J Intell Robot Syst (2013) 72:147–165
DOI 10.1007/s10846-013-9822-x

Robotic Urban Search and Rescue: A Survey from the Control Perspective

Yugang Liu · Goldie Nejat

Received: 23 July 2012 / Accepted: 13 February 2013 / Published online: 12 March 2013

Abstract Robotic urban search and rescue (USAR) is a challenging yet promising research area which has significant application potentials as has been seen during the rescue and recovery operations of recent disaster events. To date, the majority of rescue robots used in the field are teleoperated. In order to minimize a robot operator's workload in time-critical disaster scenes, recent efforts have been made to equip these robots with some level of autonomy. This paper provides a detailed overview of developments in the exciting and challenging area of robotic control for USAR environments. In particular, we discuss the efforts that have been made in the literature towards: 1) developing low-level controllers for rescue robot autonomy in traversing uneven terrain and stairs, and perception-based simultaneous localization and mapping (SLAM) algorithms for developing 3D maps of USAR scenes, 2) task sharing of multiple tasks between operator and robot via semi-autonomous control, and 3) high-level control schemes that have been designed for multi-robot rescue teams.

Keywords Urban search and rescue · Rescue robots · Robot autonomy · SLAM · Semi-autonomous control · Multi-robot control

Y. Liu (✉) · G. Nejat
Autonomous Systems and Biomechatronics Laboratory, Department of Mechanical and Industrial Engineering, University of Toronto, 5 King's College Road, Toronto, ON, Canada M5S 3G8
e-mail: yugang.liu@mail.utoronto.ca

G. Nejat
e-mail: nejat@mie.utoronto.ca

1 Introduction

Robotic Urban Search and Rescue (USAR) has attracted significant attention in the last decade with the participation of a handful of robots in the rescue and recovery operations of many past and recent devastations, such as the 2001 World Trade Center (WTC) collapse [1], the 2004 Mid Niigata earthquake in Japan [2], the 2005 Hurricanes Katrina, Rita, and Wilma in the United States [2], as well as the 2011 Tohoku earthquake and tsunami in Japan [3, 4]. In general, the overall goal of a rescue robot in a USAR mission is to explore unknown cluttered disaster scenes while searching for victims. While still in its infancy, the field of rescue robotics has exhibited promising potential in USAR applications [1–4]. The great losses in lives caused by such tragedies have demonstrated the need for the development of robotic aids to assist rescue workers in such time-critical and life-risking undertakings [5].

Urban disaster environments have been known to be very difficult to access by rescue workers due to the potential presence of asbestos, dust, poisonous gases, hazardous materials, radiation or extreme temperatures [6]. Rescue robots provide a promising solution to assist rescue workers in terms of: 1) reducing personal risk to workers and rescue dogs by entering unstable structures, 2) increasing speed of response by penetrating ordinarily inaccessible voids, and 3) through the use of multiple cameras and sensor fusion in order to extend the reach of rescue workers to regions that are otherwise inaccessible. Compared with rescue workers and trained rescue dogs, rescue robots have many advantages: i) unlike their human counterparts, a rescue robot will not become stressed or fatigued [7]; ii) rescue robots can be developed in large quantities, while experienced rescue professionals and trained rescue dogs are sparse resources [6]; and iii) robots are expendable but humans and rescue dogs are not: if a rescue robot is damaged, it can be easily repaired or replaced, but the loss of rescue workers or dogs could be very difficult due to their relationship within society [5]. There are numerous risks involved when working in a USAR task force, and the health and safety of the rescue workers and victims are always of primary concern. Namely, rescue workers are prone to physical injuries (such as cuts, scrapes, burns, and broken bones), respiratory injuries (due to hazardous materials, fumes, dust and carbon monoxide) and diseases such as gastro-esophageal reflux disease [8]. Furthermore, a rescue worker may experience psychological and emotional trauma caused by gruesome scenes [9].

While a promising and important application for robotics, urban disaster environments are very challenging for robots as they are highly cluttered, unstructured and unpredictable. Hence, it is extremely difficult for robots to autonomously navigate USAR scenes and also search for victims [5]. As a result, to date, search and rescue operations have required a human operator in the loop, in order to guide a robot remotely [1, 2]. However, human operators can have perceptual difficulties in trying to understand a cluttered 3D environment via remote visual feedback, and therefore, can become disoriented and lose situation awareness, resulting in rescue robots frequently becoming physically stuck [6]. While challenging, to address the limitations of teleoperation, research towards improving low-level robot autonomy is underway, including control approaches to improve robot mobility in negotiating irregular terrain [10–15], and simultaneous localization and mapping (SLAM) based algorithms in order to allow for a robot to autonomously map a USAR environment and localize and track its own location within this physical environment via the generated map [16–20].

To effectively integrate teleoperation with low-level robot autonomy of rescue robots, the development of semi-autonomous controllers has recently been considered [21–28]. Semi-autonomous control allows for task sharing between a robot and an operator [29]. Such task sharing can reduce the stress and mental workload of an operator and also allow a rescue robot to benefit from the experience and knowledge of the operator. With semi-autonomous control, some low-level tasks can be performed by a robot, in order for the operator to focus his/her attention on more high-level control tasks including the supervision or operation of multiple robots. Therefore, robot task autonomy in semi-autonomous control methodologies has led the way for the design of single-human multi-robot (SHMR) systems for potential application in USAR environments. The use of multiple robots can be more cost-effective as it allows for shorter robot deployment time as well as addresses the mobility constraints associated with the deployment of a single robot. As a result, teamwork, including human-robot cooperation and multi-robot coordination, becomes essential for rescue robots [30–40]. In addition, the deployment of multiple robots and the subsequent maintaining and enhancing connectivity between them in the field has, hence, also become a popular research area [34–44].

In this paper, we present an overview of the state of the art in single robot and multi-robot control for USAR environments. Namely, we discuss the proposed approaches that have been developed for low-level robot control, semi-autonomous control and multi-robot control in order to improve the performance of rescue robots in cluttered USAR scenes and minimize the workload of robot operators. The rest of this paper is organized as follows: Section 2 provides an overview of the techniques designed to provide low-level autonomy to rescue robots. In Section 3, semi-autonomous control schemes for rescue robots are discussed. Section 4 presents a discussion on control methodologies for multi-rescue robot teams. Summary remarks are provided in Section 5.

2 Low-Level Robot Autonomy

From the tracked mobile robots deployed in the 2001 WTC disaster site [1, 2], to the snake-like robot and tracked robots that were deployed recently at the sites of the 2011 Earthquake in Japan [3], most of the rescue robots deployed in real-world scenarios have been teleoperated. In general, human operators can become stressed, cognitively/physically fatigued and disoriented, resulting in low levels of alertness, lack of memory and concentration, and loss of situation awareness during vital search and rescue operations, as has been reported in [1, 2, 5]. For example, at the WTC site, the loss of situation awareness was determined to be one of the main reasons why rescue robots frequently got stuck in the rubble-filled environment, namely, the robots that were deployed were reported to get stuck 2.1 times per minute [6]. Moreover, when relying on the remote visual feedback from a robot, human operators were often unable to perceive whether the robot could pass through openings or traverse over obstacles due to scale ambiguity and the keyhole effect (trying to understand the surrounding environment through a small hole) [45].

In order to improve the performance of rescue robots in USAR environments, research into equipping these robots with some level of autonomy is currently ongoing. As previously mentioned, the main two tasks of a rescue robot consist of exploring an environment and performing victim identification. The latter task is very difficult for a robot to accomplish due to the sensing challenges in these environments. For example, vision based techniques are difficult to implement due to the inherent challenges of computer vision under unstructured lighting conditions [1]. Infrared cameras were found not to be useful in the WTC site, due to the high temperatures of certain rubble piles, which had masked the signs of survivors [1]. Furthermore, low-cost acoustics and CO2 sensors may not perform as expected due to the potential existence of noise and other gases in USAR environments [46].

The highly cluttered and unstructured characteristics of USAR environments as well as the irregularity of the terrain conditions have presented severe challenges to the development of autonomous exploration capabilities for rescue robots. A handful of researchers have been focusing on addressing mobility issues of rescue robots, in order to allow for such robots to autonomously navigate cluttered USAR environments. In particular, autonomous stair-climbing and uneven terrain traversing, which have been shown to be challenging for teleoperation, have attracted some attention from the rescue robotics community [10–15]. Furthermore, some efforts have also been reported on utilizing SLAM techniques to improve robot perception capabilities and situation awareness [16–20].

2.1 Traversing Stairs and Uneven Terrain

Stairs are omnipresent in urban areas, and controlling a robot to climb them can be very demanding for a human operator in USAR missions [10]. Teleoperated robotic stair climbing that relies on feedback from a robot's 2D camera has been proven to be inefficient and unreliable, resulting in inaccurate and slow stair climbing, collisions with the stair walls and even flipping over of the robot [47].

In [10], a vision-guided autonomous stair climbing technique was presented. The technique relies on determining a robot's rotational velocity using a 3-axial gyroscope and measuring the location of stair edges with a 2D camera to estimate via an extended Kalman filter (EKF) the robot's heading and its position relative to stair boundaries. A two-tiered controller consisting of both a centering and a heading module uses the estimates provided by the EKF to guide the robot upstairs. Results of field experiments with an *iRobot Packbot* verified that the proposed method could control the robot to perform the autonomous stair climbing task accurately and reliably.

In [11], a hierarchical control structure was developed for autonomous stair climbing of tracked mobile robots. The stair-climbing algorithm was developed on the basis of an intelligent sensor fusion technique, which integrates acoustic, visual and acceleration sensors. An over-arching broker was proposed to estimate the heading angle of a robot from the individual sensory inputs and send this information to a proportional-derivative (PD) path-tracking controller, which was employed to drive the robot upstairs. Experiments with the *Urbie* robot confirmed the robustness of the developed algorithm for climbing stairs in both dark and bright lighting conditions, and when there was no banister.

In [12], a sensor-based autonomous subtrack controller was developed to control swingable subtracks of rescue robots which are used to negotiate steps and uneven terrain. The controller allows an operator to manually control the main tracks, while the autonomous controller is used for the subtracks. LIDAR sensors on a robot provide terrain information to the controller for position control of the subtracks. The developed control strategy was verified via extensive field experiments conducted in the *RoboCup Rescue Robot League 2009* field and the *Tachikawa Regional Disaster Prevention Base*.

2.2 Self-correcting in Uneven Terrain

Flipping over has been identified to be one of the biggest challenges for rescue robots in USAR environments. To address this issue, safety ropes have been employed to help recover tracked rescue robots to the upright configuration. Although useful, safety ropes similar to tethers do get tangled in cluttered environments and cannot be retrieved without an intercession [1]. It was reported that during the WTC site trials, an operator had to pull on a robot tether approximately once per minute, in order to keep it from getting tangled [1].

To address the issue of robot tip-over stability, one promising solution is to use self-reconfigurable robots. Currently, reconfigurable rescue robots require that human operators mainly reconfigure the robots for different terrains, which can be a big burden for an operator who only has limited camera views [12]. Recently, the *iRobot Packbots* have been equipped with self-righting capabilities which allow the robots to use their manipulator arms to flip themselves over when they have toppled over in order to continue their missions [13]. Such self-righting capabilities can help a rescue robot regain its upright configuration while in a cluttered scene, however, they cannot prevent it from flipping over. Thus, it is desirable to develop control techniques that will be able to predict if a rescue robot will flip over and provide some precautionary actions to avoid this situation.

Autonomous tip-over prediction and prevention algorithms have recently been developed for tracked robots, i.e., [14, 15]. For example, in [14], robot track-stair interactions were analyzed and an online tip-over prediction algorithm was developed for a tracked mobile robot utilizing the forces generated at track-stair contact points. In [15], a real-time tip-over prevention algorithm was developed for a tracked mobile robot with a manipulator by balancing the load distribution via reconfiguring the mobile platform or adjusting the configuration of the manipulator. Though not developed or yet tested for urban disaster environments, the proposed tip-over prediction and prevention algorithms can also have potential application for rescue robots.

2.3 Performing SLAM in Rescue Environments

Rescue robots need to explore USAR environments while detecting hazards and searching for victims. To achieve these tasks, the robots can generate 3D maps of their environments and localize themselves as well as victims and objects of interest within these environments. These maps can then be used by human rescue workers for victim retrieval efforts. To address the robot localization and mapping problem, various SLAM approaches have been developed for USAR applications [16–20]. As rescue robots are still unable to autonomously navigate cluttered USAR environments, for these approaches, a human operator teleoperates a robot through such environments, while the robot autonomously builds a 3D map via SLAM using different perception techniques. For example, in [16], an extended information filter (EIF) based SLAM algorithm was proposed for building dense 3D maps in indoor USAR environments via the use of mobile rescue robots. The algorithm uses sensory information from both a *CSEM Swiss Ranger SR-2* range camera and a 2D camera. Data association between multiple images of the environment is performed using a scale invariant feature transformation (SIFT) feature detection and matching approach on 2D images obtained at different camera poses. Then RANSAC is applied to remove poor matches and outliers, and least square 3D point fitting is used to compute the initial value of the new camera pose. EIF SLAM is finally used to estimate camera poses and 3D feature positions. A 3D map of an environment is generated by combining the range images using the updated camera poses. Experiments were conducted with a teleoperated robot toy vehicle in a mock-up USAR arena. The results showed that the SLAM algorithm is suitable for mapping USAR arenas such as those used in *RoboCup Rescue* competitions.

In [17], an EKF-based SLAM technique was proposed for a robot to build a 3D map of rubble. The EKF-based SLAM technique uses sensory information from a laser range finder that can be mounted on a mobile robot in order to observe landmarks in the environment. Simulations were conducted using a virtualized rubble model in order to compare true and estimated (by SLAM) robot trajectories. The results showed that the SLAM approach is sensitive to measurement errors and initial errors, and at least three landmarks are needed to get good estimation results, hence, these factors need to be considered for implementing the SLAM technique in actual environments. In [18], the EKF-based SLAM technique was further implemented with the teleoperated snake-like rescue robot *Souryu-IV* in order to verify the use of a new laser range finder for 3D map building of a small physical USAR-like environment.

Our own group has also proposed the utilization of a SIFT-based 3D Visual SLAM method for robust and reliable landmark identification, and 3D mapping of cluttered and unknown USAR environments by mobile rescue robots [19]. Two-dimensional and 3D images of an environment are provided from a unique real-time mapping sensor that utilizes a digital fringe projection and phase shifting structured light technique. Landmark identification and matching from different 3D robot poses is obtained using a SIFT feature recognition and clustering technique. Once distinguishable landmarks are identified, they are used as input into an iterative closest point (ICP) based SLAM method. ICP-based SLAM is used to align 3D range information provided by the 3D sensor at different robot locations in order to create a corresponding 3D map of an environment. Experiments in a cluttered USAR-like scene with a teleoperated mobile robot verified the sensory system's robustness to generating real-time 2D and 3D images of such a scene, and the feasibility of the proposed landmark identification and SLAM methods in cluttered environments.

In [20], a hybrid SLAM approach was presented that combines both grid-based and topological maps. The proposed approach can work for single robot and multi-robot scenarios, where partial maps of an environment (provided by each robot) are merged for the latter case. The advantage of the method is that it can construct detailed maps without sacrificing scalability. The approach uses a layered data structure with a topological organization at the global level and small detailed metric maps at the local level. A laser range finder is used as the main sensor and scan matching is used to estimate a robot's displacement in a scene. The system was implemented in the computer-simulated emergency sites of the *Rescue Virtual Robot competition at RoboCup 2006*, where it supported up to 8 robots in a team. Experiments conducted with real robot data obtained from a publicly available data set show the potential of using the system in real-world situations.

3 Semi-autonomous Control

Semi-autonomous control integrates human teleoperation with robot autonomy. According to a robot's level of autonomy, semi-autonomous controllers developed for mobile robotic applications can be mainly categorized into fixed autonomy [48–50], and adjustable autonomy [21–24, 51–54]. Adjustable autonomy allows operators to choose the level of autonomy of the robot based on the task at hand -before deployment [21] or on the fly during deployment [22–24, 51–54].

3.1 Fixed Autonomy vs. Adjustable Autonomy

In robotic semi-autonomous control schemes with fixed autonomy, a robot focuses on low-level tasks such as, for example, terrain traversing, whereas the human operator is in charge of high-level control and supervisory tasks such as specifying the direction of travel [12]. Compared to fully teleoperated control of robots, such semi-autonomous controllers have been shown to be more effective for robot navigation and obstacle avoidance tasks [48–50]. Furthermore, they allow more neglect time for the operator, where neglect time is defined to be the amount of time the operator can abandon a robot, while attending to other robots in a team, reviewing remote visual information, or communicating with other rescue operators [48]. However, semi-autonomous controllers with fixed autonomy levels lack the flexibility needed for controlling robots in rubble-filled USAR environments. For example, if a robot gets physically stuck in a cluttered disaster environment, the operator may have to take over the control of low level operations in order to assist the robot. Similarly, high level control may be required from the robot controller when the operator cannot perform these tasks due to task overload, loss of situation awareness or communication dropout.

To address the limitations of fixed autonomy, semi-autonomous controllers with adjustable autonomy can be used for mobile robot applications [51–54]. For example, in [54], an operator is provided with the choice of different levels of autonomy; ranging between fully teleoperated control, which gives full control of the robot to the operator, to a shared mode where most of the low-level operations such as motor control and collision avoidance as well as high-level operations such as path planning are assigned to the autonomous robot controller, while the operator only specifies the overall goal tasks (e.g. search in a particular region). During target searching experiments conducted in a building, it was concluded that this latter mode resulted in the best performance in terms of minimizing the number of collisions and decreasing the workload of the operator when compared to lower levels of robot autonomy.

3.2 Adjustable Autonomy for Rescue Robots

Recently, a handful of semi-autonomous controllers with adjustable autonomy have been specifically developed for robot navigation and victim identification tasks in USAR applications [21–24].

In [21], a semi-autonomous control scheme was proposed consisting of four different modes, namely, *Tele Mode*, *Safe Mode*, *Shared Mode* and *Autonomous Mode*. In *Tele Mode*, the operator manually controls all robot movements. *Safe Mode* is similar to teleoperated control, in that all navigation and victim/object detection tasks are performed by the operator, however, the robot can intervene to prevent obstacle collisions. In *Shared Mode*, the robot provides the optimal path for navigation based on the operator's directional inputs. In *Autonomous Mode*, the robot manages navigation and obstacle avoidance, and operator intervention occurs only at the high-level tasks (i.e., defining a point to navigate to or searching a selected region). Although [21] provides the operator with the opportunity to set the level of autonomy of the controller beforehand, an operator is not able to change this level of autonomy on the fly during a search and rescue operation, which may be needed in unknown environments if either the robot or operator face a situation where they need assistance from each other during the mission.

To address this limitation, semi-autonomous control schemes have been developed to allow for on the fly adjustments of a robot's level of autonomy either by the operator [22] or the robot controller [23, 24]. In [22], a semi-autonomous control scheme was presented consisting of both a manual mode and a semi-autonomous mode. For the semi-autonomous mode, an operator was in charge of performing a semantic search of a scene which included dividing a large search area into smaller regions and ordering these regions by expected parameters such as proximity of region to the robot and the likelihood of a victim. The robot was in charge of handling the more routine and tedious systematic search, in which it searched for victims autonomously, alerted the operator when a potential victim was found and waited for operator confirmation. An additional category of search known as opportunistic search was also introduced, where a robot was able to operate on a reduced detection function on incoming data without distracting the operator or slowing the navigation control. Although certain tasks were assigned to both the operator and robot, the operator could take over the control of the robot at any given time during the operation.

In [23], three semi-autonomous control modes, *teleoperated*, *mixed-initiative* and *autonomous*, were proposed for task distribution between an operator and a rescue robot. The specific mode to implement was determined automatically by the robot based on the manner in which the operator was interacting with the overall system during a USAR operation. For example, the interaction was adjusted from operator-based to planning-based if the operator was idle for a certain period of time. The control modes were tested with a wheeled mobile robot in the *National Institute of Standard Technology* (NIST) yellow test arena which represents an indoor flat environment with minor structural damage. The results showed that in more complex settings, there was improved performance of the robot when mixed-initiative control was used over the fully teleoperated and autonomous control modes.

In [24], a semi-autonomous controller was presented for controlling a team of rescue agents in USAR environments. The system allowed an operator's commands to be weighted with the agents' own autonomous commands using a mediator. In addition to the mediator, an intervention recognition module was also utilized to identify situations during a rescue operation where human intervention is required such as: 1) when agents got stuck, 2) when agents got lost or unable to complete their goals, or 3) when a potential victim was found to verify the victim. Simulations were conducted in a simulated USAR scene using the Player/Stage simulation tool. Simulation results revealed that the developed semi-autonomous control approach had better performance than teleoperation and fully autonomous control in terms of area coverage and number of victims identified.

3.3 Learning-based Semi-autonomous Control for USAR

Learning techniques, in particular, reinforcement learning (RL) have previously been used in structured environments to address mobile robot navigation and obstacle avoidance problems in order for a robot to learn its own optimal behaviors by observing the outcome of its actions as it interacts with its environment [55–57]. To address the limitations of traditional RL techniques, in particular large search spaces and long learning times for complex large scale robotic applications, hierarchical reinforcement learning (HRL) techniques have been developed. HRL reduces the overall task of a robot into a set of individual sub-tasks for faster learning [58].

To date, there have only been a handful of cases reported where HRL has been used in mobile robotic applications, which include mobile robot navigation in a specially designed intelligent space [57], multi-agent cooperation for trash collection and automated guided vehicle scheduling [59], and robotic soccer [60]. Alternatively, our own work in this area uniquely focuses on using HRL to develop learning-based semi-autonomous controllers for rescue robots for cluttered USAR environments [25–28].

The learning-based semi-autonomous control architecture we have developed is shown in Fig. 1. The architecture consists of a robot sensors module which provides sensory information of the robot's environment to the *SLAM*, *HRI* (Human-Robot Interaction) *Interface* and *Deliberation* modules. This sensory information is used by the *SLAM* module to create a 3D global map of the robot's surroundings. The 3D map and sensory information are provided to the operator through the *HRI Interface* to allow for human control of the robot when needed. The *Deliberation* module utilizes the sensory information to determine the level of autonomy of the robot as well as the

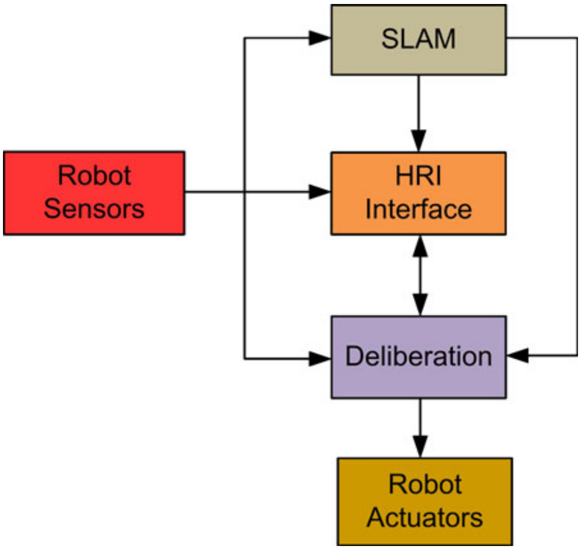
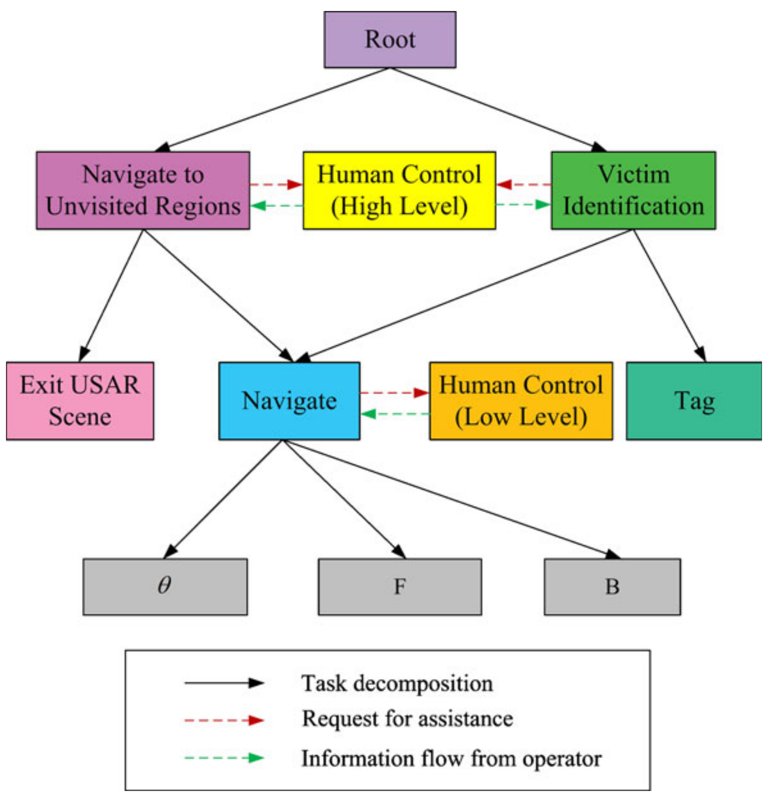


Fig. 1 A semi-autonomous control architecture for robotic USAR [26]

robot's actions using an HRL approach. The robot *actuators* module takes the robot actions provided by the *Deliberation* module and translates them into appropriate motor signals.

We have implemented a MAXQ HRL method within the context of our semi-autonomous controller for the robot exploration and victim identification problem. The objective is that the robot is able to learn from its own previous experiences and those of a human operator in order to effectively perform tasks in USAR environments. MAXQ decomposes a given Markov Decision Process (MDP) into a finite set of subtasks which define the MAXQ hierarchy and that can be learned concurrently [58]. The method requires little prior knowledge of an environment and can support state, temporal and subtask abstraction, making it suitable for unknown USAR applications. The MAXQ task hierarchy we have developed is depicted in Fig. 2. The *Root* task represents the overall USAR task, i.e., finding victims while exploring a cluttered USAR environment. This task is further decomposed into four individual subtasks, namely, *Navigate to Unvisited Regions*, *Navigate*, *Victim Identification* and *Human Control*. The purpose of the *Navigate to Unvisited Regions* subtask is to allow a robot to explore unvisited regions within disaster environments.

Fig. 2 MAXQ task graph for semi-autonomous control [28]



The *Exit USAR Scene* is used by the *Navigate to Unvisited Regions* subtask to trigger the end of exploration and to guide a rescue robot out of the USAR scene when exploration is completed. The *Navigate* subtask is used to perform local scene exploration using the primitive actions θ, F and B, which are defined to rotate the robot, and/or move the robot forward or backwards, respectively. The objective of the *Victim Identification* subtask is to identify victims in cluttered USAR scenes. The subtask can call the *Navigate* subtask in order to move the robot closer to a potential victim to make a positive identification. The primitive action, *Tag*, is used to physically locate a victim in the 3D map developed within the *SLAM* module of the architecture. The purpose of the *Human Control* subtasks is to pass over the control of the robot "on the fly" to the human operator in situations where the robot is unable to perform any of its sub-tasks autonomously. Unlike the conventional teleoperation modes reported in the literature, the operator's actions are provided to the corresponding modules in the architecture, allowing the robot to learn from the operator's actions. With the proposed control architecture, the transition from teleoperation to autonomous control can be achieved smoothly and gradually.

Experiments comparing the semi-autonomous controller to teleoperation control of a mobile robot in a rubble-filled USAR-like scene found that the former resulted in the exploration of larger areas with higher victim identification success rates. Furthermore, a questionnaire on the participants' experiences in operating the rescue robot revealed that the operators had better overall experience when using the semi-autonomous controller, as it improved their situation awareness as they did not have to complete all the tasks simultaneously on their own.

4 Control of Multi-robot Rescue Teams

USAR is a time critical task; namely, responders have approximately 72 hours to find trapped survivors, before the likelihood of finding victims still alive drops drastically [61]. As a result, teamwork, including human-robot cooperation and multi-robot coordination, become essential for rescue robots [30–40]. A typical motivation for multi-robot teams is to improve the efficiency of rescue robots that are currently mainly independently deployed and controlled. The introduction of multiple robots increases robustness through redundancy [42]. Moreover, limitations in individual robot payloads can be addressed; in particular, it would be more economical and easier to distribute the necessary hardware (i.e., sensors) for mission completion among multiple robots and then enable cooperation between these robots rather than integrate all necessary capabilities into a single robot [42].

While promising, the integration of multiple robots into a rescue team is a challenging task. Apart from cooperation between human operators and each individual rescue robot, robot coordination amongst the team is important in order to accomplish such tasks as scene exploration. Furthermore, task distribution amongst different rescue robots also needs to be defined so that the robots, whether homogeneous or heterogeneous, can work effectively together within the context of the team.

4.1 Human-Robot Teams

One of the main considerations in devising a team is to determine the ratio of human operators to robots. In [30], the effects of the number of robots controlled by an operator on the performance of a USAR task were investigated in simulations with 4–12 simulated UGVs. The simulations were performed in the USARSim environment. USARSim is a high-fidelity simulation of USAR robots and environments that is used as a research tool for investigating multi-robot coordination and HRI [62, 63]. Participants were asked to search for victims while controlling varying numbers of robots in a large simulated maze-like environment. The results showed that task performance increased when going from four to eight controlled robots but decreased when going from eight to twelve robots. Furthermore, workload increased with increasing number of robots to be controlled.

The human operator/robot ratio in a multi-robot team was investigated in [31] within the context of the *RoboCup Rescue Real Robot League*. It was found that cooperative task achievability increases with an increase in the number of robots. For example, an operator made faster decisions with less hesitation while controlling more than one robot. In addition, the effectiveness of task sharing increases when the operator/robot ratio increases. It was also found that deployment of a multi-robot multi-operator system can help reduce the operators' psychological pressure even if the operator/robot ratio is one-to-one.

In [32], the USARSim simulation platform was used again, this time to investigate the challenge of controlling multiple robots with human teams in USAR environments. Twenty-four simulated mobile robots were used to explore and search for victims in a large USAR virtual environment. The robots were controlled by teams of two people. Team performance was evaluated and compared for scenarios where the robots navigated the environment autonomously or used operator supplied waypoints. Two different modes of operator control were implemented; *assigned robot condition* and *pool robot condition*. In the former mode, each operator was assigned control of twelve separate robots, and in the latter mode the two operators shared control of all the robots. From the simulation results, the operators using autonomous path planning and navigation algorithms found more victims and marked their locations more accurately in both the *assigned robot condition* and *shared pool condition*. Comparing the regions explored, there was a substantial advantage for the autonomous exploration over manual exploration in the *shared pool condition*, while the difference was negligible in the *assigned robot condition*.

In addition to teams with human operators and robots, there has also been some work that has focused on devising teams with human rescue workers and robots. Such teams are effective when there exist some regions that are only accessible by rescue robots and others that can be visited by rescue workers [33]. In particular, in [33], the problem of jointly performing RFID-based SLAM by robots and firefighters was explored. Information sharing between the two was achieved by using RFID tags placed in the environment. Namely, human poses were tracked by analyzing acceleration patterns to determine footsteps, and robot poses were tracked using wheel odometry and IMU data. Then both the humans and robots estimated the distances between RFID tags by pose tracking and this information was sent to a central station, where a joint global graph was constructed via RFID-based SLAM. Human-robot team experiments were conducted on the campus of the *University of Freiburg*, and the results showed that information sharing between humans and robots allows for the correction of their individual paths globally.

4.2 Multi-robot Cooperation

Cooperation between multiple robots reduces the need for operators to control each robot independently and can also improve task efficiency as each robot in the team is assigned a different but complementary task. In [34], a framework for a single operator to deploy a team of heterogeneous (ground and aerial) robots for urban surveillance was presented. The framework consisted of a number of different enabling technologies that allowed a human operator to deploy the robot team to search and locate a human target by assigning different tasks to team members and monitoring the performance of the mission through a single integrated command and control interface. In particular, to realize cooperative search, identification and localization, an active sensor network architecture was implemented based on the idea that the value of a sensing action is marked by its associated reduction in uncertainty and that mutual information formally captures the utility of sensing actions in these terms. Local controllers implemented on each robotic sensor platform were used to direct the robots as well as their sensors according to a mutual information gradient approach, allowing individual robots to drive in directions that maximize their information gain locally. Experiments were conducted at the *McKenna Military Operations on Urban Terrain* site in *Fort Benning*, *Georgia*, where a team of aerial and ground robots was deployed to patrol an urban village and search for and localize human targets. The robots autonomously deployed themselves to search for a target of interest after receiving a universal commence signal from the base station. The deployment decisions dedicated to maintaining network connectivity were made automatically, and the human operator was involved in identifying targets of interest. In the experiments, aerial robots conducted an initial coarse search of the region, determined potential target locations and generated maps that were used to design navigation controllers and plan missions for the team. Then unmanned ground vehicles were deployed to conduct a more localized search and identify the targets on the basis of the aerial robots' initial assessment. The experimental results verified the effectiveness of the integration of a team of different robots with unique physical capabilities and sensors.

The deployment of a multi-robot rescue team, for search and rescue missions in indoor environments was discussed in [35]. In the proposed approach, a micro ground vehicle (MGV) was used for environment mapping and a micro aerial vehicle (MAV) and a MGV were implemented for search and localization of survivors. Two other MGVs were also utilized as a backup robot team. Initially, an MGV performs simultaneous localization and mapping, while one MAV searches the area already explored to find the target. An MGV then approaches the target to perform a rescue or security task. If more support is needed, the central control station deploys the back-up team to provide assistance. Experimental results in structured indoor environments demonstrated that the multi-robot team was able to complete the search task faster than a single rescue robot.

In [36], virtual pheromone was proposed for coordinating the actions of a robot swarm to facilitate quick survivor localization in USAR scenes. In the proposed approach, a swarm of mini-robots can be deployed in a building. Upon detecting a survivor, a robot emits a message signaling the discovery, which is relayed locally between neighboring robots. The message propagates along unobstructed paths, producing a "virtual pheromone" gradient, which can guide human rescuers or larger victim manipulating robots to a survivor. The concept design was preliminary demonstrated with a swarm of 20 custom-made pheromone robots, which were equipped with a specially designed infrared communication ring that facilitates both inter-robot communication and obstacle detection. The proposed concept provides a promising solution for robot peer-to-peer messaging in order to accomplish many coordinated activities without a centralized controller.

4.3 Multi-robot Exploration

Within the context of multi-robot cooperation for USAR environments, a well investigated and important task for robot teams is cooperative exploration. For example, in [37], both Robotic Particle Swarm Optimization (RPSO) and Robotic Darwinian Particle Swarm Optimization (RDPSO) were extended for a distributed exploration task by a team of simulated robots. RPSO consists of a group of robots that collectively search an environment for the global optimum (which can be defined as the number of victims), while performing obstacle avoidance. Each robot in the group is characterized by its pose and performance. RDPSO consists of multiple swarms of robots where each swarm individually performs just like an RPSO algorithm in search for the solution, however, with rules governing the collection of swarms that are designed to simulate natural selection (i.e., providing rewards or punishments to a swarm). Simulation results demonstrated the application potential of the developed algorithms in multi-robot search and exploration problems, with the RDPSO demonstrating better performance than the RPSO.

The control strategies of the winning teams of the *RobotCup Rescue Virtual Competition* in 2006 were summarized in [38]. The teams used the USARSim framework for the competition environment. The simulated environment consisted of roads, a multistory office building, a park, and several accident scenarios. The competition was won by the *Rescue Robots Freiburg* (*RRFreiburg*) team from the *University of Freiburg* in Germany. They were the only team to explicitly include multi-robot cooperative exploration in their strategy, whereas the other teams adopted a selfish approach that did not share mapping or victim distribution information between the team members until the end of the mission. Namely, the *RRFreiburg* team developed a multi-robot cooperative-exploration approach that allowed mobile robots to plan and explore paths based on a local view of the environment through the use of indirect communication via radio frequency identification (RFID) tags which were deployed autonomously by the robots. The robots continuously planned their paths using their local knowledge of the environment, which was maintained within an occupancy grid. In order to avoid collisions with teammates, the local displacements between robots were synchronized via RFID tags. To enable efficient multi-robot exploration, the targets for the path planner were selected in such a way as to minimize overlapping of explored areas by utilizing a utility function frontier-based exploration approach which considered an inertial factor and the density of the paths in the vicinity of a frontier. To avoid revisiting previously explored areas by a single robot or the other members in the group, each robot stored the memory of all perceptions from its past to the nearest RFID tag.

A role-based multi-robot exploration technique was proposed in [39], in which robots take on one of two roles; explorer or relay. Explorers search the far reaches of the environment using frontier exploration, and periodically return to meet with relays at previously agreed-upon rendezvous points to provide them with information about the environment. The relays in turn carry this information back to a central command center. A dynamic team hierarchy using the role swap rule allows robots to change their roles during exploration in order to obtain shorter travel paths, and hence, faster exploration. Comparison of the exploration technique in simulations using varying environments and team sizes showed improvement in inter-robot awareness and team responsiveness as compared to greedy frontier-based and static role-based exploration techniques, but did not lead to faster exploration times. The authors suggest that the former are important characteristics in search and rescue where instant control over robots is highly desirable.

4.4 Multi-robot Task Allocation

When deploying a multi-robot rescue team to solve the search coverage problem in USAR environments, one must consider the real-time allocation of tasks between the team members. While adaptive task allocation algorithms have not been specifically investigated for search and rescue, a handful of techniques have been developed for general mobile sensor agent and robot applications, i.e., [40, 64–68]. Task allocation methods for multi-robot systems can be categorized into centralized and decentralized approaches. Centralized approaches focus on using a single central task planner to determine and distribute tasks to an entire robot team, while decentralized methods focus on each team member having its own onboard task allocation planner. With respect to centralized approaches, in [40], an equilibrium task allocation method and a hybrid binary particle swarm optimization (HBPSO) task allocation method were developed and compared for robotic vehicles. These task allocation methods have been proposed for such applications as surveillance, and search and rescue. For the equilibrium task allocation method, robots are randomly assigned to search regions, and if more than one robot is assigned to a region, a gradient-descent based method is utilized to equalize the time-loading between the robots. The HBPSO task allocation method consists of two stages, where the first stage assigns a robot to a region or regions, and the second stage modifies the size of the region(s) in order to balance the time-load across all robots. Simulation results in different scenarios showed that the equilibrium task allocation method demonstrated better performance over the HBPSO method for simple scenarios, in terms of the time spent to find an efficient task allocation, and the task allocation quality. Experiments conducted at *Boeing's Vehicle Swarm Technology Lab* using the equilibrium task allocation technique with two quadrotors and two tanks validate the ability of the robots to complete search area coverage.

In [64], a dynamic task allocation and controller design methodology was proposed for medium-sized cooperative robot teams. The system can be applied to numerous applications including warehouse patrolling, service robotics and space exploration. A fuzzy-logic-based utility function was used to determine if a robot could perform a set of tasks by taking into account a robot's operational capabilities. In the proposed design, the robot team was modeled as a discrete event system, while each robot was represented by a finite state automaton model. The control methodology is based on supervisory control theory that utilizes a limited look-ahead policy that enables/disables events in the system in real time based on the evaluation of the utility function and robot availability. Simulation results on a warehouse patrolling application scenario demonstrated that a limited look-ahead control policy provided a computationally efficient approach for task allocation.

The main advantage of the aforementioned centralized approaches are that they allow for smaller and more cost effective robots, as the heavy processing requirements are placed within a central station rather than on the robots themselves [65]. However, since the robots would have to always communicate with a fixed location, the possible mission ranges that a team of robots can handle is limited, as well this creates a single potential point of failure for a mission [65]. To address these limitations, decentralized task assignment algorithms have been proposed to solve the task allocation problem, e.g. [65, 66].

In [65], two decentralized algorithms were presented to address the task allocation problem for coordinating a fleet of autonomous vehicles utilizing auctioning with greedy heuristics and a conflict resolution protocol based on consensus on the winning bid values over the team. For the single-assignment problem, in which at most one task can be assigned to a single agent, a consensus-based auction algorithm (CBAA) was proposed. The CBAA algorithm consisted of two phases: 1) an auction process, and 2) a consensus process. The auction process consisted of each agent placing a bid on a task and the formulation of a winning bids list. In the latter phase, a consensus strategy was used by the agents to converge on the list of winning bids and determine the winner. The CBAA algorithm was extended to the multi-assignment problem in which a sequence of multiple tasks was assigned to each agent in the fleet through the development of the consensus-based bundle algorithm (CBBA). In the first phase of CBBA, each agent creates a bundle of tasks and updates the bundle as the assignment process continues. In the second phase, conflict resolution rules are used to assist in determining winners of tasks. The developed task allocation algorithms can be used to produce conflict-free feasible solutions that are robust to both inconsistencies in the situational awareness across the fleet and variations in the communication network topology. Simulations showed good performance and fast convergence of the algorithms compared to other existing methods.

In [66], a new decentralized task assignment algorithm known as heterogeneous robots consensus-based allocation (HRCA) was presented for heterogeneous robotic networks. The first stage of HRCA is similar to CBBA where a consensus-based auction approach is used consisting of each robot filling its bundle with tasks based on its own skills and a conflict resolution process is implemented to obtain consensus among the robots. The second stage consists of a bundle check to ensure that robots are not overloaded. Simulations show that HRCA performs better for heterogeneous networks over extensions of the CBBA algorithm.

Decentralized task allocation approaches have many advantages over centralized approaches, such as low communication demands and robustness with respect to communication failures [67]. However, decentralized approaches may produce suboptimal solutions when the global cost function is considered [67]. To take advantage of both centralized and decentralized approaches, some hybrid approaches have also been proposed. For example, in [68], a hybrid market-based approach was developed for complex task allocation in mobile surveillance systems. The approach can be used to solve the multi-robot task allocation problem by assigning a set of surveillance tasks to a set of mobile sensing agents in order to maximize overall expected performance, while taking into account the priorities of the tasks and the skill level of the agents. Both centralized and hierarchical (hybrid) allocation algorithms were examined as winner determination strategies for different levels of allocation, and for static and dynamic search tree structures. In the centralized allocation algorithm, an auctioneer holds a series of auctions to allocate the surveillance tasks to the mobile sensing agents in order to maximize the system utility, whereas for hierarchical allocation, the tasks are allocated initially to the mobile sensing agents via a central auctioneer, and then each mobile sensing agent can hold auctions in rounds for the tasks it wins in the initial auction. Simulations conducted in 2D environments

Table 1 Summary of control schemes developed for rescue robots in USAR

Study	Task	Level of robot autonomy	Type of cooperation	Make up of team	Simulation/experimental environments
[1–4, 6]	Search for survivors in disaster scene	Teleoperation	Operator control of robot	Single operator & single robot	Field experiments in real USAR sites, including WTC and Tohoku earthquake and tsunami
[7]	Examine operator situation awareness & search team interaction	Teleoperation	Operator control of robot	Single operator & single robot	USAR disaster response training exercise
[10, 11]	Stair climbing	Low-level autonomy	Operator navigates robot to stairway & robot autonomously climbs stairs	Single operator & single robot	Indoor and outdoor stairwells
[12]	Traversing stairs and uneven terrain	Low-level autonomy	Operator controls main tracks of a tracked robot & swingable subtracks are controlled autonomously	Single operator & single robot	RoboCup Rescue 2009 field and Tachikawa Regional Disaster Prevention Base
[16–19]	SLAM	Teleoperation & autonomous SLAM	Operator control of robot & robot builds 3D map autonomously	Single operator & single robot	Experiments in USAR-like environments
[20]	SLAM	Teleoperation & autonomous SLAM	Operator control of robots & multiple robots build map autonomously	Single operator & multiple robots	Simulations in USARSim & experiments with real robot data from the Radish robotics data repository
[21–23]	Navigation, victim detection or reconnaissance	Semi-autonomous control with adjustable autonomy	Operator and robot share rescue tasks	Single operator & single robot	Lab experiments and tests in NIST yellow arena
[24]	Exploration & victim identification	Semi-autonomous control with adjustable autonomy	Operator and robots share rescue tasks	Single operator & multiple robots	Player/Stage simulation environment
[25–28]	Exploration & victim identification	Learning based semi-autonomous control	Robot performs exploration and victim identification, and asks for help from operator when needed	Single operator & single robot	Simulations & experiments in USAR-like scenes
[30–32]	Examine operator/robot ratio in USAR teams	Teleoperation	Cooperation between multiple operators to control multiple robots	Multiple operators & multiple robots	USARSim simulation environment

Table 1 (continued)

Study	Task	Level of robot autonomy	Type of cooperation	Make up of team	Simulation/experimental environments
[33]	SLAM	Teleoperation & autonomous map building at a central station	Cooperation between human rescue workers and robots to jointly build maps of USAR environments	Multiple human workers & multiple robots	Experiments on the campus of the University of Freiburg
[34]	Search for human targets	Semi-autonomous control	Robots were deployed autonomously, human operator involved in identifying human targets	Single operator & multiple heterogeneous (ground and aerial) robots	Field experiments at McKenna Military Operations
[35]	Search and rescue in indoor environments	Autonomous control	Cooperation between aerial and ground robots for environment mapping, search and localization of survivors	Multiple heterogeneous robots (MAV and MGVs)	Experiments on a floor in a building
[36]	Survivor localization in USAR environments	Autonomous control	Coordinating the actions of a robot swarm	Multiple mobile robots	Experiments in an open space
[37]	Search and exploration	Autonomous control	Multi-robot cooperative exploration	Multiple mobile robots	Simulation in Matlab environment
[38]	Exploration & victim identification	Autonomous control	Multi-robot cooperative exploration	Multiple mobile robots	USARSim simulation environment
[39]	Exploration	Autonomous control	Cooperation between explorer robots and relay robots	Multiple robots	MRESim simulation environment using a floor plan from the Robotics Data Set Repository
[40]	Task allocation for robot exploration	Autonomous control	Multi-robot cooperative exploration	Multiple heterogeneous robots (quadrotors & tanks)	Simulations and experiments in Boeing's Vehicle Swarm Technology Lab

with varying numbers of areas and sensing agents showed that the hierarchical dynamic tree task allocation approach had better performance than all the other techniques, especially in complex surveillance operations where a large number of agents were used to scan a large number of areas.

5 Summary

In this paper, we provide a survey of control methodologies that have been developed for rescue robots operating in USAR missions. Table 1 summarizes the research work in this area with respect to the main topics covered. While the majority of rescue robots that have been deployed in real-world USAR scenes have been teleoperated, there has been great interest in the past several years to provide some level of autonomy to the robots themselves. For example, some researchers have focused on improving the low-level control of rescue robots by equipping them with the ability to autonomously climb stairs [10, 11] and navigate over uneven terrain [12]. Others have focused on the development of high-level semi-autonomous control strategies that enable sharing of exploration and victim identification tasks between a rescue robot and an operator to minimize operator workload [21–28]. Teamwork is a crucial component of rescue robot deployment, whether it be human-robot cooperation [30–33] or multi-robot coordination [34–39]. Furthermore, real-time task allocation techniques are needed, i.e. [40], to distribute tasks to rescue robots in a team in order to have multiple robots work effectively together to achieve the rescue tasks at hand. The development and incorporation of SLAM algorithms will also allow a single robot [16–19], robot teams [20] and human-robot teams [33] to develop 3D maps of USAR scenes in order to locate victims within these cluttered environments.

While robot exploration and victim identification have been the main focus of rescue robot applications in the past decade, researchers have also recently focused on the task of victim manipulation [69, 70]. Namely, a future direction in the area of rescue robotics is now including robots that can help maneuver and transport trapped victims out of rubble. To date, victim manipulation has been avoided due to the fear of further injuring victims or even causing death [70]. With this new direction of research comes the need for sophisticated robotic control techniques for manipulation and coordination between teams of robots in order to remove rubble off of trapped victims and carefully lift and transport these victims to safety. Hence, overall the next few years will be an exciting time for researchers working in the area of rescue robotics as there are still a number of challenges to address.

References

1. Murphy, R.: Activities of rescue robots at the world trade center from 11–21 September 2001. IEEE Robot. Autom. Mag. **11**(3), 50–61 (2004)
2. Murphy, R., Tadokoro, S., Nardi, D., Jacoff, A., Fiorini, P., Choset, H., Erkmen, A.: Search and rescue robotics. In: Siciliano, B., Khatib, O. (eds.) Springer Handbook of Robotics, pp. 1151–1173. Springer, Berlin (2008)
3. Guizzo, E.: Japan earthquake: Robots help search for survivors. IEEE Spectrum (2011). URL http://spectrum.ieee.org/automaton/robotics/industrial-robots/japan-earthquake-robots-help-search-for-survivors
4. Guizzo, E.: Japan earthquake: more robots to the rescue. IEEE Spectrum (2011). URL http://spectrum.ieee.org/automaton/robotics/industrial-robots/japan-earthquakemore-robots-to-the-rescue
5. Casper, J., Micire, M., Murphy, R.: Issues in intelligent robots for search and rescue. In: Proceedings of SPIE, pp. 292–302 (2000)
6. Casper, J., Murphy, R.: Human-robot interactions during the robot-assisted urban search and rescue response at the World Trade Center. IEEE Trans. Syst. Man Cybern. Part B **33**(3), 367–385 (2003)
7. Burke, J., Murphy, R., Coovert, M., Riddle, D.: Moonlight in Miami: a field study of human-robot interaction in the context of an urban search and rescue disaster response training exercise. Hum. Comput. Interact. **19**, 85–116 (2004)
8. Nordqvist, C.: 9/11 ten years on -the health effects on rescue workers. Medical News Today (2011). URL http://www.medicalnewstoday.com/articles/234030.php
9. Chief: Problems and symptoms that a rescue worker may experience thru helping. Rescue Workers (2001). URL http://www.rescue-workers.com/problems.html
10. Mourikis, A., Trawny, N., Roumeliotis, S., Helmick, D., Matthies, L.: Autonomous stair climbing for tracked vehicles. Int. J. Robot. Res. **26**(7), 737–758 (2007)
11. Steplight, S., Egnal, G., Jung, S., Walker, D., Taylor, C., Ostrowski, J.: A mode-based sensor fusion approach to robotic stair-climbing. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1113–1118 (2000)

12. Okada, Y., Nagatani, K., Yoshida, K., Tadokoro, S., Yoshida, T., Koyanagi, E.: Shared autonomy system for tracked vehicles on rough terrain based on continuous three-dimensional terrain scanning. J. Field Robot. **28**(6), 875–893 (2011)
13. iRobot: User assist package for iRobot 510 Packbot (2012). URL http://www.irobot.com/~/media/Files/Robots/Defense/PackBot/iRobot-PackbotUAP.ashx
14. Liu, Y., Liu, G.: Track-stair interaction analysis and online tip-over prediction for a self-reconfigurable tracked mobile robot climbing stairs. IEEE/ASME Trans. Mech. **14**(5), 528–538 (2009)
15. Liu, Y., Liu, G.: Interaction analysis and online tip-over avoidance for a reconfigurable tracked mobile modular manipulator negotiating slopes. IEEE/ASME Trans. Mech. **15**(4), 623–635 (2010)
16. Ellekilde, L.-P., Huang, S., Miro, J.V., Dissanayake, G.: Dense 3-D map construction for indoor search and rescue. J. Field Robot. **24**(1/2), 71–89 (2007)
17. Yokokohji, Y., Kurisu, M., Takao, S., Kudo, Y., Hayashi, K., Yoshikawa, T.: Constructing a 3-D map of rubble by teleoperated mobile robots with a motion cancelling camera system. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3118–3125 (2003)
18. Kurisu, M., Muroi, H., Yokokohji, Y., Kuwahara, H.: Development of a laser range finder for 3D map-building in rubble—installation in a rescue robot. In: Proceedings of IEEE International Conference on Mechatronics and Automation, pp. 2054–2059 (2007)
19. Zhang, Z., Nejat, G., Guo, H., Huang, P.: A novel 3D sensory system for robot-assisted mapping of cluttered urban search and rescue environments. Intell. Serv. Robot. **4**, 119–134 (2011)
20. Pfingsthorn, M., Slamet, B., Visser, A.: A scalable hybrid multi-robot SLAM method for highly detailed maps. In: RoboCup 2007: Robot Soccer World Cup XI, pp. 457–464. Springer Verlag, Berlin, Heidelberg (2008)
21. Bruemmer, D., Boring, R., Few, D., Marble, J., Walton, M.: I call shotgun!: an evaluation of mixed-initiative control for novice users of a search and rescue robot. In: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, pp. 2847–2852 (2004)
22. Murphy, R., Sprouse, J.: Strategies for searching an area with emi-autonomous mobile robots. In: Proceedings of Robotics for Challenging Environments, pp. 15–21 (1996)
23. Finzi, A., Orlandini, A.: Human-robot interaction through mixed-initiative planning for rescue and search rovers. In: Bandini, S., Manzoni, S. (eds.) Springer Advances in Artificial Intelligence, vol. 3673, pp. 483–494. Springer, Berlin/Heidlberg (2005)
24. Wegner, R.W., Anderson, J.: An-agent based support for balancing teleoperation and autonomy in urban search and rescue. Int. J. Robot. Autom. **21**(2), 120–128 (2006)
25. Doroodgar, B., Ficocelli, M., Mobedi, B., Nejat, G.: The search for survivors: cooperative human-robot interaction in search and rescue environments using semi-autonomous robots. In: Proceedings of IEEE International Conference on Robotics and Automation, pp. 2858–2863 (2010)
26. Doroodgar, B., Nejat, G.: A hierarchical reinforcement learning based control architecture for semi-autonomous rescue robots in cluttered environments. In: Proceedings on IEEE International Conference on Automation Science Engineering, pp. 948–953 (2010)
27. Liu, Y., Nejat, G., Doroodgar, B.: Learning based semi-autonomous control for robots in urban search and rescue. In: Proceedings of IEEE International Symposium on Safety, Security and Rescue Robotics (2012)
28. Doroodgar, B.: A learning-based semi-autonomous control architecture for robotic exploration of search and rescue environments. Master's thesis, Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Canada (2011)
29. Cavallin, K., Svensson, P.: Semi-autonomous teleoperated search and rescue robot. Master's thesis, Department of Computing Science, Umea University, Sweden (2009)
30. Velagapudi, P., Scerri, P., Sycara, K., Wang, H., Lewis, M., Wang, J.: Scaling effects in multi-robot control. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2121–2126 (2008)
31. Sato, N., Matsuno, F., Yamasaki, T., Kamegawa, T., Shiroma, N., Igarashi, H.: Cooperative task execution by a multiple robot team and its operators in search and rescue operations. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1083–1088 (2004)
32. Lewis, M., Wang, H., Chien, S., Scerri, P., Velagapudi, P., Sycara, K., Kane, B.: Teams organization and performance in multi-human/multi-robot teams. In: Proceedings of IEEE International Conference on Systems Man and Cybernetics, pp. 1617–1623 (2010)
33. Kleiner, A., Dornhege, C., Sun, D.: Mapping disaster areas jointly: RFID-coordinated SLAM by humans and Robots. In: Proceedings of IEEE International Workshop on Safety, Security and Rescue Robotics, pp. 1–6 (2007)
34. Hsieh, M., Cowley, A., Keller, J., Chaimowicz, L., Grocholsky, B., Kumar, V., Taylor, C., Endo, Y., Arkin, R., Jung, B., Wolf, D., Sukhatme, G., Mackenzie, D.: Adaptive teams of autonomous aerial and ground robots for situation awareness. J. Field Robot. **24**(11), 991–1014 (2007)
35. Luo, C., Espinosa, A., Pranantha, D., Gloria, A.: Multi-robot search and rescue team. In: Proceedings of IEEE International Symposium on Safety, Security and Rescue Robotics, pp. 296–301 (2011)
36. Payton, D., Estkowski, R., Howard, M.: Pheromone robotics and the logic of virtual pheromones. In: Sahin, E., Spears, W. (eds.) Swarm Robotics WS2004 (LNCS), vol. 3342, pp. 45–57. Springer Verlag, Berlin, Heidelberg (2005)
37. Couceiro, M., Rocha, R., Ferreira, N.: A novel multi-robot exploration approach based on particle swarm optimization algorithms. In: Proceedings of IEEE International Symposium on Safety, Security and Rescue Robotics, pp. 327–332 (2011)

38. Balakirsky, S., Carpin, S., Kleiner, A., Lewis, M., Visser, A., Wang, J., Ziparo, V.: Towards heterogeneous robot teams for disaster mitigation: results and performance metrics from robocup rescue. J. Field Robot. **24**(11), 943–967 (2007)
39. de Hoog, J., Cameron, S., Visser, A.: Dynamic team hierarchies in communication-limited multi-robot exploration. In: Proceedings of IEEE International Workshop on Safety Security and Rescue Robotics, pp. 1–7 (2010)
40. Meuth, R., Saad, E., Wunsch, D., Vian, J.: Adaptive task allocation for search area coverage. In: Proceedings of IEEE International Conference on Technologies for Practical Robot Applications, pp. 67–74 (2009)
41. Nourbakhsh, I., Sycara, K., Koes, N., Yong, M., Lewis, M., Burion, S.: Human-robot teaming for search and rescue. In: Pervasive Computing, pp. 72–78 (2005)
42. Wong, C., Seet, G., Sim, S.: Multiple-robot systems for USAR: key design attributes and deployment issues. Int. J. Adv. Robot. Syst. **8**(1), 85–101 (2011)
43. Hsieh, M., Cowley, A., Kumar, V., Tayloar, C.: Maintaining network connectivity and performance in robot teams. J. Field Robot. **25**(1), 111–131 (2008)
44. Tardioli, D., Mosteo, A., Riazuelo, L., Villarroel, J., Montano, L.: Enforcing network connectivity in robot team missions. Int. J. Robot. Res. **29**(4), 460–480 (2010)
45. Woods, D., Tittle, J., Feil, M., Roesler, A.: Envisioning human-robot coordination in future operations. IEEE Trans. Syst. Man Cybern. Part C **34**(2), 210–218 (2004)
46. Murphy, R., Casper, J., Hyams, J., Micire, M., Minten, B.: Mobility and sensing demands in USAR. In: Proceedings of 26th Annual Conference of IEEE Industrial Electronics Society, pp. 138–142 (2000)
47. Casper, J., Murphy, R.: Workflow study on human-robot interaction in USAR. In: Proceedings in IEEE International Conference on Robotics and Automation, pp. 1997–2003 (2002)
48. Crandall, J., Goodrich, M.: Characterizing the efficiency of human robot interaction: a case study of shared-control teleoperation. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1290–1295 (2002)
49. Shen, J., Ibanez-Guzman, J., Ng, T., Chew, B.: A collaborative-shared control system with safe obstacle avoidance capability. In: Proceedings of IEEE Conference on Robotics, Automation and Mechatronics, pp. 119–123 (2004)
50. Perrin, X., Chavarriaga, R., Colas, F., Siegward, R., Millan, J.: Brain-coupled interaction for semi-autonomous navigation of an assistive robot. Robot. Auton. Syst. **58**(12), 1246–1255 (2010)
51. Law, C., Xu, Y.: Shared control for navigation and balance of a dynamically stable robot. In: Proceedings of IEEE International Conference on Robotics and Automation, pp. 1985–1990 (2002)
52. Tang, H., Cao, X., Song, A., Guo, Y., Bao, J.: Human-robot collaborative teleoperation system for semi-autonomous reconnaissance robot. In: Proceedings of IEEE International Conference on Mechatronics and Automation, pp. 1934–1939 (2009)
53. Goodrich, M., Olsen, J.D., Crandall, J., Palmer, T.: Experiments in adjustable autonomy. Tech. rep., Computer Science Department, Brigham Young University (2001). URL http://faculty.cs.byu.edu/mike/mikeg/papers/IJCAI01.pdf
54. Marble, J., Bruemmer, D., Few, D.: Lessons learned from usability tests with a collaborative cognitive workspace for human-robot teams. In: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, pp. 448–453 (2003)
55. Yu, H., Liu, S., Liu, J.: A new navigation method based on reinforcement learning and rough sets. In: Proceedings of International Conference on Machine Learning and Cybernetics, pp. 1093–1098 (2008)
56. Watkins, C.: Learning from delayed rewards. Ph.D. thesis, Cambridge University, Cambridge, UK (1989)
57. Jeni, L., Istenes, Z., Szemes, P., Hashimoto, H.: Robot navigation framework based on reinforcement learning for intelligent space. In: Proceedings of International Conference on Human System Interactions, pp. 761–766 (2008)
58. Dietterich, D.: Hierarchical reinforcement learning with MAXQ value function decomposition. J. Artif. Intell. Res. **13**, 227–303 (2000)
59. Makar, R., Mahadevan, S., Ghavamzadeh, M.: Hierarchical multi-agent reinforcement learning. In: Proceedings of International Conference on Autonomous Agents, pp. 246–253 (2001)
60. Stone, P., Sutton, R.: Scaling reinforcement learning toward robocup soccer. In: Proceedings of International Conference on Machine Learning, pp. 537–544 (2001)
61. Voyles, R., Choset, H.: Editorial: search and rescue robots. J. Field Robot. **25**(1), 1–2 (2008)
62. Wang, J., Lewis, M., Gennari, J.: A game engine based simulation of the NIST urban search and rescue arenas. In: Proceedings of the 2003 Winter Simulation Conference, pp. 1039–1045 (2003)
63. Sourceforge: Usarsim. (2012). URL http://sourceforge.net/apps/mediawiki/usarsim/
64. Tsalatsanis, A., Yalcin, A., Valavanis, K.: Dynamic task allocation in cooperative robot teams. Robotica **30**(5), 721–730 (2012)
65. Choi, H.L., Brunet, L., How, J.P.: Consensus-based decentralized auctions for robust task allocation. J. Field Robot. **25**(4), 912–926 (2009)
66. Di Paola, D., Naso, D., Turchiano, B.: Consensus-based robust decentralized task assignment for heterogeneous robot networks. In: Proceedings of American Control Conference, pp. 4711–4716 (2011)
67. Zhang, K., Collins Jr. E. G. , Shi, D.: Centralized and distributed task allocation in multi-robot teams via a stochastic clustering auction. ACM Trans. Auton. Adap. **7**(2), Article 21 (2012)

68. Khamis, A., Elmogy, A., Karray, F.: Complex task allocation in mobile surveillance systems. J. Intell. Robot. Syst. **64**, 33–55 (2011)
69. Iwano, Y., Osuka, K., Amano, H.: Posture manipulation for rescue activity via small traction robots. In: Proceedings of 2005 IEEE International Workshop on Safety, Security and Rescue Robots, pp. 87–92 (2005)
70. Yim, M., Laucharoen, J.: Towards small robot aided victim manipulation. J. Intell. Robot. Syst. **64**(1), 119–139 (2011)