RoboCup 2018
RoboCup Rescue Simulation League
Virtual Robot Competition

Rules Document

Technical Committee:
Fatemeh Pahlevan Aghababa, Masaru Shimizu

Organization Committee:
Francesco Amigoni, Amirreza Kabiri

Executive Committee:
Arnoud Visser

version 1.2: May 12, 2018

Abstract

The purpose of this competition is to provide a common benchmark to demonstrate scientific progress in the application of robotics to Urban Search and Rescue. The rules of this competition are loosely inspired by the rules of the RoboCup Rescue Robot League and of the Agent Competition of the RoboCup Rescue Simulation League. As in the Rescue Robot League, a devastated area has to be explored for victims by a team of robots controlled by an operator. Compared to the Rescue Robot League, the focus is on exploring larger areas with multiple robots rather than the mobility of individual robots. As in the Agent Competition, the disaster situation is not known before a competition run. The main difference between the two simulation competitions is our focus on a realistic sensory and actuation in addition to planning.The Virtual Robot Competition of this year continues to follow the outcomes of The Future of Robot Rescue Simulation Workshop[1] that has set a new milestone and designed settings which are planned to be used inside the competition for some years. This version of the rules reflects these settings.

1 Foreword

The design and implementation of a RoboCup competition is an ongoing process that is possible thanks to many people around the world volunteering a significant part of their time to this event. Contributions include improvements to the simulation engine, creation of competition worlds, and running the competition itself. The generous contribution of all volunteers in the past is warmly acknowledged. They have enabled a competition that is sustainable without excessive human effort and can remain viable. This is especially true this year, the second one after a big change in simulation platform, scenarios, and, as a consequence, in rules.

The rules described in this document for the RoboCup 2018 Virtual Robot Competition strive to further improve the progress made on *comprehensive missions*. Several years ago, the competition also addressed several sub-problems in the form of elemental test runs as a qualification for the final rounds. Since these elemental tests considered very specific capabilities, namely teleoperation, mapping, and deployment of an *ad hoc* network, they

---

[1] See https://staff.fnwi.uva.nl/a.visser/activities/FutureOfRescue/index.php, where several technical issues are discussed under "Installation Requirements" and tutorials for the different days. Some updates for the RoboCup 2017 competition are also available.

were also easier to score automatically. One problem with such tests, however, is that scores do not always directly relate to performance, i.e., rescued victims, in the comprehensive mission. Furthermore, the setup of such tests enforces the types of approaches teams can take for solving the comprehensive mission. For example, focusing solely on multi-robot coordination without good mapping, no teleoperation, and no connected communication network could also lead to good performance in the comprehensive mission. Yet, a team with such emphasis will fail most elemental tests. To remove these artificial constraints the Technical Committee hopes that the focus on a single comprehensive mission is beneficial for the competition, fostering a multitude of approaches that will be judged by a common global performance metric. The main goal of this competition is finding victims and teams are evaluated according to the number of victims found and to the time required.

As usual, suggestions, constructive feedbacks, and volunteer work are welcome, appreciated, and needed.

All teams participating in the 2018 competition agree to follow the latest version of these rules.

2 League Objectives & Background

The major technical goal of this competition is to encourage intuitive operator interfaces and autonomous and semi-autonomous algorithms that can be used to supervising and control multiple heterogeneous robots operating in challenging environments. Additionally, we aim to have a competition with a low barrier of entry for new teams and allowing a variety of approaches. In this sense, the adoption of Gazebo as simulation framework allows to easily leverage on the large amount of code available in the ROS ecosystem[2]. This should allow permanent installations of servers, each with its own world, which can be used for testing in preparation of the RoboCup event. Further, it allows teams to test their approach prior to the competition which lowers the barrier of entry for new teams.

The challenges are then a result of the environments that robots are deployed in, while the metric combines the number of victims found, the time required, and the area covered by the robots. Additionally, to foster the competition aspect, the scores are largely computable in real-time and can be displayed to an audience during each competition run, but won't be announced till all the teams have completed their map runs.

Finally, the scores should reasonably reflect performance for relevant real world problems that are modeled in simulation. Here, the simulation aspect of our league has the advantage of reproducible comparisons since all activity can be logged and ground truth data is readily available.

3 Comprehensive Missions

During the competition, indoor and outdoor search and rescue scenarios may be encountered. Before the run, the teams will be given basic information about the scenario. This will include the location of the disaster (indoor/outdoor) and possible dangers. Teams will be required to search for victims located in different places in the arena. As there are two types of victims in each map including: alive and dead victims, in order to count a victim as a successfully detected victim, at the end of each run each team should park their robots "near" each alive victim and also provide a map which is generated by their robots explorations and this map should be marked with the location of dead detected victims (near is defined as somewhere in radius of 1.5 meters around the victim, but Technical Committee has the final word on when a point is enough "close"). Teams are asked to submit maps generated by their robots to the Technical Committee for scoring, comparing performance, and sharing information between teams. Scoring will be made according to the number of correctly detected victims, the time required to detect victims, and the amount of area covered by the robots. It should be noted that fault detections will cost penalty points.

There can be two types of victims: alive or dead victims. Alive victims are characterized by at least one of the following features:

(a) They have a *hot* skin color that cannot be distinguished by the operator (human) eye looking at the images returned by cameras onboard robots, but can be distinguished using thermal cameras[3].
(b) They move.

[2] See tutorials at http://wiki.robocup.org/Rescue_Simulation_Virtual_Robot_Competition
[3] https://github.com/m-shimizu/p3at_for_ros_with_modelsdf

(c) They emit sound, which can be detected using a microphone in the surroundings of the victim.

In missions in which there are two types of victims, the number of found victims is calculated according to the number of successfully detected victims. Alive victims are successfully detected if they are discovered (seen) and a robot is parked close to each of them. Dead victims are successfully detected if they are discovered (seen) and their locations should also be marked near the victim on the map. It is also required that, when a victim is found, its status (alive or dead) is clearly declared. If each of the above detections *–the victim detection and the victim's status detection-* are made autonomously there will be more points considered. The details of scoring formula could be found in Section 5.

Only robots validated before the competition will be allowed to be used in the competition. The list of accepted robots and sensors is reported in this document (Section 8). Note that not every combination is possible. The sensor load will be examined and the Technical Committee reserves the right to disallow any unrealistic combination of robots and sensors.

Robot teams will be formed by a number of mobile robots, plus a base station (a robot that does not move) that provides interface to the operator.

4 Running Missions

4.1 Simulation environment

Simulations will run in Gazebo 8 on a Ubuntu 16.04.04 LTS (Xenial 64 bit) with ROS Kinetic machine[4]. During the competition, the organizers will provide two server computers. Gazebo will be run in server mode on each server computer. The two servers allow for a parallel run of two teams. Each team has 20 minutes to setup on its assigned server. The run starts at the scheduled time. If a team is not ready, time will start anyway. Each competition run lasts about 20 minutes (exact time to be announced before the run). Each team will run its client code on its own machines. A single TCP/IP cable will be provided to the team to connect to a server.

Connection to the simulation server happens using direct ROS control, each team sends commands according to the general ROS-Gazebo interface[5]. The server runs Gazebo 8 plus the GazeboUsarSim plugin and robot_state_publisher ROS node. The teams know the IP address of the server. To ensure the fairness of the competition and the fact that teams are not cheating in communicating with the server, the Technical Committee reserves the right to inspect the code of the team at any time to ensure that the client controlling robots exchange with the simulator only data compatible with realistic situations. Reading data from sensors mounted on robots is realistic, sending commands to robot actuators is realistic. The Technical Committee reserves the right to decide about the interpretation of the term "realistic".

At the prescribed start time, the robots will be instantiated in the world. Starting poses of the robots forming a team will be provided as follows. Just before the run, the Technical Committee will provide a launch file built starting from a template provided by the team (and including the robot configurations the team would like to use) and modified with the starting poses of the robots (decided by the Technical Committee). Alternatively, the Technical Committee can provide the list of starting poses of the robots by other means (e.g., on paper).
All robots must be spawned at the start of a run, though teams can decide to activate them at their convenience.

4.2 Communication between robots

Communication between mobile robots and between them and the base station is based on realistic models.

In order to facilitate broader participation during the transition from USARSim to Gazebo, for the 2018 competition the communication between robots will be assumed to be unconstrained (both in terms of connections and of bandwidth). However, in 2018, teams will be required to run some experimental sessions with constrained communication, using the Wireless Communication Simulator (WCS) available at: https://github.com/taherahmadi/WCS

In the competition at RoboCup 2018, the Technical Committee will not enforce any segregation of the code of different robots of the team. However, for future, the Technical Committee is exploring these two possibilities:

---

[4] See installation notes at https://staff.fnwi.uva.nl/a.visser/activities/FutureOfRescue/day0.php.

[5] See https://github.com/nkoenig/pioneer3at_demo for spawning multiple robots with different namespaces.

(1) One ROS master running with Gazebo, and separating robots and delivering messages according to their namespaces (for example, with ros_con gateway).

(2) One ROS master running with Gazebo and separate ROS masters for each robot (for example, using multimaster_fk).

This applies to physical robot-environment communications, which should be kept local. All other communications (operator-robot and robot-robot) will use the above mentioned WCS, which substitutes the Wireless Communication Server WSS[6] and is running on the server machine, simulating constrained and intermittent wireless network links in a disaster setting, where multi-hop routing and autonomous behavior of robots are required. All communications between robots and between robots and base station must go through WCS. As long as the WCS is not used, the teams may assume unlimited communication, but should be aware that this situation will change next year. With WCS, the operator at the base station can send commands to a robot and will obtain measurements and video images from a robot only when that robot is in radio contact. The location of the base station and the wireless cutoff strength will be provided as *a priori* data and announced before each run. During the competition, all communications with the robots can be logged. Therefore, the Technical Committee will be able to check for communications that bypass the WCS. Teams that violate this policy are immediately disqualified, and the reason for the disqualification will be posted on the web.

5 Performance Metrics

For scoring purposes, a team member is counted as a human operator who is supposed to do the followings:

- Starts a robot, enters initial points,
- Actively drives a robot around,
- Stops a robot before the run is over (for example, to prevent it from bumping into victims),
- Is involved in any way in the victim recognition process.
- Hands the final map to the technical committee

Each team can have only one human operator for each run.

Let v be the number of victims a team detected successfully (see Section 3 for the definition of successfully detected victims) and t be the completion time for the team, i.e., the time until either all V victims in the arena have been found or the maximum mission time T is exceeded. Note that, the number v_r of successfully detected victims is the sum of:

- the number of successfully detected dead victims (with a marked point on the map close to them) and of
- the number of successfully detected alive victims (which must have a robot parked close to each of them) which is represented by v_{ar}

The score of a team is calculated the following formula:

$$Score = [\beta \times S + (1-\beta)\times(\frac{v_r - 0.5 v_w}{V} + max(0, \frac{2v_{ar} - v_{aw}}{2\,V_a}))]\alpha$$

Where the indexes r, and w respectively represent successfully detection of the victim, fault detection of the victim, and the index a represents the alive victim, while α $(0.1 < \alpha < 2)$ is proportional to the difficulty level of the map, defined by the map designer, β is a constant $(0.1 < \beta < 1)$ that balance the weight of finding victims and of exploring the environment, and S is the ratio of the area explored by robots, which is measured as follows. Each map has a number N of “invisible portals”[7] that represent important points of the environment and that are defined by the map designer (for example, invisible portals are intersections of corridors, doorways, and other relevant points

[6]The old WSS used in connection with USARSim was developed and documented by Max Pfingsthorn. The latest Windows version 0.6.1 is available on http://usarsim.cvs.sourceforge.net/viewvc/usarsim/usarsim/Tools/WSS/. This version of the WSS can also be compiled under Ubuntu and works in the noop mode (available at https://github.com/MRL-VR/WSS-Ubuntu thanks to Mohammad H. Shayesteh and Sanaz Taleghani).

[7] Invisible portals mimic the idea of “map fiducials” used in the RoboCup Rescue Robot League (see rules for the 2015 competition at: http://wiki.ssrrsummerschool.org/doku.php?id=rrl-rules-2015.

according to the environment[8]), but unknown to the teams. Explored area ratio S is calculated as n/N, where n is the number of invisible portals discovered by the robot out of N total invisible portals, representing the portion of the environment explored by robots of a team.

To encourage safe robot behavior, any contact between a robot and a (alive) victim causes the team a 20% score penalty for that victim (instead of 10 points, the team gets 8 points after a first contact, 6 points after a second contact, ..., down to 0).

Besides, before each run, the Technical Committee can communicate to the teams that operators are not allowed to control the robots manually (*handoff period*) for *k* minutes, during which robots are supposed to explore automatically.

6 Open Source Policy

The winning teams are required to provide a fully functioning copy of their software to the organizers before the final ceremony. Failure to do so will result in team disqualification. All other teams are also requested to provide their code, though not before the awards event. The software will be posted on (or linked from) the competition web pages and wikis, giving proper credit to the authors.

All data logs collected during the competition can be made available on the web for public use, including, but not limited to, scholarly work devoted to performance evaluation and benchmarking.

7 Summary

The intention of the competition is to stimulate research in robotics that allows for autonomous and safe exploration of significant parts of the environment providing aid to first responders to rescue victims.

8 Allowed Robots and Sensors

Teams are expected to use combinations of the following robots (whose models are available in Gazebo) and the following sensors (note that some configurations in *italics* are currently being tested and are not guaranteed to work properly):

P3AT (odometry, camera, battery, sonar, gps, laser range finder, thermal camera)
P2DX (odometry, camera, battery, sonar, gps, laser range finder, thermal camera)
Turtlebot (odometry, camera, battery, sonar, gps, laser range finder)
Quadrotor (camera, battery, gps, laser range finder)

Sensor data can have added noise.

Sensor load will be examined. The Technical Committee reserves the right to disallow any unrealistic combination of robots and sensors. Prior to the competition, the technical committee can publish a number of reasonable configurations that can be used during the competition. Additionally, if, during the competition, unrealistic behavior is detected for a robot or a sensor, this device can be excluded for further usage during the rest of the competition.

---

[8] In future competitions, invisible portals could be automatically generated from a map. For example, considering an indoor map, invisible portals can be generated according to a Voronoi decomposition of the map.

9 Acknowledgments

Special thanks to Amir. H. Abdi, Behzad Tabiban, Andreas Kolling, Stephen Balakirsky, and Stefano Carpin who co-authored rules of the previous years.

10 List of Known Open Issues

The Technical Committee will put any effort in solving the following issues for the next competitions (* denotes more urgent issues):

- * Check models of robots and sensor configurations (especially, battery, sonar, and gps).
- Segregation of robots of a team within ROS.
- * Create sample of launch file.
- * Check “gazebo-white-ip”: list of IP addresses that can access Gazebo and that should be disabled for prevent cheating.