RoboCup Rescue Simulation Innovation Strategy

Arnoud Visser[1], Nobuhiro Ito[2], and Alexander Kleiner[3]

[1] Universiteit van Amsterdam, Science Park 904, Amsterdam, The Netherlands
[2] Aichi Institute of Technology, 1247 Yachigusa, Yakusa Cho, Toyota City, Japan
[3] Linköpings Universitet, SE-581 83 Linköping, Sweden

Abstract. The RoboCup rescue simulation competitions have been held since 2001. The experience gained during these competitions has supported the development of multi-agent and robotics based solution for disaster mitigation. The league consists of three distinct competitions. These competitions are the agent competition, the virtual robots competition, and the infrastructure competition. The main goal of the infrastructure competition is to increase every year the challenge and to drive the innovation of the league, while the agent and virtual robot competition are focused on developing intelligent agents and robot control systems that can cope with those challenges. This paper provides an overview on the current state-of-the-art in the league and developments and innovations planned for the future.

1 Introduction

Robots are designed to do work in dull, dirty and dangerous environments. A disaster can be classified as dirty and dangerous. In several occasions rescue robots were applied after a disaster, as described by Murphy [1]. An example is the application of the Quince robot at the Fukushima Daiichi Nuclear Plant. The prototype of the Quince robot was developed at the RoboCup [2]. Yet, the typical application of a robot in this circumstance is the teleoperation of a single robot, while the scale of the disaster could benefit from a multi-agent approach.

The rescue simulation league (RSL) aims to develop realistic simulation environments for benchmarking intelligent software agents and robots which are expected to make rational decisions autonomously in a disaster response scenario. The RSL has two major competitions, namely the *agent* and the *virtual robot* competition. The agent competition consists of a simulation platform that resembles a city after an earthquake. In that environment, intelligent agents can actively mitigate the impact of the disaster and influence the cause of events after the disaster occurred. The agents have the role of police forces, fire brigades, and ambulance teams, and are mainly in charge to remove debris from the roads, extinguish fires, and to rescue civilians. The virtual robot competition has as its goal the study of how a team of robots can work together to get a situation assessment of a devastated area as fast as possible. This will allow first responders to be well informed when they enter the danger zone.

This paper builds upon the leagues' status report from eight years ago [3]. In Section 2 developments and advancements of the rescue simulation league during the last decade are described. Section 3 provides an overview on the current state-of-the-art of the league and Section 4 outlines developments and directions in the future.

2 Past 10 years: Rescue Simulation League

The RoboCup Rescue Simulation League (RSL) started in 2001. The RSL aims to develop simulation environments which benchmark the intelligence of software agents and robots with the capabilities for making the right decisions autonomously in a disaster response scenario. Both the two major competitions of the RoboCup Simulation League, namely the *Agent* and the *Virtual Robot* competitions are described in the subsequent sections.

2.1 Agent Competition

The rescue agent competition aims to simulate large scale disasters and to explore new ways of autonomous coordination of rescue teams as an approach of disaster relief after real world incidents [4, 5]. The competition consists of a simulation platform which resembles a city after an earthquake. For example, Fig. 1 depicts the simulation of fires and building collapse on a model of a virtual city. Teams of fire brigade, police and ambulance team agents try to extinguish fires and rescue victims in the collapsed buildings. Scoring is based on the number of victim saved on time and the number of buildings with fire damage. The problem of disaster mitigation requires the coordinated action of several heterogeneous and decentralized units. Due to the variation of different potential disaster scenarios, typically strongly diverse teams of actors are required rather than a single agent type. RSL fosters the development of algorithms for coordinating heterogeneous agents. Different types of intelligent agents can be spawned into the simulation environment for mitigating the effects of virtual threats, such as building collapse and fire. To this end, the agents may take on different roles such as *police force*, *fire brigade*, and *ambulance teams* each having different capabilities.

The overall goal is to develop robust software systems that are capable of efficiently coordinating large agent teams for Urban Search and Rescue (USAR). This goal raises several research challenges, such as the *exploration* of large scale environments in order to search for survivors, the *scheduling and planning* of time-critical rescue missions, *coalition formation* among agents, and the *assignment* of agents and coalitions to dynamically changing tasks, also referred to as *extreme teams* [6]. In the rescue domain, this issue is even more challenging due to restricted communication bandwidth. Moreover, the simulated environment is highly dynamic and only partially observable by a single agent. Under these circumstances, the agents have to plan and decide their actions asynchronously in real-time while taking into account the long-term effects of their actions.

Several authors tried to solve this challenge with a formal approach [7–9]. Their approaches are generally applicable, but still have difficulties to deal with real-time constraints when compared to heuristic approaches of specialized champion teams. To overcome this problem, the simulation league has initiated a new

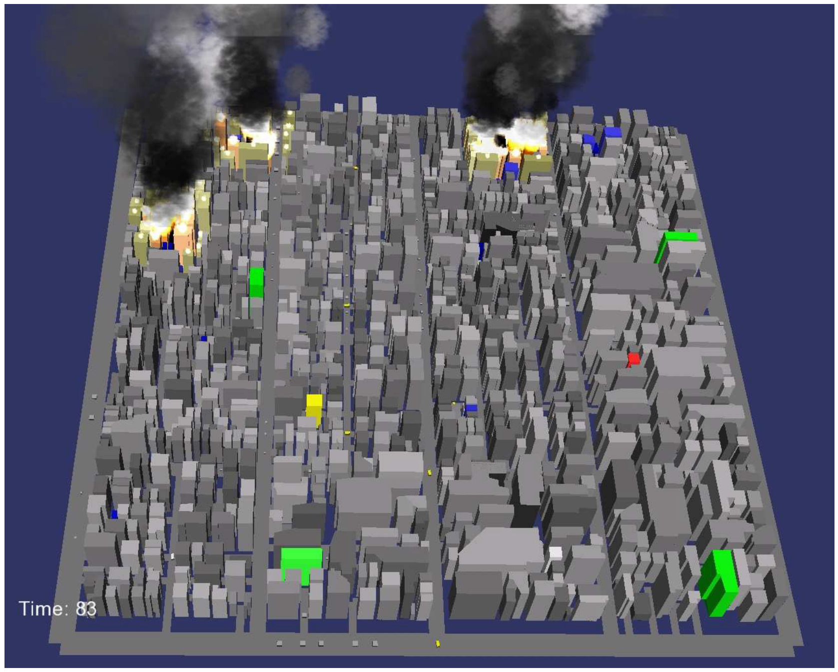

Fig. 1. A typical RoboCup Rescue Simulation scenario.

type of challenge [10]. The idea is to extract from the entire problem addressed by the agents certain aspects such as task allocation, team formation, and route planning, and to present these sub-problems in an isolated manner as stand-alone problem scenarios with an abstract interface. As a consequence, participating teams can focus on their research on an aspect of the game, without having to solve all low-level issues. This challenge is more detailed described in Section 3.2.

Over the years the winning entries in the competition showed a strong focus on highly optimized computations for multi-agent planning and model-based prediction of the outcome of the ongoing incidents. Several techniques for multi-agent strategy planning and team coordination in dynamic domains have also been developed based on the rescue simulator. Nairo *et al.* [11] first described the task allocation problems inherently found in this domain. Ferreira *et al.* [8] evaluated solutions developed for distributed constraint optimization problems (DCOPs) using problems generated by the simulation. Ramchurn *et al.* [12] modeled the problem as coalition formation with spatial and temporal constraints (CFST) and also adopted state-of-the-art DCOP algorithms for solving CFSTs. Kleiner [13] identified and described the problem of scheduling rescue missions and also introduced a real-time executable solution based on genetic algorithms.

Furthermore, there has been substantial work on building information infrastructure and decision support systems for enabling incident commanders to efficiently coordinate rescue teams in the field. For example, Schurr *et al.* introduced a system based on software developed in the rescue competitions for the training and support of incident commanders in Los Angeles [14].

Urban disaster relief simulator (early releases) The first rescue simulation package (version 0) has been released in 1999. This package was further improved and used until 2009. The server was mainly coded in the C/C++ language on FreeBSD. The structure, as shown in Fig. 2, is similar to the package of the soccer simulation league: A central kernel intermediates between the actions performed by the agents (shown within the dashed box) and the state updates computed by the simulators (shown as boxes around the kernel). The initial state and structure of the city model is provided by the GIS component, Communication between all modules is implemented by TCP/IP based message passing.

The purpose of this release was twofold. Firstly, to provide a challenging testbed to the robotics and multi-agent communities. Second, to develop through several competitions intelligent and efficient disaster relief strategies that can make contributions to mitigation solutions developed in the real world.

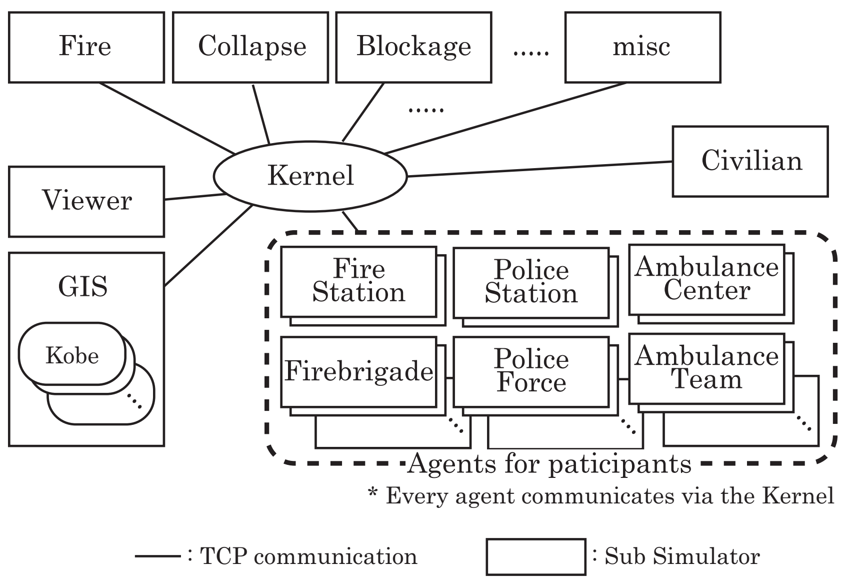

Fig. 2. The structure of rescue simulation package

Urban disaster relief simulator (advanced releases) The second simulation package (version 1) is written in Java and thus made the code much more accessible to participants during the last years. The new simulator uses area-models with polygonal representation of the road network for computing the traffic simulation. In contrary, previous traffic simulation adopted a simple graph structure to represent roads and intersections, which made the moving of agents rather simple. The new traffic simulation provides much more realistic situations, such as congestions, when multiple agents act simultaneously on the map. In order to simplify the implementation of path planning for participants, a SDK for navigation on the map is provided with the release.

During the last years, several variations of the basic competition setup were introduced. These variations were, for example, maps of constantly increasing size, an increasing of the number of agents that have to be controlled, an increasing number of fire ignition points, changing communication conditions between agents, also including no communication at all, and adding gas stations and water points. Also the simulators within each release were constantly improved. For example, teams from the league developed a more realistic fire simulator, a 3d viewer, and several other simulators and tools. Most of these improvements were essentially suggested through the Infrastructure competition.

In agent competition has organized itself with a maintenance committee. The committee maintains and implements new features for the simulator. Therefore, this maintenance committee also takes part in the rule discussion with technical committees.

2.2 Infrastructure competition

The infrastructure competition has started in 2004 with the purpose of promoting the development of new simulators and tools to continuously improving the rescue simulator. The simulation of various disaster situations turns out to be complicated and difficult to validate. Therefore, the infrastructure competition has been launched for supporting the maintenance and development of the simulator. For example, the fire simulator [15] and 3D viewer (Fig. 1) were both developed by the winner of the infrastructure competition in 2004.

The Aladdin project[4], funded by the Engineering and Physical Sciences Research Council, has strongly stimulated the conversion towards version 1 of the simulator. The original version of the simulator was chosen as the testbed for ALADDIN technologies and was further improved and extended as part of the project. The end-result is a well-engineered simulator with realistic traffic simulation (developed in collaboration with Meijo University) and GIS map conversion (in collaboration with Freiburg University) that allows disasters to be simulated in selected parts of any given map (OpenStreetMap or GIS formats).

Another simulator component proposed in the infrastructure competition, which is actively used, is the flood simulator [16], illustrated in Fig 3. Recently, also an extension towards flying robots has been proposed, both in the Virtual Robot [17] and Agent competition [18].

The winner of the infrastructure competition is expected to join the maintenance committees.

[4] http://www.aladdinproject.org/robocuprescue/

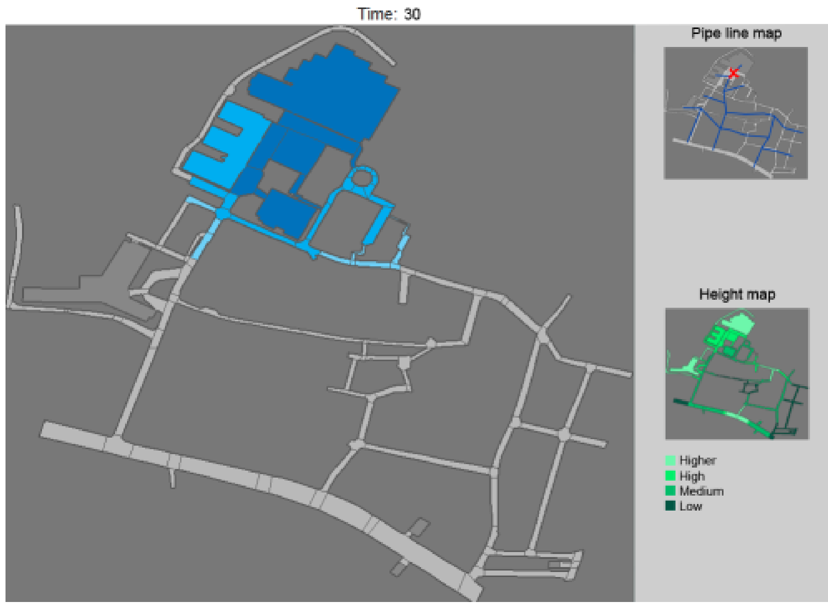

Fig. 3. The flooding of the city based on a height map. Courtesy [16]

2.3 Virtual Robot competition

RoboCup Virtual Robot competitions are being held since 2006. The competition attracts mainly academic teams from universities often with experience in the RoboCup Rescue Robot League. The important research issues related to the competition are:

Utility-based mapping - autonomous generation of maps from the fused data of multiple vehicles to be used by both robots and humans for exploration and marking victims.

Victim detection - automatic detection of victims from fused sensor data (image processing, acoustics, etc.).

Advanced mobility - robust control algorithms capable of autonomous navigation in small spaces on non-flat flooring.

Multi-robot control - the ability to control multiple platforms with a single operator in realistic environments means that the robots have to be semi-autonomous.

Originally a single scoring formula was used to evaluate the solutions associated with these issues. Simplified challenges were later introduced in 2009 to create more objective measures of performance. Each challenge was about a particular sub-problem with only one measure of performance and a corresponding automatic scoring tool. These challenges, namely mapping, navigation and coordination over a mesh network, were used for qualification for the semifinals. Since 2011 the challenges have been combined again into a single mission. The goal is now defined in terms of an entirely automated scoring procedure. The scoring program is expected to allow for head-to-head competition between two competing teams. This also allows permanent installations of servers, each with its own world, which can be used for testing in preparation of the RoboCup. In this way, the teams can test their approach prior to the competition hence providing a lower barrier of entry for new teams. Furthermore, the scoring procedure takes into account the individual sub-problems solved in the comprehensive challenge so that the teams can assess their performance in all of the domains independently.

The main challenge for the teams is the control of a large team of robots (typically eight) by a single operator. This is state-of-the art; the only real comparison is the champion of the Magic competition [19], where 14 robots were controlled by two operators. The single operator has to use high level commands (such as the areas to be searched, routes to be followed, etc.) but is also needed to verify observations whether or not one of the robots has detected a victim (based on color and/or shape). Due to poor lighting and the number of occlusions, the conditions are generally not favorable for automatic victim detection, and manual conformation is always needed. The approach to a victim is quite critical (the robot should come within the communication range ($< 1m$), but is not allowed to touch the body or any of the limbs). This means that the workload for the operator is quite high, providing an advantage for the teams which are able to automate the decision making within the robot team as far as possible, and only involve the operator when needed.

The shared map generated during the competition has a central role in the coordination of such large robot teams. This is where the sensor information selected to be broadcasted via often unreliable communication links is collected and registered. The registration process is asynchronous; some information may arrive at the base-station even minutes after the actual observation. There is no guarantee that the operator has time to look at this information directly, which implies that the map within the user interface has to be interactive and should allow the operator to call back observations that were made at any point of interest (independent of when the observation was made and by which robot). At the same time the registration process should keep the map clean (no false positives or wrong associations), because it is the area where the coordination of the team behaviors is done.

3 Current Rescue Simulation League

3.1 Overview

The main purpose of current simulator (version 1) is the benchmark for decision making for multi-agent systems [20, 7, 8]. By providing the benchmark framework of disaster relief simulator, it is possible to compare the effectiveness of the different approaches.

The initial purpose was mainly to stimulate contributions which could be directly applied in the real world. Yet, realistic planning problems include hundreds of agents, which stimulated the major version update from the earlier releases to the advanced releases. The current competitions include the challenge to plan optimal decisions for large teams of agents.

Changes of the number of pre-registered teams are shown in Table 1. In the Table 1, we can see the number of teams has become about half from 2010, and the number of teams has not been recovered yet. Besides, the rate of newcomers among the participants is rather low. This implies that our current approach is not attractive enough to extend our community.

Table 1. The numbers of teams participating in the Agent competition

2001	**2002**	**2003**	**2004**	**2005**	**2006**	**2007**	**2008**	**2009**	**2010**	**2011**	**2012**	**2013**	**2014**
7	8	30	34	40	42	31	35	22	33	21	22	16	24

In the following section we describe approaches the technical committee has undertaking in order to make the competition more attractive to a larger community.

3.2 Multi-Agent Challenge

Our first approach is to provide the useful benchmark framework, *RMasBench*. The RMasBench is a new type of challenge that has been introduced in 2011 [10]. The idea is to extract from the entire problem addressed by the agents certain research relevant aspects such as task allocation, team formation, and route planning, and to present these sub-problems in an isolated manner as stand-alone problem scenarios with an abstract interface. Consequently, the participating teams are able to focus more on topics relevant to their own research rather than dealing with all the low-level issues. At the current stage, *RMasBench* introduces a generic API for distributed constraint optimization problem (DCOP) algorithms and reference implementations of state-of-the-art DCOP solvers, such as *DSA* [21] and *MaxSum* [22].

3.3 Communication Library

The second approach is the development of a library that implements communication protocols for agens. The rescue simulation league releases every year the source codes from the top-three teams and the specific scenarios that were used during the competitions. However, it turned out that the code released by teams in the past is hard to be re-used by other researches. To solve this problem, we are developing the communication library, of which the first version has already been released. When teams use this communication library, it is easy to share their source code, since team approaches are based on the same communication protocol. This library might also allow to create something comparable to the drop-in challenge of the Soccer competition; the cooperation of fire-agents from one team with police-agents from another team.

3.4 Virtual evacuation with 3D viewer

The current 2D viewer is not very attracti e for the audience. To solve this problem, the 3D viewer has been released for the earlier simulator (version 0). But, the viewer is not ported for the advanced simulator (version 1). Therefore, we also provide the challenge to develop new 3D viewer. Through this new 3D viewer, the spectator is embedded inside the disaster experience, which could be well used in the training of rescue forces (Fig. 4).

Fig. 4. 3D viewer for the agent competition

3.5 Evacuation simulator

By expanding the current simulator, an evacuation simulator has been developed [23]. It enables to simulate whether the evacuation instructions of commanders are effective or not. Besides, the situations are also directly visualized through the simulations. Also this extention could be used in the training of rescue force commanders.

3.6 Virtual Robot competition

The number of participating teams is relatively small (Table 2). The small number of teams is partly due to difficulty of the challenge, and partly due to the strict qualification rules of this competition. Yet, the potential for a large participation in this competition could be quite large, as is demonstrated with the 100 teams which have registered in the simulation track of the DARPA Robotics Challenge [24, 25]. Unfortunately, the RoboCup cannot offer the same prize money as the DARPA organization, so we should attract teams with our inspiring academic climate.

Table 2. The numbers of teams participating in the Virtual Robot competition

2006	2007	2008	2009	2010	2011	2012	2013	2014
8	8	10	11	5	6	4	4	5

The challenge has the tendency to become harder ever year. Semi-autonomy was always a pre-requisite to be able to control a large team of robots with a single operator. In 2013 a hands-off period of k minutes was introduced in the rules; a period where the robots have to explore the environment fully autonomously. In 2014 this period is already increased to $k = 10$ minutes: 50% of the duration of a mission.

In addition, the scale of the simulation has increased. Outdoor maps with a size of 1km by 1km have been created, which is close to the size of the maps used in the agent competition. Yet, the game engine is not (yet) able to simulate 100 robots, including their sensory payload, in this environment.

4 Future Directions

In the short term the league could step-by-step by further developed towards the diverse situations which could be encountered during an disaster. An example is the introduction of a smoke simulator in the agent competition, comparable with the smoke simulator introduced in the virtual robot competition [26].

The RoboCup federation formulates the goal of the Rescue Simulation League as follows: *The purpose of the RoboCup Rescue Simulation league is twofold. First, it aims to develop simulators that form the infrastructure of the simulation system and emulate realistic phenomena predominant in disasters. Second, it aims to develop intelligent agents and robots that are given the capabilities of the main actors in a disaster response scenario.* On the one hand side, because the virtual robot competition simulates the real robot competiton, it is in line with this purpose. But, the competition requires advanced skills in perception, mapping and exploration while multi-agent researchers want to concentrate on decision making at the exploration level. On the other hand side, the agent competition has brought the planning and high-level decision making to another level. The competition has been proven to be a stable benchmark for a select group of researchers. Still, it remains one of our goals to integrate the agent competiton with the virtual robot competition, and aim to improve our simulator so that it can contribute the results in the real world.

5 Conclusion

In RoboCup, the rescue simulation league allows many academic teams without the resources to travel with several rescue robots over the world to contribute to the developments in this field. Many researchers have contributed to this league in the past. Therefore, we have to decide the direction of our community so as to be sufficient for both new and established researchers. To differentiate from the soccer simulation, our purposes should be directed to the contributions for real world through our simulations. The competition was initiated to show that the robotics community could contribute with a social relevant application. The league will continue to contribute to RoboCup and the real world by keeping innovating.

References

1. Murphy, R.: Disaster Robotics. Intelligent Robotics and Autonomous Agents Series. MIT Press (2014)
2. Nagatani, K., Kiribayashi, S., Okada, Y., Otake, K., Yoshida, K., Tadokoro, S., Nishimura, T., Yoshida, T., Koyanagi, E., Fukushima, M., et al.: Emergency response to the nuclear accident at the fukushima daiichi nuclear power plants using mobile rescue robots. Journal of Field Robotics **30** (2013) 44–63
3. Skinner, C., Barley, M.: Robocup rescue simulation competition: Status report. In Bredenfeld, A., Jacoff, A., Noda, I., Takahashi, Y., eds.: RoboCup 2005: Robot Soccer World Cup IX. Volume 4020 of Lecture Notes in Artificial Intelligence. Springer Berlin / Heidelberg (2006) 632–639
4. Kitano, H., Tadokoro, S., Noda, I., Matsubara, H., Takahashi, T., Shinjou, A., Shimada, S.: RoboCup Rescue: Search and rescue in large-scale disasters as a domain for autonomous agents research. In: IEEE Conf. on Man, Systems, and Cybernetics(SMC-99). (1999)
5. Tadokoro, S., Kitano, H., Takahashi, T., Noda, I., Matsubara, H., Shinjou, A., Koto, T., Takeuchi, I., Takahashi, H., Matsuno, F., Hatayama, M., Nobe, J., Shimada, S.: The RoboCup-Rescue project: A robotic approach to the disaster mitigation problem. In: Proceedings of the IEEE International Conference on Robotics and Automation. (2000)
6. Scerri, P., Farinelli, A., Okamoto, S., Tambe, M.: Allocating tasks in extreme teams. In: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, ACM (2005) 727–734
7. Chapman, A., Micillo, R.A., Kota, R., Jennings, N.: Decentralised dynamic task allocation using overlapping potential games. The Computer Journal **53** (2010) 1462 –1477
8. Ferreira, P., Dos Santos, F., Bazzan, A., Epstein, D., Waskow, S.: Robocup rescue as multiagent task allocation among teams: experiments with task interdependencies. Autonomous Agents and Multi-Agent Systems **20** (2010) 421–443
9. Dos Santos, F., Bazzan, A.L.: Towards efficient multiagent task allocation in the robocup rescue: a biologically-inspired approach. Autonomous Agents and Multi-Agent Systems **22** (2011) 465–486
10. Kleiner, A., Farinelli, A., Ramchurn, S., Shi, B., Maffioletti, F., Reffato, R.: RMASBench: Benchmarking dynamic multi-agent coordination in urban search and rescue. In: Proceedings of the Twelfth International Conference on Autonomous Agents and Multiagent Systems (AAMAS2013). (2013) 1195–1196
11. Nair, R., Ito, T., Tambe, M., Marsella, S.: Task allocation in robocup rescue simulation domain. In: Proceedings of the International Symposium on RoboCup. (2002)
12. Ramchurn, S., Farinelli, A., Macarthur, K., Jennings, N.: Decentralized coordination in robocup rescue. The Computer Journal **53** (2010) 1447–1461
13. Kleiner, A., Brenner, M., Bräuer, T., Dornhege, C., Göbelbecker, M., Luber, M., Prediger, J., Stückler, J., Nebel, B.: Successful search and rescue in simulated

disaster areas. In Bredenfeld, A., Jacoff, A., Noda, I., Takahashi, Y., eds.: RoboCup 2005: Robot Soccer World Cup IX. Volume 4020 of Lecture Notes in Computer Science., Springer (2005) 323–334
14. Schurr, N., Tambe, M.: Using multi-agent teams to improve the training of incident commanders. Defence Industry Applications of Autonomous Agents and Multi-Agent Systems (2008) 151–166
15. Nüssle, T.A., Kleiner, A., Brenner, M.: Approaching urban disaster reality: The ResQ firesimulator. In Nardi, D., Riedmiller, M., Sammut, C., Santos-Victor, J., eds.: RoboCup 2004: Robot Soccer World Cup VIII. Volume 3276 of Lecture Notes in Computer Science., Springer (2004) 474–482
16. Shahbazi, H., Abdolmaleki, A., Salehi, S., Shahsavari, M., Movahedi, M.: Robocup rescue 2010 – rescue simulation league team description paper - brave circles - infrastructure competition. In: Proc. CD of the 14th RoboCup International Symposium. (2010)
17. Dijkshoorn, N., Visser, A.: An elevation map from a micro aerial vehicle for urban search and rescue. In: Proceedings CD of the 16th RoboCup International Symposium. (2012)
18. Gohardani, P.D., Ardestani, P., Mehrabi, S., Yousefi, M.A.: Flying agent: An improvement to urban disaster mitigation in robocup rescue simulation system. Mechatronics Research Laboratory, Qazvin, Iran (2013)
19. Olson, E., Strom, J., Morton, R., Richardson, A., Ranganathan, P., Goeddel, R., Bulic, M., Crossman, J., Marinier, B.: Progress towards multi-robot reconnaissance and the MAGIC 2010 competition. Journal of Field Robotics **29** (2012) 762–792
20. Oliehoek, F.A., Visser, A.: A Decision-Theoretic Approach to Collaboration: Principal Description Methods and Efficient Heuristic Approximations. In: Interactive Collaborative Informations Systems. Volume 281 of Studies in Computational Intelligence. Springer-Verlag, Berlin Heidelberg (2010) 87–124
21. Fitzpatrick, S., Meetrens, L.: Distributed Coordination through Anarchic Optimization. In: Distributed Sensor Networks A multiagent perspective. Kluwer Academic (2003) 257–293
22. Farinelli, A., Rogers, A., Petcu, A., Jennings, N.R.: Decentralised coordination of low-power embedded devices using the max-sum algorithm. In: Proceedings of the Seventh International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008). (2008) 639–646
23. Okaya, M., Takahashi, T.: Proposal for everywhere evacuation simulation system. In Röfer, T., Mayer, N.M., Savage, J., Saranlı, U., eds.: RoboCup 2011: Robot Soccer World Cup XV. Lecture Notes in Artificial Intelligence. Springer, Berlin, Heidelberg, New York (2012) 246–257
24. Ackerman, E.: Darpa robotics challenge trials: What you should (and shouldn't) expect to see. IEEE Spectrum **19** (2013)
25. Luo, J., Zhang, Y., Hauser, K., Park, H.A., Paldhe, M., Lee, C.S.G., Grey, M., Stilman, M., Oh, J.H., Lee, J., Kim, I., Oh, P.: Robust ladder-climbing with a humanoid robot with application to the darpa robotics challenge. In: IEEE Int'l. Conf. on Robotics and Automation. (2014)
26. Formsma, O., Dijkshoorn, N., van Noort, S., Visser, A.: Realistic Simulation of Laser Range Finder Behavior in a Smoky Environment. In: RoboCup 2010: Robot Soccer World Cup XIV. Volume 6556 of Lecture Notes in Artificial Intelligence. Springer (2011) 336–349