Advanced Robotics, 2015
http://dx.doi.org/10.1080/01691864.2015.1015443

Taylor & Francis
Taylor & Francis Group

FULL PAPER

Enhanced frontier-based exploration for indoor environment with multiple robots

Mohammad Al khawaldah[a] and Andreas Nüchter[b]*

[a]*Al-Balqa Applied University, Al-Salt, Jordan;* [b]*Informatics VII : Robotics and Telematics, Würzburg, University of Würzburg, Germany*

(*Received 29 November 2013; revised 7 July 2014; accepted 12 December 2014*)

In this paper, the exploration and map-building of unknown environment by a team of mobile robots is intensively investigated. A new exploration technique is proposed to increase the exploration efficiency. In particular, the new technique has two main objectives: firstly, it aims at reducing the exploration time and the traveled distance by reducing the overlap which takes place when a certain area in the environment is explored by more than one robot. To achieve this, a new procedure to assign the next target location for each individual robot is proposed. And secondly, it aims at reducing computations complexity required by target selection and path planning tasks. More importantly, the proposed technique obviates the need for environment segmentation complex procedures which is adopted in some previous important research works. The new technique is intensively tested with different environments. The results showed the effectiveness of the proposed technique.

Keywords: robot; coordination; exploration; map; frontier

1. Introduction

Exploration is the 'act of moving through an unknown environment while building a map that can be used for subsequent navigation'.[1] Exploration and map-building of an unknown environment is one of the main issues in autonomous mobile robotics due to its wide range of real-world applications. Such applications may include search and rescue, hazardous material handling, military actions, planetary exploration, path planning, and devastated area exploration.[2] Generally, autonomous robot is able to incrementally construct a model (map) for its environment based on the sensory information gathered in an online fashion, i.e. while navigating through the environment. This process requires choosing the best next location for the robot to visit, planning the shortest path to reach that location and finally controlling the robot's motion in its journey to that location.

Mobile robots need a map to effectively navigate in their environment. The ability of mobile robots to autonomously move in an unknown environment to gather the sensory information required to build a map for navigation is called autonomous exploration. Simultaneous localization and mapping technique is often used to construct a map for the environment and localize the robots on it. As the robots move to unexplored new areas, these areas are then included in the map. The main challenge in autonomous exploration is how robots plan the order to visit the remaining unexplored areas while minimizing the total traveled distance.[3]

The use of cooperative multi-robot systems for exploration purposes has several advantages over single robot systems. Mainly, cooperating robots have the ability to perform a single task quicker than a single robot because the exploration is performed simultaneously.[4] Moreover, using several robots introduces redundancy which makes teams of robots more fault-tolerant than only one robot. One more advantage of robot teams is due to the merging of overlapping information that can help compensate for sensor uncertainty. For instance, a team of robots has been shown to localize themselves more efficiently and precisely, especially when they have different sensor capabilities. On other hand, when robots operate in teams or groups there is the risk of possible interferences between them. 'For example, if the robots have the same type of active sensors such as ultrasound sensors, the overall performance can be reduced due to cross-talk between the sensors. Also, the more robots are used the longer detours may be necessary in order to avoid collisions with other members of the team'.[5]

The aim of this paper is to increase the exploration efficiency (i.e. to reduce the environment exploration time required to accomplish the exploration task). In particular, the goal of the proposed technique is to have multiple mobile robots exploring an unknown environment as fast as possible, while coordinating their actions and sharing their local maps in certain time instances to save time and robot motor energy. In the suggested technique, each robot is equipped with a laser scanner to scan the environment. An

*Corresponding author. Email: andreas@nuechti.de

advanced frontier-based exploration technique is proposed to guide the robots during the exploration. Moreover, the new technique led to a significant reduction of the required computations for the exploration task. The new technique was intensively tested and the results show that the new technique is robust and leads to promising results.

2. Related work

In this section, the most relevant works to the proposed approach are summarized. In exploration of unknown environment with multiple robots, there are two main known ideas: First is frontier-based exploration where the robots are directed to the front lines (borderlines between explored and unexplored areas) and second map segmentation where the environment is partitioned into different segments and each robot is dedicated to explore one of these segments.

2.1. *Frontier-based Exploration*

Most of the published multi-robot exploration algorithms have relied on the use of 'frontier cells', e.g. [5–10]. The concept of frontier-based exploration was initially introduced by Yamauchi [8]. He stated that 'To gain the most new information about the world, move to the boundary between open space and uncharted territory'. He presented a technique to build grid maps by which the environment is represented by evenly-spaced grids (2D map). Each grid cell has a numeric value that indicates the presence of an obstacle in the corresponding region of the environment. The robots exchange information about the map that is continuously updated whenever new information sets come through sensors. To discover the environment, each robot moves toward the closest frontier cell. A frontier cell is a free (not occupied) cell for which at least one of its neighboring cells is unexplored. When a robot is directed to such a cell, it is expected that it will gain information about the unexplored area when it arrives. Because a map may contain more than one unexplored area, the challenge arises of how to plan the exploration mission by choosing the most appropriate frontier cell. When more than one robot is involved in the exploration, it is important to avoid more than one of them moving to the same cell. In [11], Yamauchi proposes an exploration technique in which the distance from the robot to the frontier cell is the unique parameter that is considered to select the robot future cell (location).

A more advanced technique was proposed by Burgard et al. [5] in which 2D occupancy grid maps were employed to represent the environment to be explored. The exploring robots start at known initial positions. The aim is to minimize the overall exploration time by choosing suitable target points (frontier cells) for individual robots so that they explore different sections of the environment and the overlapping between them is minimized. In this technique, each robot chooses its next target cell by calculating a bidding value for each target cell. The bidding value of a frontier cell depends on the utility of the frontier cell (the area of environment that is expected to be explored if the robot visits the frontier cell) in addition to the distance from the robot to the frontier cell. The bidding value of a frontier cell is the difference between the frontier cell utility and cost. The robot chooses the frontier cell which has maximum bidding value and then it plans a path to its target.

A slightly different technique from Burgard's was presented by Sheng et al. [10]. Sheng considered the limited communication range between the robots. A nearness measure was introduced in the bidding algorithm which keeps the robots together within the communication range. Robots start from initial positions which are close to each other and the relative positions are known to all for each robot. The 2D occupancy grid was again used to represent the environment. The bidding function selects the cell with the maximum exploration information and the minimum cost with respect to each robot.

Ziparo et al. [12] presented an interesting technique in which the goal is to reduce the exploration time by using radio frequency identification (RFID) tags as coordination points. Robots, in this technique, deploy tags in the environment to form a network of reachable locations. In this approach, a two-layered algorithm is used. At the first layer, there is a local part, where robots are coordinated by RFID chips and perform a local search. And at the second layer, based on the local part, there is a global part which is responsible for monitoring the local exploration.

In summary, in frontier-based exploration technique robots are directed to the front lines of the so-far explored area which are the edges between explored and non-explored regions. The main challenge is how each robot finds its next target or future cell? The general solution is based on a bidding value calculated for each frontier cell. The bidding value mainly depends on the distance and on the utility of the frontier cell. The robot then chooses the frontier cell that has the maximum bidding value as a next target. A path to the target is then planned and the robot start moving toward its target cell. However, such techniques require very complex computations and the overlap problem has not been seriously solved.

2.2. *Map Segmentation*

Several researchers employed segmentation techniques [2, 13–17] to reduce the overlap among robots and to avoid the situation in which more than one robot covers same area or segment. In these techniques, the so-far discovered part of the environment is divided into segments. Then, to reduce the overlap, each robot is assigned as different segment. While the robot is inside its own segment, it employs the ordinary above-mentioned frontier-based algorithm to explore the segment. The most challenging step in these techniques is how to divide an unknown environment into different segments. The most popular solution is based on partitioning of a graph named as Voronoi graph.[2,13]

To construct Voronoi Graph $G(m) = (V, E)$ of the current map m, we need to find the set $O_p(m)$ which contains for each point p in the free-space C of the map the set of closest obstacle points. The Voronoi Graph then is given by the set of points in $O_p(m)$ for which there are at least two obstacles (occupied cells)with an equal minimal distance.

$$V = \{p \in C || O_p(m)| \geq 2\} \quad (1)$$

$$E = \{(p, q)| p, q \in V, p \text{ adjacent to } q \text{ in } m\} \quad (2)$$

The Voronoi Graph can be generated from occupancy grid maps as given in [14,18]. In particular, this can be done by applying the Euclidean distance transformation [19] to a grid-based map. The result is a distance map which holds for each grid cell the distance to the closest obstacle.

The separation takes place in the points where the obstacles or walls form restrictions or narrow passages such as doorways. So the closest corresponding two obstacles are connected with an edge (see the red lines in Figure 1 right). In this figure, the environment is separated into three clusters or segments blue, gray, and green. The robot does not leave its segment until it is completely explored. Despite the fact that such techniques have better performance regarding the overlap problem, a very complex procedure (computations) is required to divide and continuously update the division of the environment into different segments. Moreover, if there are obstacles in the environment, a large number of smaller segments is expected to appear. For example, there might be plenty of narrow passages due to some obstacles such as chairs and tables inside a room in an office-like environment. As a result, plenty of segments might appear in a single room while it should be a single segment. To eliminate these false candidates and to combine them in one segment, a more complex procedure is required [8]. Therefore, a new, simpler and more efficient solution is proposed in this paper where no need to build a Voronoi graph nor to segment the environment.

3. The proposed technique

Exploration with single robot is much easier than exploration with more than one robot. This due to the fact that the need for coordination, map merging, data exchange (such as sensory data, position, next target) become a challenging task. For simplicity, we firstly explain the proposed exploration technique for single robot before extending to multiple robots.

The core challenge of exploration of unknown environments is how to plan the order of the next target location for each individual robot. Target selection mainly depends on the state of the robot's current map and robot's current position. In the following paragraphs, a detailed explanation of the proposed target selection procedure is given.

3.1. Exploration with single robot

In our proposed technique, the environment is represented as a grid-based map and each robot is equipped with a laser scanner that is able to scan a circle of radius r centered at the scanner (robot) position. The robot starts the exploration with 360 scan while standing on its starting position, as a result, certain number of frontier cells appear. The number of the frontier cells which appear from the first scan is denoted as n_1, and the number of the frontier cells that appear from the second scan is denoted as n_2 for and so on. The frontier cells subset F_1 which results from the first scan is $\{F_{1i}\}_{i=1}^{n_1} = \{f_{11}, f_{12}, f_{13}, \ldots, f_{1n_1}\}$ and the frontier cells subset F_2 which results from the second scan is $\{F_{2i}\}_{i=1}^{n_2} = \{f_{21}, f_{22}, f_{23}, \ldots, f_{2n_2}\}$ and so on. The frontier cells of all scans for a given robot can be denoted as

$$\{F_{ji}\}_{i=1, j=1}^{n_j, j} = \begin{Bmatrix} f_{11}, f_{12}, f_{13}, \ldots, f_{1n_1} \\ f_{21}, f_{22}, f_{23}, \ldots, f_{2n_2} \\ f_{31}, f_{32}, f_{33}, \ldots, f_{3n_3} \\ \vdots \\ f_{j1}, f_{j2}, f_{j3}, \ldots, f_{jn_j} \end{Bmatrix} \quad (3)$$

where j is the scan number, and n_j is the number of the frontier cells in the corresponding scan j. In particular, the robot builds a 2D array that keeps all the frontier cells which appeared during its navigation. This array is called frontiers array. In each scan, new row containing the new frontier cells is added to the frontiers array. After each scan the robot chooses one of these frontier cells to be its next target and moves toward this new target. Once arrived, it makes another scan and new frontier cells are expected to appear.

It should be noted that while the map is continuously updated, any frontier cell is deleted from the frontiers array once it becomes not frontiers. In other words, the frontier cells are instantly deleted from the frontier array when they are no more on the edge between explored and unexplored areas.

The robot chooses a cell from its frontiers array to be its target. The selection depends on the newly proposed frontier-obstacle concept. The idea of the frontier-obstacle came from the Voronoi graph exploration method.[2,13,14,16,17,20] The frontier-obstacle is an occupied cell that lies on the edge between explored and unexplored areas. Each frontier cell is assigned a value equal to its Euclidian distance from the closest frontier-obstacle (CFOD: closest frontier-obstacle distance) detected in the continuously updated map. The frontier cell with maximum CFOD value is selected to be the next goal cell g for the robot, i.e.

$$g = \max_i \text{CFOD}_i$$

The frontier cells with maximum CFOD value are expected to be on the midway between two frontier-obstacles as shown in Figure 2. In Figure 2, the white color represents the area which is explored by the robot (whose trajectory is also shown) and found to be free. While the gray area

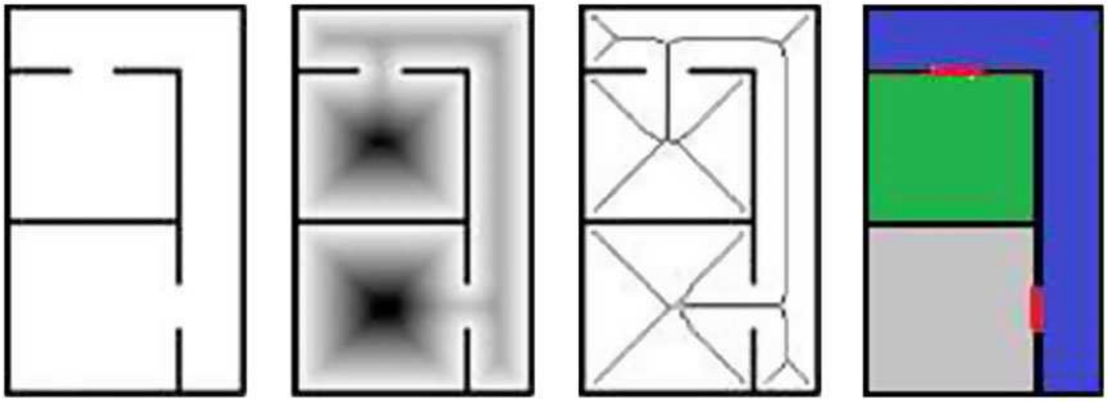

Figure 1. An example for generation of the Voronoi Graph. From left to right: (1) Example grid-based map, (2) Map with distance transform (the darker a point the larger the distance to the closest obstacle), (3) Map and Voronoi Graph generated from the distance transform, and (4) The segmented environment.

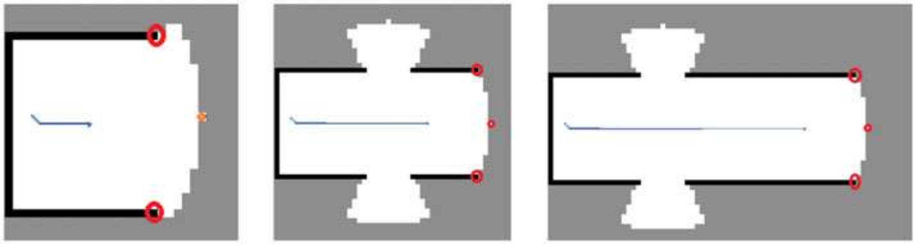

Figure 2. Exploration snapshots with robot trajectory in an office-like environment.

represents the yet unexplored area, the red small circle is the target cell that the robot has selected according to frontier-obstacle concept, and the two red circles on the lower and upper black walls indicates the frontier obstacles. Clearly, the selected cell lies on the midway between two frontier-obstacle cells. This allows taking advantage of the maximum scanning efficiency of the laser scanner, especially in free areas. Visiting such a cell is expected to uncover a relatively large area with one scan. This method is called the frontier-obstacle method. In an office-like environment, this technique helps to discover the corridors at the beginning of the exploration provided that the corridors are wider than the doorways.

The robot checks the frontier cells in the last row of its frontiers array (i.e. the most recent frontier cells) and applies the above-mentioned frontier-obstacle method to select its next target cell. If there is no any frontier cell in the last row (this occurs, for example, when the robot finishes exploring a room in an office-like environment where no frontier cells appear in its last scan) the robot checks the frontier cells of the previous row. Similarly, if there is no any frontier cell in the previous row, it checks the row before and so on. If there is no any frontier cell in the frontiers array, then whole environment is detected and the exploration is finished.

In summary, the robot checks the frontier cells appeared in its recent scan and apply the frontier-obstacle concept to choose its next target cell. If no frontier cells appeared in its last scan, it checks the frontier cells appeared in the scan just before the last scan. The process is repeated until at least one frontier cell is found. This behavior is very beneficial to reduce the exploration time. Figure 3 shows a detailed flow chart representing the proposed technique for single robot.

For example, in an office-like environment, the robot explores the corridor first and then it enters one of the adjacent rooms. Once entered, the robot start exploring this room and it does not leave it until it is fully explored and no need to come back again to it. This reduces the exploration time and obviates the need for environment segmentation which is a complex process that needs complex computational capabilities. Then the robot goes to an adjacent room and so on until the whole environment is discovered. When the robot finishes exploring a room, it goes to the adjacent room (close room) as it always checks only the most recent frontier cells which are expected to be relatively close to the robot and will not waste the time moving among rooms. The process repeats until the whole environment is explored and frontiers array becomes empty.

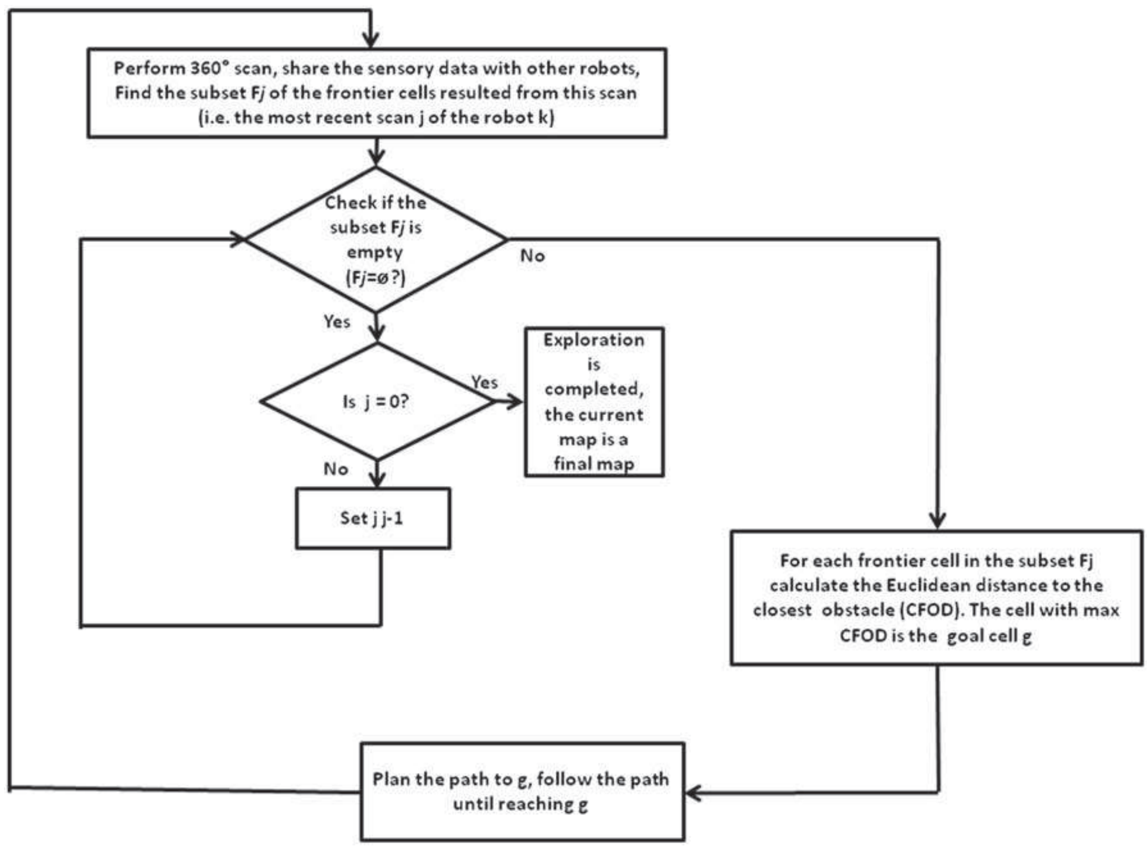

Figure 3. A detailed flow chart representing the proposed Frontier-Obstacle technique for single robot.

3.2. Extention to multiple robots

To extend the proposed approach to multiple robots, several issues need to be considered. The most important issue is how to minimize the overlap. For example, if two robots are exploring an office-like environment, it would be more efficient for each room to be completely explored by only one robot. It would be time consuming if both robots explore both rooms. When to exchange data or maps among robots is also a core challenge in multi-robot systems.

As in exploration with single robot, each robot builds its own frontiers array and applies frontier-obstacle method to choose its target cell from frontier cells that resulted from the most recent scan. If the there are no frontier cells resulted from the most recent scan, the frontier cells of the previous scan are considered and so on. If there are not any frontier cells from all of the scans of the robot (i.e. there is not any frontier cell in its frontiers array), the robot applies the above-mentioned frontier-obstacle method to the frontier cells produced by all other robots (see Equation (4)) to choose the closest frontier cell that has local CFOD maxima. The closest frontier cell with local CFOD maxima is expected to be on the closest doorway or narrow passage. If the chosen cell is in the sensor range of other robots (i.e. close to other robots), it is ignored and the frontier-obstacle method is again applied to choose the next closest frontier cell with CFOD local maxima.

$$\{F_{kji}\}_{i=1,j=1,k=1}^{n_j,j,k} = \{F_{1ji}\}_{i=1,j=1}^{n_j,j_1} \cup \{F_{2ji}\}_{i=1,j=1}^{n_j,j_2} \times \cup \{F_{3ji}\}_{i=1,j=1}^{n_j,j_3} \cdots \cup \{F_{kji}\}_{i=1,j=1}^{n_j,j_n} \quad (4)$$

where k is the corresponding robot.

Figure 4 shows a detailed flow chart representing the proposed technique for multiple robots.

Clearly, there are some core differences between the proposed method and the Voronoi-based exploration. These differences (advantages) are summarized as follows:

(1) In the proposed frontier-obstacles method, no need to build the Voronoi graph for the environment, alternatively, the robot calculates for each frontier cell resulted from the most recent scan, the distance to the closest obstacle lies on the border between the explored and unexplored areas. The frontier cell with maximum distance is then chosen as a next target cell for the robot.

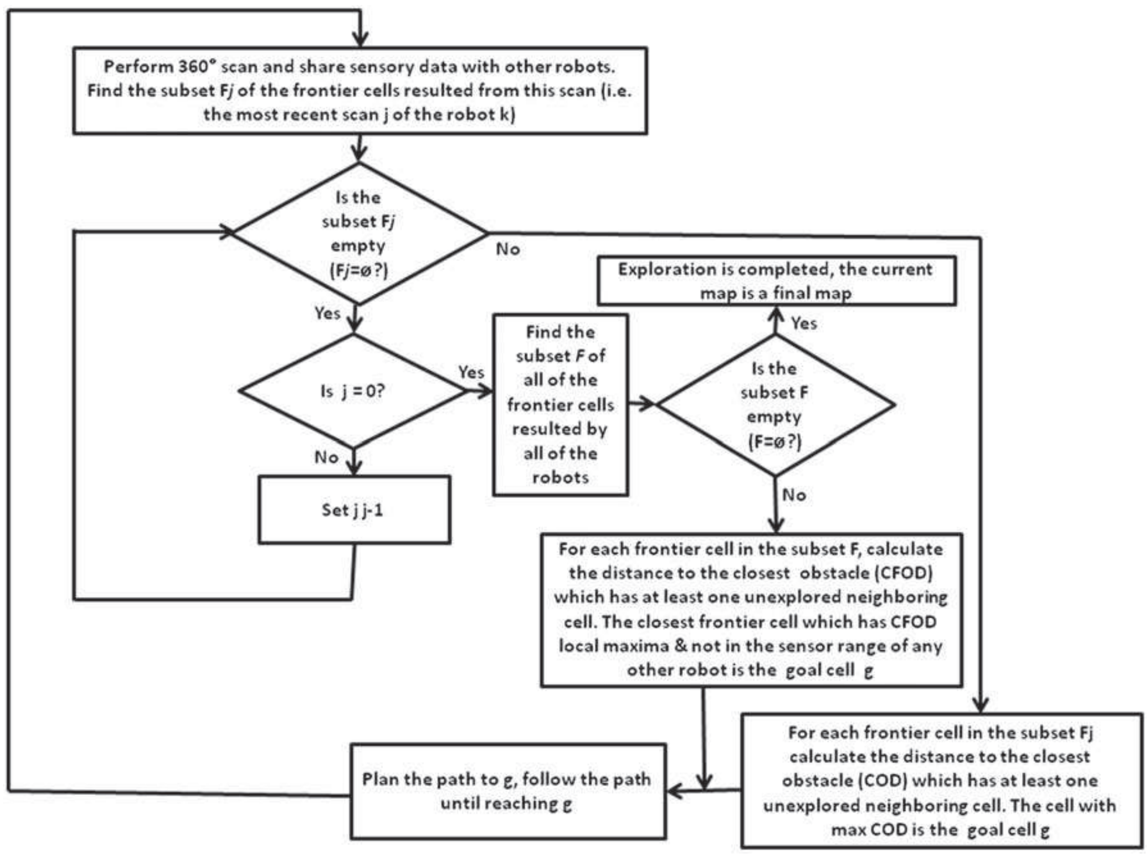

Figure 4. A detailed flow chart representing the proposed Frontier-Obstacle technique for multiple robots.

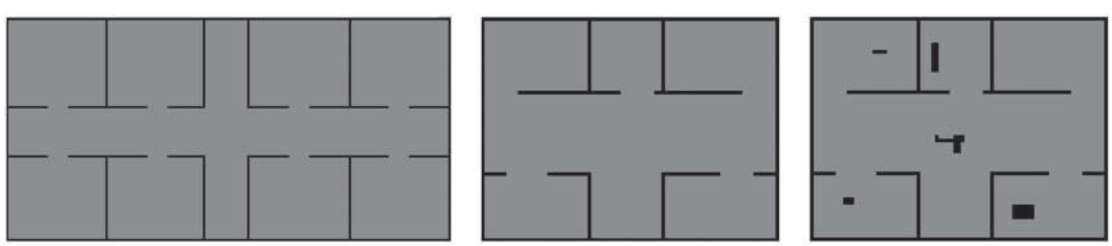

Figure 5. Office-like environments used for testing the proposed exploration technique without obstacles (left & middle) and with some arbitrary obstacles (right).

(2) The frontier-obstacle method considers only the frontier cells resulted from the most recent scan, while in Voronoi-based exploration methods, the distance between the robot and all of the explored cells are calculated.

(3) In the Voronoi-based exploration, to reduce the overlap among robots, the so-far explored area is continuously segmented into different segments and each robot is assigned a different segment. In the frontier-obstacle method no need to segment the environment, alternatively, the overlap is minimized in a straight forward fashion.

(4) The proposed frontier-based approach is not affected by the presence of obstacles.

Employing frontier-obstacles method is also very beneficial to reduce the computation required to select the target cell from the available frontier cells. In this method, unlike all of the published works, only the Euclidian distance of each frontier cell to the closest frontier-obstacle is calculated. On the contrary, in the published works, the free-obstacles path is computed for each frontier cell. This is usually done based on flood fill [6] or *A* [1] algorithms which require complex computational capabilities compared with computing the Euclidian distance in the proposed method.

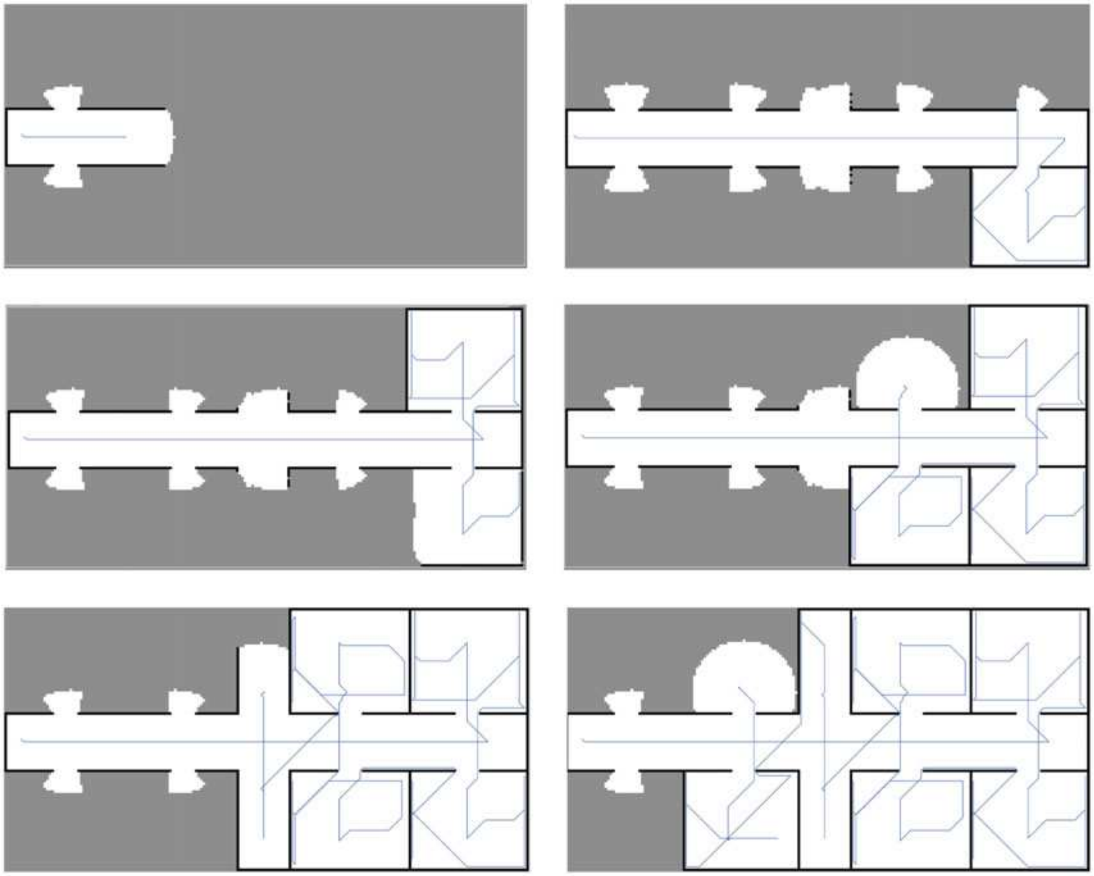

Figure 6. Exploration snapshots and robot trajectory of an office-like environment with single robot. The robot explores the corridor first then it explores the rooms, one by one (see http://youtu.be/zeWhOU672NY).

4. Experimentation and results

In this section, the experiments and the results of the proposed technique are given. We firstly show how the proposed technique perfectly distributes the robots over the environment (regardless of the number of robots and the environment complexity) and obviates the need for map segmentation. And secondly, we compare the results of the proposed technique, i.e. the exploration time and the total traveled distance, with the standard well-known frontier-based exploration method. The environments used for testing the proposed exploration technique are shown in Figure 5. The size of the environment shown in Figure 5 (left) is 200×100 cell, while the size for the other two environments is 80×60 cell. All of the experiments presented in this research work are implemented through simulation.

4.1. Robot distribution

The environment shown in Figure 5 (left) is chosen to evaluate the way in which the proposed technique distributes the robots in the environment. This environment is selected as it contains more rooms than others, this allows for better evaluation. Figure 6 shows the exploration snapshots when exploring with single robot. It is clear that the robot explored the corridor first and then it explored an adjacent room and so on until the whole environment is explored. More importantly, when the robot enters a room it does not leave it until it is completely discovered. In Figure 7, the exploration snapshots when exploring with two robots are shown. It is clear that one robot explores the corridor (called corridor robot) and finds the doorways while the other robot enters the closet doorway (local maxima) and completely explores the room before it goes to the next closest doorway and so on. When the corridor robot finishes exploring the corridor it directly goes to (enters) the closest doorway and completely explore the room and so on until the whole environment is explored. Similarly, Figure 8 shows the exploration snapshots when exploring with three robots. As in exploring with two robots, one robot explores the corridor (corridor robot) and finds the doorways while each of the other robots enters the closet doorway and completely explores the room before it goes to the next closest doorway and so on. When the corridor robot finishes exploring the corridor it directly goes to (enters) the closest doorway and completely explore the room and so on until the whole environment is explored.

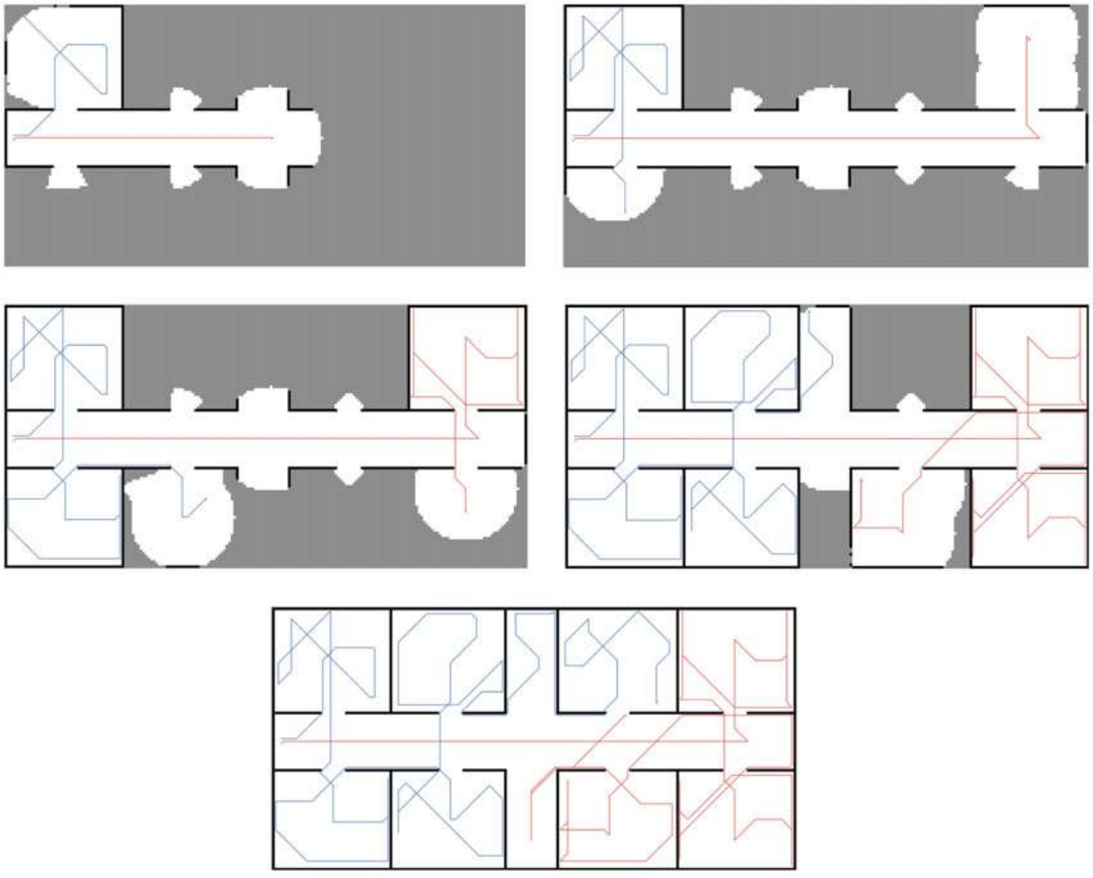

Figure 7. Exploration snapshots and robots' trajectories of an office-like environment with two robots (see http://youtu.be/AMeBD6btkmg).

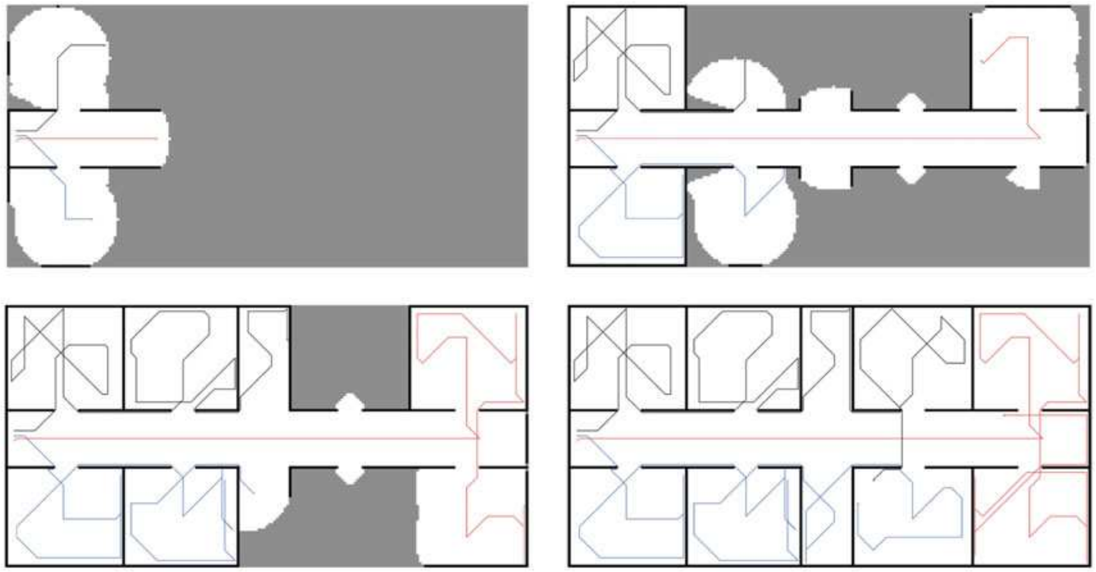

Figure 8. Exploration snapshots and robots' trajectories of an office-like environment with three robots (see http://youtu.be/2fb8N2mdj8U).

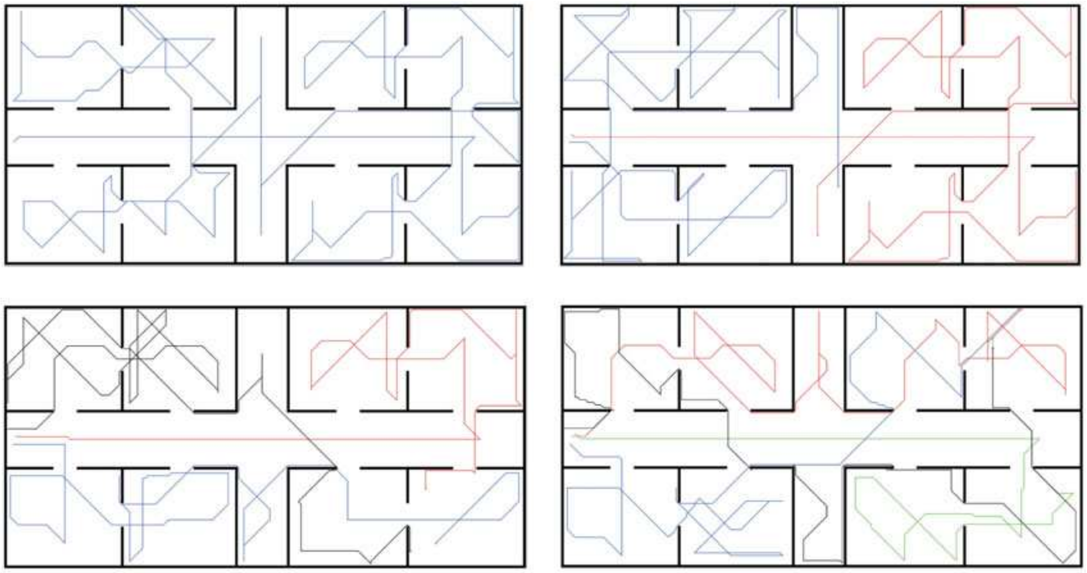

Figure 9. The trajectory maps for the environment shown in Figure 5 (left) with more four doors when explored with one, two, three, and four robots.

In summary, the proposed technique controls the exploration progression through the following rules (1) the robot that completes the first scan before other robots is called corridor robot. The corridor robot explores the whole corridor and finds the doorways. (2) Each of the other robots selects a different doorway and completely explores the corresponding room. (3) Each room (or section) is explored by only one robot except in the case that there more robots than the available doorways. For example, if there are three doorways and four robots, each of the doorways will be assigned a robot. The fourth robot will choose one of the doorways (the closest one) to help the robot that already assigned to this room. Once a new doorway is discovered by the corridor robot, the fourth robot or its colleague in the same room directly leaves its current room and goes to the newly discovered doorway.

Another set of experiments was performed to evaluate the frontier-obstacle technique with more complex environments in which each room has more than one door. For the office-like environment shown in Figure 5 (left), new four doors are introduced (one door between any two adjacent big rooms), as shown in Figure 9. Figure 9 shows the trajectory maps when explored with one, two, three, and four robots. It is clear that, as stated before, one of the robots is dedicated to explore the corridor then each two adjacent big rooms (considered as one room because of the door between them) are fully explored before the robot leaves them and so on. Figure 10 shows the trajectory maps for the same environment with more four additional doors when explored by one and two robots. As before, one robot is dedicated to explore the corridor first producing two separated upper and lower unexplored areas. The robot then fully explores one of these areas before leaving it to the other one. When exploring with two robots, each robot is directed to one of these unexplored areas. In case one of them finishes its area before the other one, it goes to

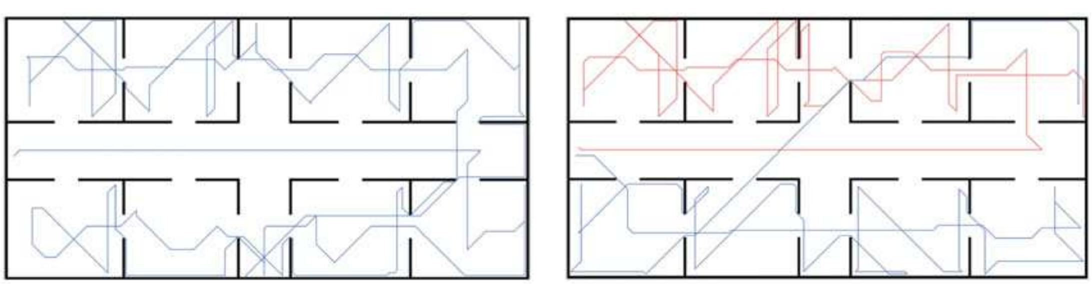

Figure 10. The trajectory maps for the environment shown in Figure 5 (left) with more eight doors when explored with one and two robots.

the other area to help its partner. The result of this set of experiments confirms the efficiency of the frontier-obstacle technique in more complex environment.

Last set of experiments presented in this section was proposed to assess the frontier-obstacle technique with different starting position for each single robot. Figure 11 shows the trajectory maps for the environment shown in Figure 5 (left) when explored with one, two, three, and four robots. Figure 11 shows that even when the robots start from different starting position, which may not be close to each other, each robot is assigned a different room and it does not leave it until it is completely explored. It seems that starting from different starting positions helps to further reduce the overlap among robots.

4.2. Comparison with standard Frontier-based Exploration

The proposed exploration technique was tested with three office-like environments shown in Figure 5. The exploration experiments were run as follows: Each environment is tested with different numbers of robots (1 to 5) then the experiment is repeated five times and the average time to complete the exploration is recorded. Also the total distance traveled by all robots, which is equal to the sum of the distances traveled by each individual robot, is recorded. The experimentation with the proposed technique started with the environment shown in Figure 5 (left); it was explored with one robot, then this experiment was repeated five times, then the average time to complete the exploration in addition to the total traveled distance are recorded. Then it is tested with two robots and repeated five times and, as before, the average time and the total traveled distance are recorded. This procedure is repeated until the number of robots is five. Same procedure is repeated for the other environments. Finally, we compared our proposed technique to the standard frontier-based approach in which each robot is assigned to the closest frontier cell.[10,11,21] In the standard frontier-based exploration, each robot chooses the closest frontier cell with simple procedure to reduce the overlap. In [21], Holz et al. stated that 'Although exploring closest frontiers is a rather simple and native strategy, the resulting paths are reasonably short and do not rank behind those acquired using more sophisticated strategies'. To further reduce the overlap associated with the standard frontier-based exploration, the selected frontier cell has not to be within the sensor range of any other robot as proposed in [11]. In other words, the selected frontier cell should not be close to any other robot.

Figure 12 (left) shows the exploration time with the number of robots for the environment shown in Figure 5 (left), and (right) shows the total traveled distance with number of robots for the same environment. Figure 13 (left) shows the exploration time with the number of robots for the environment shown in Figure 5 (middle), and (right) shows the total traveled distance with number of robots for the same environment. Figure 14 (left) shows the exploration time with the number of robots for the environment shown in Figure 5 (right), and (right) shows the total traveled distance with number of robots for the same environment.

From Figures 12–14, it can be noticed that the proposed enhanced Frontier-based exploration technique significantly outperforms the standard Frontier-based exploration technique. The exploration time with the proposed technique is much less than that of the standard frontier-based technique for the three environments. Moreover, the proposed

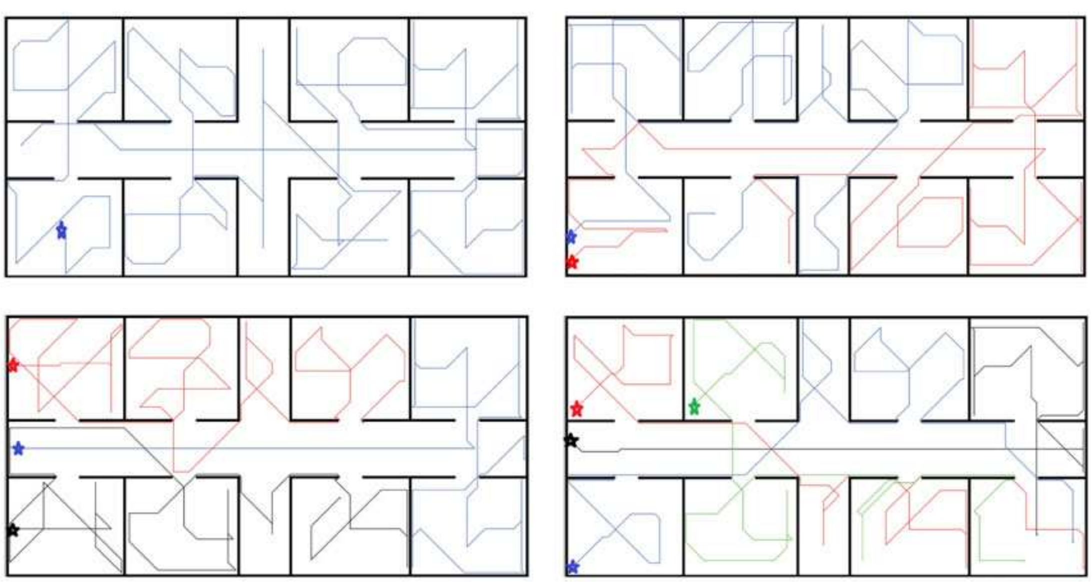

Figure 11. The trajectory maps for the environment shown in Figure 5 (left) when explored with one, two, three, and four robots starting from different positions.

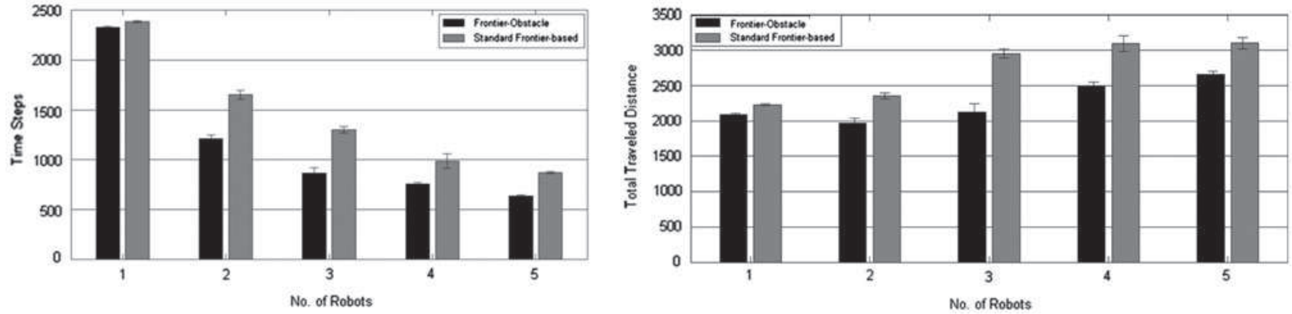

Figure 12. Left: Exploration time with the number of robots for the environment shown in Figure 5 (left). Right: Total traveled distance with number of robots for the same environment.

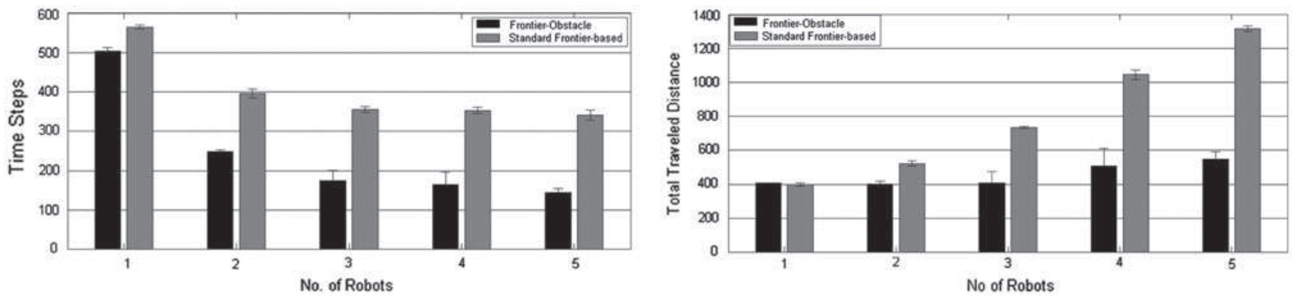

Figure 13. Left: Exploration time with the number of robots for the environment shown in Figure 5 (middle). Right: Total traveled distance with number of robots for the same environment.

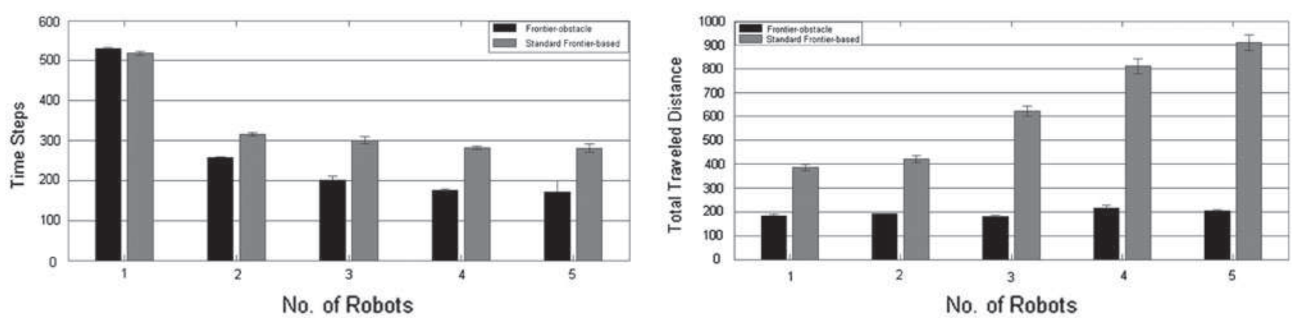

Figure 14. Left: Exploration time with the number of robots for the environment shown in Figure 5 (right). Right: Total traveled distance with number of robots for the same environment.

technique also reduced the total traveled distance by the robots for all of the three environments which means that the proposed technique is an energy saving technique.

5. Conclusions

In this paper, a novel and efficient technique is proposed to coordinate a team of mobile robots in order to explore unknown environment. The aim is to minimize the exploration time and the traveled distance required to accomplish the exploration task. In this technique, each robot chooses one of the frontier cells that have been discovered by the most recent scan of its laser scanner. Usually, each scan produces more than one frontier cell. A frontier-obstacle method is proposed to choose the most appropriate frontier cell among the ones produced by the most recent scan. In case that a given robot does not have any frontier cell available from its own scanner, it applies frontier-obstacle method to the frontier cells produced by other robots to choose the closest frontier cell that lies on the closest doorway or on a narrow passage. The results of the proposed technique led to a balanced distribution of the exploring robots over the environment. The balanced robots distribution reduces the risk of interference and collisions between robots. More importantly, the new exploration technique led to shorter exploration time and shorter traveled distance compared to the standard well-known frontier-based exploration approach.

Most of the well-known exploration algorithms require complex computational capabilities. There are two main reasons for the need for the complex computations, target cell selection and path planning. For target cell selection, the new technique effectively selects the appropriate cell with simple computations based on a new concept named frontier-obstacle method. For path planning, the published exploration techniques employ famous path planning algorithms such as breath-first and A* which requires complex computations. Whereas in the proposed technique, a very simple procedure is used to find the optimal path for each individual robot. Simply, if the goal cell was discovered from the recent scan then there is a clear straight-line path to it. In such a case, a path planning algorithm is only required when the robot finishes exploring a room or a segment and wants to move to another room or segment.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work was partially supported by the Deutsche Forschungsgemeinschaft (DFG Nu-230/6, Nu-230/7, and Nu-230/8).

Notes on contributors

Mohammad Al Khawaldah received his BSc in Electronic Engineering from University of Technology, Baghdad, MSc in Mechatronics Engineering from Al-Balqa Applied University, Jordan and PhD degree (Exploration and Map-building by Cooperating Mobile Robots) from University of Hertfordshire, UK. Currently, he is an assistant professor in Department of Mechatronics Engineering at Al-Balqa Applied University. Recently, he became the Head of Mechatronics Engineering Department at Al-Balqa Applied University. His research interests include: Mobile robots exploration and Discrete control.

Andreas Nüchter holds a tenured associate professorship of Telematics (telecommunication, automation and informatics) at the University of Würzburg. Before, he headed as an assistant professor the Automation group at Jacobs University Bremen and was a research associate at the University of Osnabrück. Further, past affiliations were with the Fraunhofer Institute for Autonomous Intelligent Systems (AIS, Sankt Augustin) and the University of Bonn, from which he received the diploma degree in computer science in 2002 (best paper award by the German society of informatics (GI) for his thesis). He holds a doctorate degree (Dr rer. nat) from University of Bonn. His work is focused on sensor systems, computer vision, robotics, and automation. His main research interests include 3D environment mapping, 3D vision, and laser scanning technologies, resulting in fast 3D scan matching algorithms that enable robots to map their environment in 3D using six degrees of freedom poses. The capabilities of these robotic SLAM approaches were demonstrated at RoboCup Rescue competitions, ELROB, and several other events.

References

[1] Holz D. Autonomous exploration and inspection. Master's thesis, Hochschule Bonn-Rhein-Sieg, Grantham-Allee 20, 53757. Sankt Augustin, Germany. 2009.

[2] Wurm KM, Stachniss C, Burgard W. Coordinated multi-robot exploration using a segmentation of the environment. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '08); Nice, France: Sept 2008. p. 1160–1165.

[3] Julia M, Gil A, Reinoso O. A comparison of path planning strategies for autonomous exploration and mapping of unknown environments. Auton. Robots. 2012;33:427–444.

[4] Zhao J, Su X, Yan J. A novel strategy for distributed multi-robot coordination in area exploration. In: Proceedings of the IEEE International Conference on Measuring Technology and Mechatronics Automation (ICMTMA 09). Vol. 2, 2009; Zhangjiajie. p. 24–27.

[5] Burgard W, Moors M, Stachniss C, Schneider FE. Coordinated multi-robot exploration. IEEE Trans. Robot. 2005;21:376–386.

[6] Digor E, Birk A, Nüchter A. Exploration strategies for a robot with a continously rotating 3d scanner. Proceedings of the Second International Conference on Simulation. Modeling and Programming for Autonomous Robots (SIMPAR '10), Lecture Notes in Computer Science; Darmstadt, Germany. Nov 2010. p. 374–386.

[7] Al-khawaldah M, Nüchter A. Multi-robot exploration and mapping with a rotating 3D scanner. In: Proceedings of the 10th International IFAC Symposium on Robot Control (SYROCO '12). Vol. 10. Dubrovnik, Croatia, Sep 2012.

[8] Yamauchi B. A frontier-based approach for autonomous exploration. In: Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA '97); 1997; Monterey, CA. p. 146–151

[9] Scott AF, Yu C. Cooperative multi-agent mapping and exploration in webots r. In: Proceedings of the 4th International Conference on Autonomous Robots and Agents (ICARA '09); 2009; Wellington. p. 56–61.

[10] Sheng W, Yang Q, Tan J, Xi N. Distributed multi-robot coordination in area exploration. Rob. Auton. Syst. 2006;54:945–955.

[11] Yamauchi B. Frontier-based exploration using multiple robots. In: Proceedings of the second International Conference on Autonomous Agents. 1998; Minneapolis, MN. p. 47–53.

[12] Zipparo V, Kleiner A, Nebel B, Nardi D. Rfid-based exploration for large robot teams. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '07); Rome, Italy. May 2007. p. 4606–4613.

[13] Choset H, Burdick J. Sensor-based exploration: the hierarchical generalized voronoi graph. Int. J. Robot. Res. 2000;19:96–125.

[14] Thrun S. Learning metric-topological maps for indoor mobile robot navigation. Artif. Intell. 1998;99:21–71.

[15] Zivkovic Z, Bakker B, Kröse B. Hierarchical map building and planning based on graph partitioning. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '06); 2006; Florida, FL. p. 803–809.

[16] Friedman S, Pasula H, Fox D. Voronoi random fields: extracting topological structure of indoor environments via place labeling. In: Proceedings of the International Conference on Artificial Intelligence (IJCAI '07); 2007; Hyderabad.

[17] Kuipers B, Byun Y-T. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. J. Rob. Auton. Syst. 1991;8:47–63.

[18] Choset H, Burdick J. Sensor based planning, part I: the generalized voronoi graph. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '95); Nagoya, Japan. 1995.

[19] Meijster A, Roerdink JBTM, Hesselink WH. Mathematical morphology and its applications to image and signal processing, chapter A general algorithm for computing distance transforms in linear time; Kluwer Acadeic Publishers. 2000; Dordrecht. p. 331–340.

[20] Beeson P, Jong NK, Kuipers B. Towards autonomous topological place detection using the extended voronoi graph. In: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '05); 2005.

[21] Holz D, Basilico N, Amigoni F, Behnke S. Evaluating the efficiency of frontier-based exploration strategies. In: Proceedings of Joint 41th International Symposium on Robotics and 6th German Conference on Robotics; Munich. June 2010.