2017 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS2017)
5-7 Oct. 2017, Ottawa, Canada

Robot Exploration in Unknown Cluttered Environments When Dealing with Uncertainty

Farzad Niroui, *Student Member, IEEE*, Ben Sprenger, and Goldie Nejat, *Member, IEEE*

***Abstract*—The use of autonomous robots in urban search and rescue (USAR) missions has many potential benefits in terms of assisting rescue workers and increasing efficiency in these time-critical environments. However, the cluttered and unknown nature of these environments introduces uncertainty in both the sensing and actuation capabilities of a rescue robot. Such uncertainty has not been directly incorporated into the modeling of the USAR problem for existing robots. In this paper, we present the novel use of a partially observable Markov Decision Process (POMDP) method which directly incorporates uncertainty within the decision-making layer of the controller for a rescue robot. A hierarchical task structure is used to decompose the overall exploration and victim identification task of a robot into smaller subtasks. These subtasks are modeled as POMDPs taking into account sensory and actuation uncertainty. Our proposed approach was tested in numerous experiments in unknown and cluttered USAR-like environments. The results should that the approach was able to successfully explore the environments and find victims, while dealing with sensor and actuator uncertainty.**

***Keywords*—Urban Search and Rescue, Unknown and Cluttered Environments, Robot Exploration, Decision Making with Uncertainty.**

I. Introduction

Urban search and rescue (USAR) is a time sensitive operation with the primary objective of exploring a disaster site containing cluttered rubble in order to find trapped victims [1]. Mobile robots can be deployed in these harsh environments to aid rescue workers [2]. To minimize workload, fatigue and the stress placed on robot operators, these robots are being designed to have some level of autonomy. Moreover, there is a need to move towards full robot autonomy as rescue workers are a scarce resource [3].

In USAR environments, uncertainty exists with respect to both robot perception and mobility. Due to the unstructured and unpredictable nature of the environment, sensory noise can be high, and the robot can slip while trying to navigate the rough terrain [3],[4]. Other environmental factors, such as smoke, dust and fire also reduce the sensing capabilities of the robot [5]. This uncertainty affects the robot's ability to explore and map an unknown environment, while localizing and identifying victims.

To date, a number of semi-autonomous [2],[6]-[15], and autonomous [4],[5],[16],[17] navigation and exploration techniques have been proposed for search and rescue scenarios. The semi-autonomous techniques deal with uncertainty, in the USAR environment, by requiring an operator to plan the robot tasks or directly control the robot via teleoperation.

Our own previous work for USAR applications has focused on using learning-based semi-autonomous controllers in order for operators to share the USAR tasks of exploration and victim identification with either a single robot [13] or teams of robots [6],[15]. The robots were able to learn from their own experiences as well as those of the operator in order to effectively explore such unknown cluttered environments. Operator assistance was only requested by the robots when needed. These controllers used a MAXQ hierarchical reinforcement learning technique which models the problem as a Markov Decision Processes (MDP), and decomposed the overall USAR mission into smaller subtasks such as exploration, navigation and victim identification. These subtasks were individually learned and then combined into an overall task mission.

Existing autonomous exploration and navigation techniques, on the other hand, have not directly considered the uncertainty associated with the robot in their modeling of the robot tasks, e.g. [4],[5],[16],[17].

In order to improve autonomy of rescue robots in cluttered unknown environments, we present the novel use of a partially observable Markov Decision Process (POMDP) approach which directly incorporates uncertainty within the decision-making layer of the robot controller. The robot chooses its actions based on its belief state of the environment. The belief state is updated as the robot moves in the environment based on the new sensory information available to it (i.e., new observations of the environment). This allows information about the features of the environment to be updated while the robot is exploring. We utilize a hierarchical POMDP approach to allow for task abstraction of the multiple subtasks used during a USAR mission, which also reduces the number of state-action pairs needed. Each subtask is solved using its own POMDP. Exploration of the unknown cluttered USAR scenes is achieved using a direction-based exploration technique.

II. Robot Exploration in Unknown and Cluttered Environments

When moving from a structured and known environment to a cluttered and unknown environment, standard path planning algorithms cannot be used. This is due to the incomplete map of the environment and the existence of unknown regions within the map [18]. To allow exploration in unknown

This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), and the Canada Research Chairs (CRC) Program.

F. Niroui, B. Sprenger, and G. Nejat are with the Autonomous Systems and Biomechatronics Laboratory (ASBLab) in the Department of Mechanical and Industrial Engineering, University of Toronto, (email: farzad.niroui@mail.utoronto.ca), (e-mail: nejat@mie.utoronto.ca).

environments, the most commonly used exploration technique is frontier-based exploration [19]. In this technique, a robot navigates regions on the boundary between open visited space and unknown space to explore new regions [19]. Occupancy grids can be used to represent the environment for this technique [20]. The majority of semi-autonomous [11]-[13],[15] and autonomous [4],[5],[15]-[20] rescue robots utilize frontier-based exploration or a modified version to explore USAR environments. What makes these approaches different from each other is their specific path planning techniques, exploration strategies and how uncertainty is handled.

In [20], seven frontier-based exploration techniques were compared. These techniques were either cost-based or utility-based approaches which either determined to traverse to: 1) the nearest frontier, 2) the frontier that provided the most information about the environment, or 3) a hybrid of both. In simulated experiments of a robot in office-like environments, the exploration time, quality of map, and computation time for each technique were compared. It was found that the specific demands of the problem governed which strategy was best to use. For example, for time sensitive applications, a cost-based approach was recommended while utility-based approaches were recommended for applications requiring the most knowledge of the environment in the shortest amount of time.

A Multi-Criteria Decision Making exploration strategy was presented in [16] which considered the amount of free area beyond the frontier, probability that the robot can communicate information from the frontier back to the base station, and the distance between the frontier and the robot. The objective of this approach was to find the best frontier candidate based on the aforementioned criteria, and formulate a path to it based on the current map information. The strategy was tested in a simulated USAR-like approach against other exploration strategies which used hybrid distance and information gain, and nearest frontier methods. One or two robots were deployed to explore indoor environments with different levels of clutter. At various time intervals, the quality of the map for each strategy was recorded. Results showed that, on average, the multi-criteria exploration strategy outperformed the other strategies.

In [17], a modified version of frontier-based exploration was used. Each cell in the occupancy grid was assigned a distance transform and an obstacle transform value. The distance transform was the cost of navigating to a cell and the obstacle transform was the proximity of obstacles to the cell. A path transform was computed from these two values and by combining it with the frontier-based exploration, the robot moved to the shortest safest frontier available. This approach was successfully tested on a physical robot which was able to explore and map an indoor environment. By inspecting the quality of the map and the sequence of the exploration, it was concluded that the robot was following a sub-optimal strategy but had a behavior which kept it from colliding with obstacles.

In general, frontier-based exploration techniques do not inherently model uncertainty during decision making. To address this, a small number of techniques have been developed to be used with frontier-based exploration to avoid specific uncertain situations when navigating [4],[11]. For example, in [4] the frontier-based exploration technique was used with a two-level navigation method based on modified versions of the Probabilistic Roadmap and Randomized Kinodynamic Planning techniques. This approach allowed a robot to generate a global path and locally adjust the robot's trajectory to safely follow the path. A set of movement constraints were defined to help the robot navigate the environment. Additionally, the movement of the robot was restricted at certain known parts of a rescue environment to prevent the robot from getting into situations such as being stuck or crashing. In both simulated and robot experiments, the approach was able to safely navigate a robot in a cluttered and partially unknown environment.

In [11], frontier-based exploration was used to enable a robot to explore an unknown environment. To overcome the challenge of navigating rough terrain, data from inertial sensors and wheel encoders were fused with the current drawn from the battery to classify the resistance of the terrain for robot movement. This allowed the robot to determine what it was able to traverse. In highly cluttered environments, a human operator controlled the robot and increased its navigation speed.

In general, the majority of existing autonomous frontier-based robot exploration techniques have not directly incorporated environment, sensor or actuator uncertainty within their decision making model. Therefore, the states of the environment are assumed to be fully observable by the robot when making navigation decisions. Some have dealt with this challenge by restricting a robot to only navigate and traverse known parts of the environment or terrain that it can climb over, however these approaches also do not directly consider sensor or actuator noise. In cluttered and unknown environments, such as USAR scenes, such uncertainty is prevalent, and needs to be considered at the modeling and decision making layer in the robot controller. In our work, we uniquely model sensor and actuator uncertainty directly when determining optimal robot actions in order to achieve improved exploration performance in unknown and cluttered environments. We do this by modeling the subtasks of the robot exploration problem using POMDPs.

III. PROPOSED POMDP APPROARCH FOR ROBOTIC URBAN SEARCH AND RESCUE

In USAR applications, a robot needs to explore an unknown and cluttered environment, while navigating the environment to find victims. This overall task can be decomposed into a group of discrete subtasks which are associated with each other through a hierarchical task structure. Each individual subtask is modeled as a POMDP.

A. USAR Task Hierarchy

The USAR problem can be described using the following four subtasks [13]: *Root, Navigate to Unvisited Regions (NUR), Victim Identification (VI), and Navigation*. This decomposition allows each subtask to only include the states and actions that are necessary only for that particular subtask. Therefore, reducing the computational complexity of the overall problem. Figure 1 shows how the subtasks are associated with each other within a task hierarchy.

1) State and Action Space of the Subtasks

Root Task

The *Root* subtask represents the overall USAR problem. As previously mentioned, this problem is defined as exploring an unknown environment while searching for victims. The state space for this task is defined as $S(L_R, M_{xyz}, V)$, where L_R is the robot's location with respect to the global coordinate system, M_{xyz} is the generated 3D map of the environment. In this work, the map is represented in an occupancy grid configuration with terrain information (open space, voids, non-climbable obstacles and climbable obstacles), and V is the presence of a potential victim. The robot starts in the *Root* task, and it can either implement the *NUR* subtask to globally explore unvisited regions or the *VI* subtask to identify potential victims.

Navigate to Unvisited Regions (NUR)

The *NUR* subtask allows the robot to explore the unknown environment while building a map. To enable the exploration of an unknown and cluttered environment, a direction-based exploration technique which we previously developed is employed [13]. This utility-based exploration technique takes into consideration the type of terrain, distance to the robot, and amount of unknown area to determine the exploration direction the robot should follow as defined by *North*, *East*, *South*, and *West* directions. The state space for this task is $S(L_R, M_{xyz}, V)$. The robot can either *Navigate* the local environment, or *Exit* the exploration subtask (when the robot has finished exploring the scene or when a potential victim is present in order for *VI* to take over). The *Exit* action signals the end of exploration, and the robot exits into the *Root* task.

Victim Identification (VI)

In this subtask, the robot identifies potential victims, and tags their locations within the map. The state space is defined as $S(L_R, L_{V/R}, M_{xyz})$ where $L_{V/R}$ is the relative location of a potential victim with respect to the robot's location. The available actions are *Navigate* and *Tag*. *Navigate* will allow the robot to move closer to the victim's location if needed to do victim identification, and *Tag* will tag the victim location in the map.

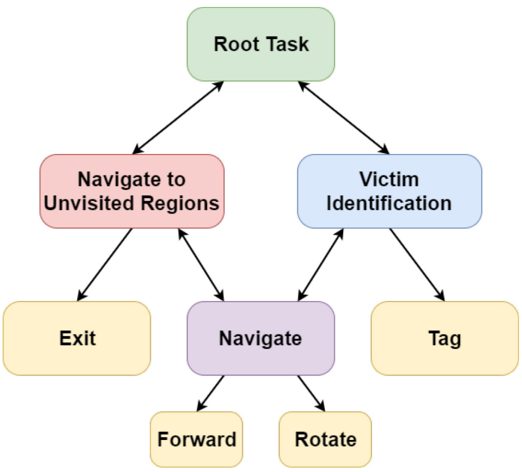

Fig. 1. The task hierarchy for the overall USAR task.

Navigate

The *Navigate* subtask allows the robot to traverse the terrain in the environment by performing local navigation. If the robot has implemented the *NUR* subtask first, then the robot tries to move in the exploration direction. If it is coming from the *VI* subtask, *Navigate* is used to move the robot closer towards the victim location for improved identification. The state space is defined as $S(L_R, M_{xyz}, D_E, L_{V/R})$ where D_E is the desired exploration direction. The robot's navigation actions are *Forward* and *Rotate*. The move forward allows the robot to move from one cell to an adjacent cell in the environment, and the rotate allows the robot to turn in order to access cells to the left, right and behind the robot.

B. POMDP Modeling of the Subtasks

A POMDP model is used here to allow a robot to make decisions when uncertain scenarios exist, which can be common in unknown and cluttered USAR environments. In general, a POMDP is represented as a tuple $<S, A, \Omega, T, O, R>$, where S is the finite set of states, A is the finite set of actions, Ω is a finite set of observations, T is the transition probability function, O is the observation function and R is the reward function [21]. Moreover, a belief state is represented as a probability distribution over the state space [21]. In POMDP, a policy is formulated which maps the belief state to an action. In the proposed USAR task hierarchy, a POMDP is used to model each of the subtasks.

Observation

Whenever the robot attempts an action, an observation of its surrounding environment is made. The observation includes the sensed information of the cells surrounding the robot, and the existence of potential unidentified victims in close proximity to the robot.

Transition Probability and Observation Functions

The transition probability function helps define the probability of going from state s to state s', after taking action a. The observation function defines the likelihood that the robot makes observation o, after taking action a and enters state s'. These functions are used to directly incorporate actuation, and sensor uncertainty.

The transition and observation probabilities can be represented as [22]:

$$T(s,a,s') = \Pr(s'|s,a) \quad , \tag{1}$$

$$O(s',a,o) = \Pr(o|s',a) \quad . \tag{2}$$

Reward Function

After taking specific actions, the robot is either rewarded or punished. $R(s)$ is the immediate reward that the robot receives for being in state s and executing action a. Table I contains the different actions and their associated rewards for the reward function.

TABLE I. THE REWARD FUNCTION USED BY POMDP FOR THE USAR TASK.

Actions	**Rewards**
Finished USAR mission	+100
Exit NUR (incompleted exploration)	-25
Navigate to unvisited cell	+10
Tag Victim Correctly	+25
Collision with obstabcle while *Navigating*	-20
Navigate to already visisted cell	-1

Belief State

The robot starts out with an initial belief state b^0. In the USAR exploration problem, the initial belief favors a single state, as the robot's actual state is known. If the robot takes action a to go from state s to s', it will make an observation $o \in \Omega$ of state s'. From this observation, it will update its belief state. When an observation is made, the belief state is updated using Bayes' rule [21]:

$$b^{ao}(s') = \frac{\Pr(o|s', a)}{\Pr(o|b, a)} \sum_{s \in S} \Pr(s'|s, a)\, b(s) \quad , \qquad (3)$$

where

$$\Pr(o|b, a) = \sum_{s' \in S} \Pr(o|s', a) \sum_{s \in S} \Pr(s'|s, a)\, b(s) \; . \quad (4)$$

Based on the state transition, the robot receives a reward as described in Table I. In POMDP, the robot chooses an action that maximizes its value function. At belief b, the value of a policy π is defined by the expected total discounted reward [21]:

$$V_\pi(b) = E\left(\sum_{t=0}^{\infty} \gamma^t R\big(b_t, \pi(b_t)\big) | b_0 = b\right) \quad , \qquad (5)$$

where γ is the discount factor and t is the time step.

Policy Computation

The policy that maximizes V_π is the optimal policy π^*. To solve the POMDP models for the subtasks, an optimal policy which satisfies the Bellman optimality equation needs to be computed [21]. There are several offline and online techniques which can be used to compute a policy.

Although POMDP has the benefit of being able to handle uncertainty, the process of formulating a policy can be computationally expensive. The belief space grows exponentially with the number of states and so does the observation-action history considered for planning [22]. This is especially evident in offline techniques where all future scenarios must be considered. Moreover, in offline policy computation, prior knowledge of the environment is required.

In online policy computation, the robot searches for a single best action, executes that action and updates its belief [22]. For this reason, prior knowledge of the environment is not necessary, but most online solvers still suffer from the large belief space and observation-action history of the POMDP [22]. The *Determinized Sparse Partially Observable Tree* (DESPOT) [22] is chosen in this work as our online solver. DESPOT overcomes the abovementioned problem by searching only a set of randomly selected scenarios. A scenario is defined as an abstract trajectory $(s_0, a_1, s_1, o_1, a_2, s_2, o_2, \ldots)$.

A common online policy computation approach is to construct a belief tree, where the tree branches off at each level based on the number of available actions and then observations [22]. This approach searches ahead for a policy that maximizes the value function. DESPOT operates in a similar manner, but with several observation branches removed based on the sampled scenarios [22]. At each time step, the optimal policy is computed and the first action of the policy is taken. The robot makes an observation and updates it belief. Then, the process is repeated under a new set of random scenarios.

IV. SIMULATED EXPERIMENTS

To validate the performance of our proposed POMDP approach in exploring and finding victims in unknown and cluttered environments, we conducted 132 trials in simulated USAR-like environments. Namely, we developed a 2.5D simulator using Qt 5.9 to represent varying environments for a robot to explore. Randomly generated scenes using the three different scene sizes of 10x10 m^2, 20x20 m^2 and 30x30 m^2 were generated, with each cell size being 1x1 m^2. The number of victims in the scenes varied from 4-8 and the amount of clutter (climbable and non-climbable obstacles) was also varied from 20%-50%, respectively. Examples of the environments are shown in Fig. 2.

Each cell was classified as open, climbable obstacle, non-climbable obstacle, a non-traversable void, or containing a victim. Climbable obstacle cells had varying heights. The higher the obstacle height, the more likely that the robot would fail to navigate into that particular cell. This actuation uncertainty was modelled as a decreasing linear relationship between actuation success rate and obstacle height.

Sensory information from a thermal camera for victim identification, and a 3D sensor for obstacle height detection and mapping were modeled with added noise. For the added noise, pseudo-random numbers were generated and added to the probability of the sensors to correctly classify a victim or a cell type in the map. This introduced uncertainty to the sensor classifications.

A. *Performance Metrics*

The performance metrics utilized were defined as: 1) the number of victims found and tagged, 2) the percentage of environment that was covered by the robot, and 3) the number of collisions the robot had with non-climbable obstacles.

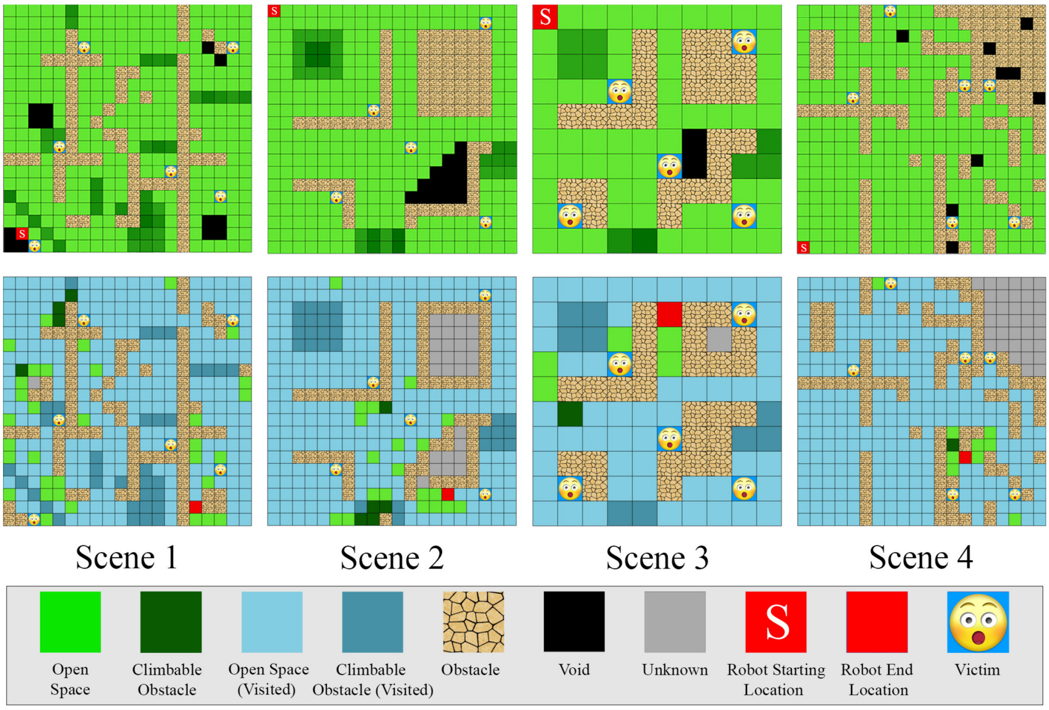

Fig. 2. Four different scene setups for measuring the performance of the POMDP approach (top row) and a sample of their corresponding generated map that the robot generated (bottom row).

B. Results

Table II presents the experimental results for each scene size and clutter level.

Number of Victims Found

On average, the robot found 100%, 99.2%, and 97.5% of the victims for the scene sizes of 10x10 m^2, 20x20 m^2, and 30x30 m^2, respectively. In the cases where not all victims were tagged, the robot failed to formulate a path which allowed it to explore the local area where the victims were located. These areas were highly cluttered when compared to the rest of the environment and the obstacle layout made it so that the robot had only a tight entry to the area.

TABLE II. AVERAGE PERFORAMNCE METRIC RESULTS FOR DIFFERENT SCENE SIZES AND LEVEL OF CLUTTER.

	Scene Size (m^2)	Level of Clutter (%)			
		20	30	40	50
Average Number of Victims Found	10x10	4/4	4/4	4/4	4/4
	20x20	6/6	5.9/6	5.9/6	6/6
	30x30	8/8	7.9/8	7.5/8	7.8/8
Average Percent Coverage	10x10	98.5	100.0	99.7	99.4
	20x20	98.8	98.4	99.6	99.0
	30x30	99.6	97.2	93.4	95.6
Average Number of Collisions	10x10	10.6	7.3	10.7	7.5
	20x20	18.3	24.5	19.8	18.5
	30x30	25.7	26.4	18.9	13.2

Percent Coverage

The average percent coverage for varying levels of clutter based on scene size is presented in Fig. 3. Overall, the robot was able to explore on average 98% of the scenes. The coverage of the robot was slightly lower for some of the 30x30 m^2 cases with 40% and 50% clutter. This is due to the robot exiting the exploration subtask before it had explored the entire scene. Additionally, the effect of robot starting location on the coverage was investigated, with the robot starting at 1-4 different starting locations in the scenes. There was no effect of starting location on exploration found.

Number of Collisions

Normalizing the number of collisions based on the number of movement steps in each trial, for scene sizes of 10x10 m^2, 20x20 m^2, and 30x30 m^2, there were on average 4.1, 2.2, and 0.8 collisions per 100 steps, Fig. 4. The decreasing trend in the number of collisions as the scene size increases is due to the fact that in larger scenes there are more open spaces for the robot to safely navigate to.

V. Conclusion

In this paper, we present a novel POMDP approach for robot exploration and victim identification in cluttered and unknown environments. The approach is able to incorporate sensor and actuator uncertainty in order to autonomously explore USAR-like environments and identify victims in these environments. The overall USAR task was decomposed into four subtasks *Root Task, Navigate to Unvisited Regions, Victim Identification,* and *Navigation*. Each subtask was modeled as a POMDP and the subtasks were related to one another within a hierarchical task structure. The online POMDP solver, DESPOT, was used to determine the robot actions for each subtask. Simulated experiments in cluttered USAR-like environments were conducted with varying scene sizes, obstacle layouts, and level of clutter. The results showed that the robot was able to identify victims while exploring the majority of the scenes, regardless of the scene size and amount of clutter. Future work will include integrating our approach and testing it in larger real environments.

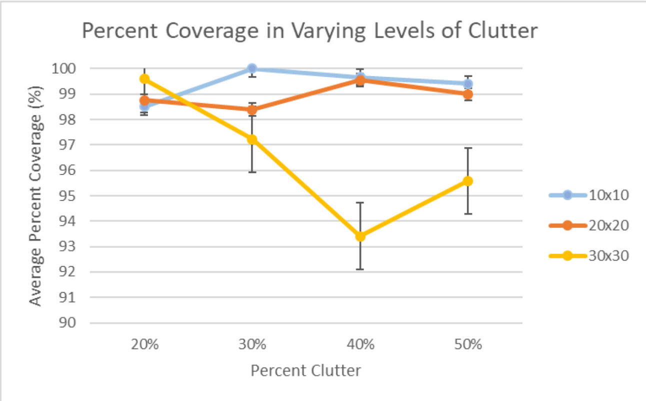

Fig. 3. Average percent coverage for different scene setups.

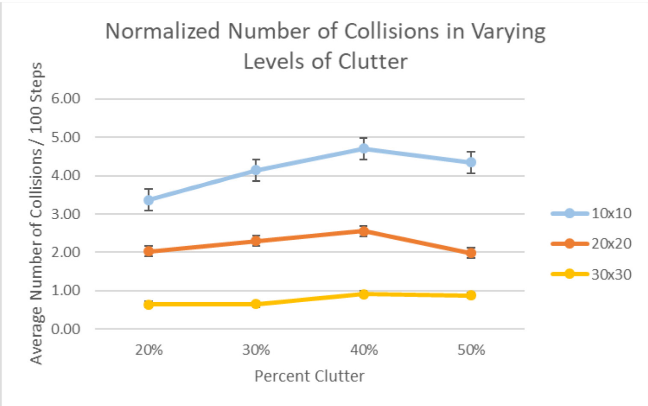

Fig. 4. Average normalized number of collisions for different scene setups.

REFERENCES

[1] C. S. Goodman and D. E. Hogan, "Urban Search and Rescue", *Disaster Medecine*, 2nd ed., J. L. Burstein and D. E. Hogan, Ed. Philadelphia: Lipincott Williams & Wilkins, pp. 72-85, 2007.

[2] Y. Liu and G. Nejat, "Robotic urban search and rescue: a survey from the control perspective", *Journal of Intelligent and Robotic Systems*, vol. 72, no. 2, pp. 147-165, 2013.

[3] A. Birk and S. Carpin, "Rescue robotics – a crucial milestone on the road to autonomous systems", *Advanced Robotics*, vol. 20, no. 5, pp. 595-605, 2006.

[4] D. Calisi et al., "Autonomous Navigation and Exploration in a Rescue Environment", in *Proceedings of the 2nd European Conference on Mobile Robotics*, pp. 110-115, 2005.

[5] A. Kleiner and C. Dornhege "Real-Time Localization and Elevation Mapping within Urban Search and Rescue Scenarios", *Journal of Field Robotics*, vol. 24, pp. 723-745, 2007.

[6] Y. Liu, G. Nejat, J. Vilela, "Learning to cooperate together: a semi-autonomous control architecture for multi-robot teams in urban search and rescue", in *Proceedings of IEEE Symposium on Safety Security and Rescue Robotics*, 2013.

[7] D. J. Bruemmer et al. "'I call shotgun!': an evaluation of mixed-initiative control for novice users of a search and rescue robot", in *Proceedings of IEEE Conference on Systems, Man and Cybernetics*, pp. 2847-2852, 2004.

[8] R. Murphy and J. Sprouse, "Strategies for Searching an Area with Semi-Autonomous Mobile Robots", in *Proceedings of Robotics for Challenging Environments*, pp. 15-21, 1996.

[9] A. Finzi and A. Orlandini, "Human-Robot Interaction through Mixed-Initiative Planning for Rescue and Search Rovers", *AIIA-05*, pp. 483-494, 2005.

[10] R.Wegner and J. Anderson, "Agent-Based Support for Balancing Teleoperation and Autonomy in Urban Search and Rescue", *International Journal of Robotics and Automation*, vol. 21, no. 2, pp. 120-128, 2006.

[11] D. Bruemmer, D. D. Dudenhoeffer, and J. Marble, "Dynamic-Autonomy for Urban Search and Rescue", in *Proceedings of the 2002 AAAI Mobile Robot Workshop,* Edmonton, Canada, 2002.

[12] A. Birk et al., "The IUB Rugbot: an intelligent, rugged mobile robot for search and rescue operations", in *Proceedings of IEEE International Workshop on Safety, Security, and Rescue Robotics (SSRR)*, 2006.

[13] B. Doroodgar, Y. Liu, and G. Nejat, "A Learning-Based Semi-Autonomous Controller for Robotic Exploration of Unknown Disaster Scenes While Searching for Victims", *IEEE Transactions on Cybernetics,* vol. 44, no. 12, pp. 2719-2732, 2014.

[14] Y. Liu and G. Nejat, "Multi-robot cooperative learning for semi-autonomous control in urban search and rescue applications", *Journal of Field Robotics*, vol. 33, no. 4, pp. 512-536, 2016.

[15] Y. Liu, G. Nejat, and J. Vilela, "Learning to cooperate together: a semi-autonomous control architechture for multi-robot teams in urban search and rescue", in *Proceedings of the 11th IEEE International Symposium on Safety, Security, and Rescue Robotics*, 2013.

[16] N. Basilico and F. Amigoni, "Exploration strategies based on multi-criteria decision making for searching environments in rescue operations", *Auton Robot*, pp. 401-417, 2011.

[17] S. Wirth and J. Pellenz, "Exploration Transform: A stable exploring algorithm for robots in rescue environments", in *Proceedings of the 2007 IEEE International Workshop on Safety, Security, and Rescue Robotics*, pp. 1-5, 2007.

[18] A. Zelinsky, "A mobile robot exploration algorithm", *IEEE Transactions on Robotics and Automaton*, vol. 8, no. 6, pp. 707-717, 1992.

[19] B. Yamauchi, "A frontier-based approach for autonomous exploration", in *Proceedings of IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pp. 146-151, 1997.

[20] M. Juliá, A. Gil, and O. Reinoso, "A comparison of path planning strategies for autonomous exploration and mapping of unknown environments" *Auton Robot*, vol. 33, pp. 427-444, 2012.

[21] M. Spaan, "Partially Observable Markov Decision Processes", *Reinforcement Learning: State of the Art*, ch. 12, pp. 387-414, 2012.

[22] A. Somani et al., "DESPOT: Online POMDP Planning with Regularization", *Advances in Neural Information Processing Systems*, pp. 1772-1780, 2013.