to put the ball back into play after a kick-in, a drop ball is given after 30 seconds automatically.

If repeatedly no player of the team that has to start the game displays efforts to move toward the ball, the waiting time can be suitably shortened by human referee dropping the ball manually. The goal is to keep the game running as smoothly as possible while giving the teams a fair chance to exert their rights.

If in a play-on situation no player goes to the ball, or no player can kick the ball, the referee can drop the ball after 30 seconds. Especially in this case, referees are encouraged to use some common sense. E.g. if a player needs to run a long way to get to the ball and is nearly there after 30 seconds they should be given the chance to take the kick before dropping the ball. The ball should not be dropped before the 30th second. But if repeatedly no player can kick the ball, the referee should drop the ball in a shortened time.

6 Fouls

Currently kick-ins are detected automatically by the 3D soccer server. Sometimes, however, fouls occur which can only be detected by the human referee who has to award a free kick to the disadvantaged team. Some reasons to call a foul are including but not limited to:

- if one team surrounds the ball so that the other team cannot reach the ball
- if the goal is blocked by excessive players so that the ball could not go in (rough guideline: a wall of players blocking the goal)
- if a team intentionally blocks the movement of opponent players
- anything else that appears to violate the fair play commitment may also be called as a foul after consultation with the rule committee.

7 Fair Play

The goal of the game is to play soccer according to fair and common sense understanding of soccer and to restrictions imposed by the virtual simulated world of the 3D soccer simulator. Circumvention of these is considered violating the fair play commitment and its use during the tournament games is strictly taboo.

Violation of the fair play commitment play includes for example:

- using another teams binaries in your team
- jamming the simulator by sending excessive commands per client
- direct communication between players by using other communication means, such as inter-process communication

Any of these is strictly forbidden.

Other strategies might be found violating the fair play commitment, after consultation with the rule committee. However, we expect it to be pretty clear what a fair team should like. In particular, the destructive disruption of opponent agent operation or the gain of advantage by other means than explicitly offered by the simulator counts as not fair play. If you are in doubt of using a certain method, please ask the rule committee before the tournament starts. If a team is found to use unfair programming methods during the tournament, it will be immediately disqualified.

8 Valid Game

8.1 Agents Failure

In case of agents failure due to any programmatic problems related to the agent itself, the following procedure will be applied.

- If your agents get disconnected from the simulator before the 30th second, the game would be restarted up to three times.

- If the problem yet exists by agreement of the other side, you can do minor changes in your code or change your team binary in less than 2 minutes. It would be the last opportunity for you. After that if the agents again do not run properly; the game would not get disturbed. The game will be continued or you will lose with a score of 3:0.
- If your agents get disconnected from the simulator after the 30th second, the game will continue, and if it is impossible to continue you will lose with a score of 3:0.

8.2 Server Failure

In case of any failure in the simulator that can be regarded as to be in favor of either sides the human referee will decide on continuation of the game by consulting the rule committee. If the committee decides on stopping the game, the match will be restarted from the beginning.

9 Remote Participation

General Remark: Remote participation is only possible in extreme cases.

We will not have the resources to search for problems in remote participants startup procedure, so, in their own interest, remote participants are asked to make sure that a 3rd party (i.e. we) can start up your code easily and smoothly on a platform that might be different from their development platform. We unfortunately are not able to guarantee the participation of a remotely participation agent team if we do not succeed in getting the code quickly and smoothly to run.

10 After Competition

Teams will be published automatically after the competition. To help us with publishing your team, install everything necessary to start your team below `/home/[teamdir]/RC2005/`. In case a team fails to provide a working executable, we will publish the complete home directory of that team. Also, it may cause disqualification in RoboCup 2006 competitions.