Multi-Robot Fire Searching in Unknown Environment

Ali Marjovi, João Gonçalo Nunes, Lino Marques and Aníbal de Almeida

Abstract Exploration of an unknown environment is a fundamental concern in mobile robotics. This paper presents an approach for cooperative multi-robot exploration, fire searching and mapping in an unknown environment. The proposed approach aims to minimize the overall exploration time, making it possible to locate fire sources in an efficient way. In order to achieve this goal, the robots should cooperate in an effective way, so they can individually and simultaneously explore different areas of the environment while they identify fire sources. The proposed approach employs a decentralized frontier based exploration method which evaluates the cost-gain ratio to navigate to target way-points. The target way-points are obtained by an A* search variant algorithm. The potential field method is used to control the robots' motion while avoiding obstacles. When a robot detects a fire, it estimates the flame's position by triangulation. The communication between the robots is done in a decentralized control manner where they share the necessary data to generate the map of the environment and to perform cooperative actions in a behavioural decision making way. This paper presents simulation and experimental results of the proposed exploration and fire search method and concludes with a discussion of the obtained results and future improvements.

1 INTRODUCTION

Search operations inside buildings, caves, tunnels and mines are sometimes extremely dangerous activities. The use of autonomous robots to perform such tasks in complex environments will reduce the risk of these missions. Exploration of an unknown environment is a fundamental issue in mobile robotics. As autonomous exploration and map building becomes increasingly robust with a single robot, the next stage is to extend these techniques to teams of robots. Using multiple robot systems may potentially provide several advantages over single robot systems, namely speed, accuracy, and fault tolerance [1], [2], [3] and [4] . Nowadays, swarm based exploration and mapping where the robots can be smoothly added or removed to the operation is an area with increasing interests to the robotics community [1].

This study is a part of a European project named Guardians[1]. The Guardians are a swarm of autonomous robots applied to navigate and search an urban environment. The project's central example is search and rescue in an industrial warehouse

The authors are with the Institute for Systems and Robotics, Department of Electrical and Computer Engineering, University of Coimbra, 3030-290 Coimbra, Portugal. {ali,jgnunes,lino,adealmeida}@isr.uc.pt

[1] http://www.guardians-project.eu

in smoke, as proposed by the Fire and Rescue Service of South Yorkshire. The job is time consuming and dangerous; toxins may be released and human senses can be severely impaired. They get disoriented and may get lost. The robots warn for toxic chemicals, provide and maintain mobile communication links, infer localization information and assist in searching. Map exploration and fire source detection are the topics in this paper.

The problem of coordination and control of multiple robots for mapping and exploration has been almost addressed through several research approaches. Most approaches rely on centralized control to direct each vehicle in the swarm. This centralized approach has been popular in the robotics community, because it allows near optimal behaviours in well understood environments. However, its performance decreases in new unidentified environments. Yamauchi [5] proposed a distributed method for multi-robot exploration, yielding a robust solution even with the loss of one or more vehicles in the swarm. A key aspect of this approach involves sharing map information among the robotic agents so they execute their own exploration strategy, independently of all other agents. While this technique effectively decentralizes control, exchange of map information is not enough to prevent inefficient cooperative behaviours. This approach also required known starting positions and failed to provide a robust mechanism for map merging [6].

Simultaneous localization and mapping (SLAM) has been a topic of much interest because it provides an autonomous vehicle with the ability to discern and represent its location in a feature rich environment. Some of the statistical techniques used in SLAM include Kalman filters, particle filters (Monte Carlo methods) and scan matching of range data. But if there is a local or global localization system where robots know their relative positions, SLAM techniques are not required.

There are some proposed methods for exploration based on cooperation between agents. Several researchers have suggested stigmergy methods [7] and [6]. The central theme of the study presented in [7] is the use of stigmergy to achieve effects-based control of cooperating unmanned vehicles. They accomplished stigmergy through the use of locally executed control policies based upon potential field formulas. Nevertheless, this method is mainly useful when there are a lot of small robots working together.

Most of the existing approaches to coordinate multi-robot exploration assume that all agents know their locations in a shared (partial) map of the environment. Effective coordination can be achieved by extracting exploration frontiers from the partial map and assigning robots to frontiers based on a global measure of performance [1], [2], [3] and [8]. As illustrated in Fig. 1, frontiers are the borders of the partial map, between explored free space and unexplored areas [2]. These borders, thus, represent locations that are reachable from within the partial map and provide opportunities for exploring unknown terrain, thereby allowing the robots to greedily maximize information gain [9]. Compared to the problems occurring in single robot exploration, the extension to multiple robots poses new challenges, including:

Coordination and cooperation: Since there are several robots working in the same area, they must have some kind of cooperation with each other in order to prevent collisions and share tasks. Effective cooperation can be achieved by guiding

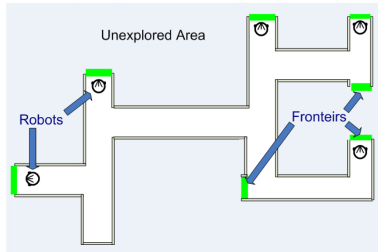

Fig. 1 Frontiers base exploration

the robots into different non-overlapping areas [2], [3], [10]. The idea is that at a given time each robot should be dedicated to exploring one and only one frontier.

Integration of information collected by different robots into a single map: The main goal of exploration is to build a general map representing the environment. The robots should integrate all the data into a single map. Map merging is a big challenge in this field that has been address in several studies [11].

Uncertainty in localization and sensing: The effect of sensor errors ("noise") and errors in sensing the gradient of a "resource profile" (e.g., a nutrient profile) should be considered. Several researchers have illustrated that the agents can forage in noisy environments more efficiently as a group than individually [12], [13].

Decision making, reasoning, task sharing and navigation: Decision making for each robot in an unknown environment is a very complex problem. Since nobody knows what lies beyond the frontier of an unexplored area, there is no unique optimum algorithm that is completely reliable. In each situation, a robot should make a decision to progress exploring task based on partial existing map and also the other robots' positions and objectives.

In terms of decision making algorithms, a lot of studies for multi-robot exploration do not address unknown environments. Moreover, most of the research in this field is based on centralized control of the agents. More significant approaches for multi-robot exploration have been presented in [14] and [5]. In both techniques, the robots share a common map which is built during the exploration. Singh and Fujimura [14] presented a decentralized online approach for heterogeneous robots. Most of the time, the robots work independently. When a robot finds a situation that is difficult to resolve by itself, it will send the problem to another robot which is likely to be able to resolve the situation. The candidate robot is chosen by trading off the number of areas to be explored, the size of the robot and the straight-line distance between the robot and the target region. This technique generates a grid geometric map; therefore, the accuracy of the map depends on the grid size. Moreover, all the robots need to have a huge memory to keep the entire map. In the approach of Yamauchi [5], the robots move to the closest frontier, which is the closest unknown area around the robot according to the current map. However, there is no coordination component which chooses different frontiers for the individual robots.

Our approach, in contrast, is specifically designed to coordinate the robots so that they automatically do not choose the same frontier. It generates a topological map which needs much less memory capacity. As a result, this method needs significantly less time to accomplish the task. The objective in this research is to generate the map of an unknown environment and also localize all the fire sources in the area. In fact, the final future goal is to create a fire risk map of an unknown environment with multiple robots. We do not address the problems of risk maps here.

During the exploration process, if there is a fire source, robots should report it. The authors have addressed this issue in previous papers [15], [16], [17] and [18]. The last achievement of that research is kheNose. The kheNose is a device developed by the authors to sense olfactory information through the use of gas sensors, anemometers, a temperature and humidity sensor [19]. In the current study, the last version of kheNose has been used to detect the fire sources.

Collision avoidance between the robots during the exploration is a considerable issue that has not been addressed pragmatically in the previous studies. In this study, we propose a new practical method for multi-robot unknown environment exploration with fire source detection which takes "collision avoidance" and "task sharing" into consideration. This method has been tested in the real world and also in simulation. The effect of complexity of the environment and also the numbers of robots are the main parameters that have been studied in this paper.

2 The Proposed Method

This section explains the concept of the proposed multi-robot cooperation technique. This method is illustrated in schematic diagram Fig. 2, in a global perspective. As shown in the diagram, the method includes three main tasks: navigation and exploration, decision making, and fire source detection. The following parts describe some of the elements of the diagram (Fig. 2).

2.1 Decision making

The main goal of the exploration process is to cover the whole environment in the minimum possible time. Therefore, it is essential that the robots share their tasks and individually achieve the objectives through optimal paths. In an unknown environment, the immediate goals are the frontiers. Most of the time, when the robots are exploring the area, there are several unexplored regions, which poses a problem of how to assign specific frontiers to the individual robots. We want to avoid sending several robots to the same frontier, which may result in collision concerns. The other issue is that we do not want to have a central station for moderating the robots. To address these problems, the proposed method is based on a decision-theoretic exploration strategy. The decision making algorithm is shown in Algorithm 1.

The frontier is selected based on the cost of reaching it and the utility it can provide to the exploration. The cost is calculated through the A* method which simultaneously determines the optimal path that has to travel to reach the frontier and its distance. The cost is proportional to the distance that the robot has to pass to reach the frontier.

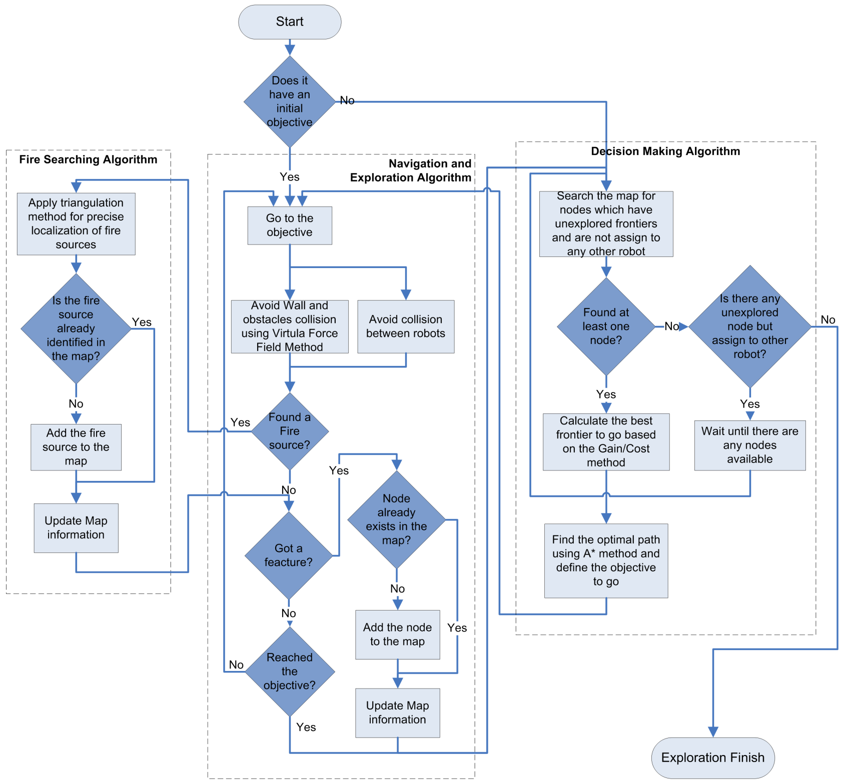

Fig. 2 The entire system schematic

The utility depends on the number of the robots and their proximity to the frontier, which means that if there are several frontiers at similar distances, the robot will go to the one that has higher utility. This procedure will make the robots disperse and explore the environment in a more efficient way.

2.2 *Task sharing and map generation*

The cooperation between the robots permits the exchange of data, allowing for task sharing and, consequently, an efficient distributed exploration. During the exploration, there is only one global shared map in the system. This map is in a server that sends and receives the map to the robots whenever they request it. Within this map, besides having some information regarding the kind of nodes and their position, it also has data describing the location of the robots and their frontier target, as can be seen in Fig. 3. Through this data, a robot can see which frontiers are unexplored, their position and if any robot has targeted them as its objective, thus allowing a distributed efficient exploration (see Algorithm 1 and Fig. 2).

Algorithm 1: Map exploration algorithm - Decision making method

```
Receive_map_from_server()
while there is at least one unexplored link in the map do
    do Follow the potential field algorithm until (getting a different feature in the environment)
    Receive_map_from_server()
    if the new node exists in the map then
        Update the map's data with new information.
        Send_map_to_the_server()
    else
        Add_new_node_to_map()
        Send_map_to_the_server()
    if the current node has any unexplored link then
        Calculate direction gain of taking each unexplored link, based on the position of the other robots and
        get the higher gain direction to follow
    else
        Determine the best not-assigned unexplored frontier, based on their gain / cost.
        Assign the frontier to the robot.
        Calculate the best path to take, based on the A* algorithm and get that direction to follow.
End of algorithm
```

Algorithm 2: Collision avoidance between the robots(and increasing efficiency of collaborating)

```
Calculate a confined circular area around the robot
if any other robot is inside the circle then
    Determine if the robots are in a collision pattern
    if there is a possibility of being in a collision pattern then
        //Follow the rules of engagement:
        if they are in a direct collision path then
            reevaluate their goals.
        else
            if they are both currently exploring frontiers OR they are both moving inside explored
            area then
                give priority to the one which has lowest ID.
            else
                Give priority to the one that is exploring a frontier.
else
    Continue exploration algorithm.
End of algorithm
```

In dealing with multiple robots in one environment, collision between robots is a very important aspect. For instance, two robots might be in a narrow corridor with different directions and they may want to pass but cannot because they are facing each other or they may even treat each other as a dead end. It is necessary to avoid these types of problems. Therefore, we have implemented some rules of engagement justly in order to prevent these types of situations. If two robots are going against each other in a corridor, it's because the task sharing has not been done previously in a efficient way. These rules of engagement are shown in Algorithm 2.

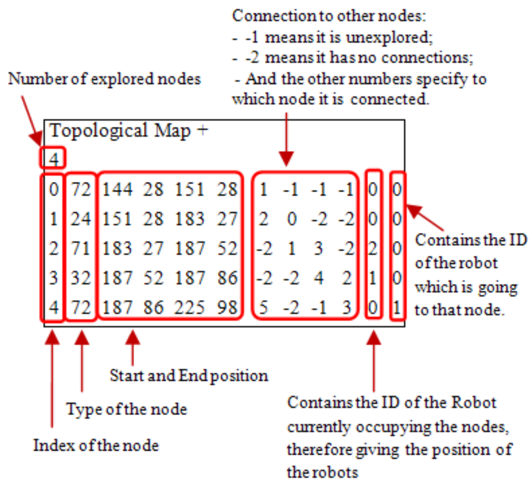

Fig. 3 Example of topological map data

Fig. 4 Khepera III and kheNose

2.3 Fire source detection

During exploration and navigation, the robots are simultaneously acquiring information from the environment (see Fig. 2). All the robots are equipped with a board developed by this research group, which integrates temperature and chemical sensors named kheNose (Fig.4).

When the robots are mapping the environment, they are constructing the map and verifying if the current node they have acquired is not already on the map, thus, assuring the coherence of the map and making the merging process simple, where most of the time it is only necessary to add new nodes to the global shared map.

They use an eight element thermopile array sensor to measure the absolute temperature as well as the ambient temperature on the robot to be able to distinguish the heat values. When the temperature data received from the sensor informs that there is an increase in the temperature or any evidence of existence of a fire source in the environment, the robot identifies it as a heat source clue and proceeds to make a sequence of movements to acquire more accurate values in order to be certain that it isn't a random value. After the fire source is determined, the robot performs a triangulation method to estimate its actual position.

3 Experiments

As a test plan, we built several maze-like environments which are combinations of corridors, corners, crossings, T-junctions and dead-ends (one of them is shown in Fig. 5). Three Khepera III robots are used for testing the algorithm. These Khepera III robots are able to run the algorithms mentioned in the previous section.

Fig. 5 Real maze experiment

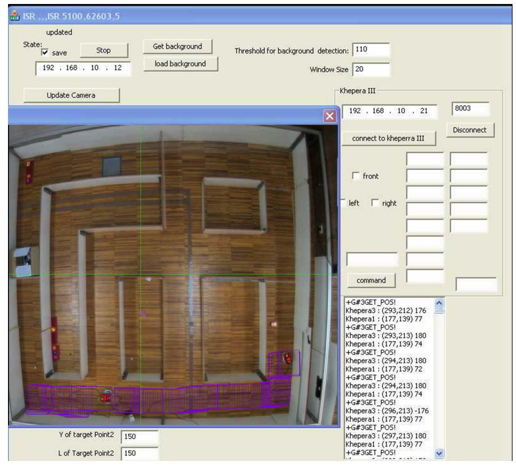

Fig. 6 Visual positioning application screenshot

Robots should know their positions in the environment to be able to explore and navigate. Therefore a network camera is mounted on the top of the environment and an image processing computer program is able to track and locate each robot. Each robot has two coloured labels on the top that can be seen by the camera. The camera is connected to the network and an image processing program tracks the robots' position and provides the absolute position of each robot via wireless network. Image processing program is an object tracking application developed by the authors. By recognizing the center of each coloured label and calculating the line crossing from these two centers, the orientation of the robot can be computed. The program is written in C++. Fig.6 shows a screen shot of this program.

In terms of feature extraction, based on values measured by sonar and infrared sensors, the robot recognizes the features and should take an action and modify the shared map; it will save this data in the map structure as a new node, and will also update the data related to the previous feature. For each feature, the robot saves the data in the topological map, including start position and end position of that node and some other information (Fig.3).

The system has been tested with different start positions for the robots in different maze structures. There is a small candle acting as a heat source in the environment which robots try to locate. All of the robots are equipped with kheNose boards for heat source detection.

The algorithm has been tested in real world and also in a simulation world. For optimizing the exploration algorithm and measuring its performance, we used the Player/Stage simulator [20]. In the real world, there are a lot of constraints that do not allow for testing the proposed method very easily. It is not effortless to build various test plans with different scales for testing and developing the method. Since there is no reliable simulator for fire and smoke in Player/Stage, the whole system has been tested in the real world.

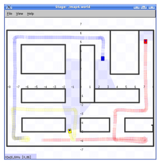

Fig. 7 A maze with 34 nodes

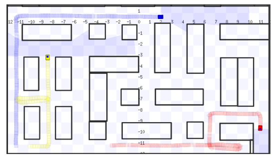

Fig. 8 82 nodes

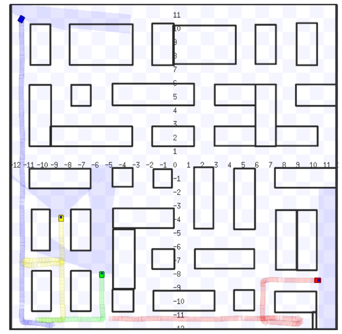

Fig. 9 135 nodes

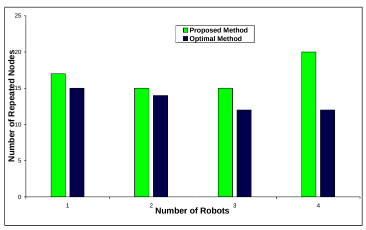

Fig. 10 Number of repeated nodes, Comparing the results of the proposed method with optimal method

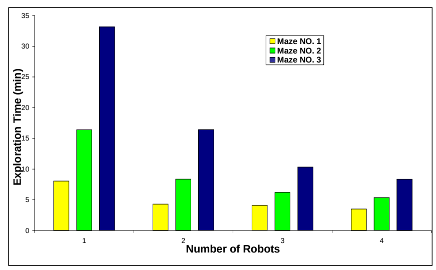

Fig. 11 Test of various numbers of robots against complexity of the environment, 1: maze in Fig.7, 2: maze in Fig.8, 3: maze in Fig.9

Fig. 5 shows two robots exploring a small maze and finding a fire source. Both robots started from the same point but not at the same time. We intentionally ran one of the robots a few seconds after the first one. The red footprint shows the first robot's path and the blue footprint is related to the second robot. As shown, the first robot found the fire source. For an example of the coordination algorithm, when the second robot reached the junction it figured out that the path in the front was already explored and it chose the right path.

Since there is no accepted standard benchmark, measuring the performance of a behavioural based multi-robot unknown area exploration algorithm is a very difficult job. One of the possible ways to do that is to compare the proposed method with the optimal possible method. But the issue is that there is no optimal method for exploring an unknown world. However, there is an optimal solution for minimizing the travelling path if the world (maze) is completely known before exploration.

The algorithm has been tested with different number of robots in a specific maze. The model of that maze is also given to the optimal method and then we compared the results of the proposed algorithm with the optimal method. Since the optimal method has the world's model but the proposed method is exploring the unknown world, it is obvious that the results of the proposed method are always worse than optimal method but it can be a good criteria for evaluating the method.

The number of repeated nodes during travel can be a good parameter for measuring the performance of the method. A repeated node is a node that robots pass more than once. Fig. 10 shows the number of nodes that have been repeated more than once in the optimal method as well as in the proposed algorithm for the maze shown in Fig. 7. A good conclusion from the graph in Fig. 10 is that there is a trade-off between the number of robots and the size of the world. It shows that the proposed approach is acceptably comparable with the optimal method.

Another parameter for evaluation of the method is the exploration time. The proposed method has been tested with a different number of robots in different mazes. The environment shown in Fig. 6 that is a 3.5 x 4 meters maze is tested by one, two and three Khepera robots separately. One robot could explore the environment in 412 seconds. This environment has been explored by two robots in 254 seconds. The exploration time for the same maze with three Kheperas was 212 seconds. Each result is the average of five similar tests. Different tests with constant conditions had similar results with about seven percent variance. In all the tests, the maximum speeds of the Kheperas are kept constant.

In simulation, the mazes shown in Fig.7, Fig.8 and Fig.9 have been tested separately with one, two, three and four robots and the results are shown in Fig.11. The graph shows the average of five tests for each data. The variance was less than one percent. It is obvious that, with more robots, exploration time will be improved. Another conclusion from the graph is that having more robots is more advantageous in a complex maze than in a simple maze. This also proves that the cooperation algorithm in this approach is efficiently functional.

In the real experiment the fire source detection was also tested and the robots could locate the fire sources during the exploration. The performance of fire source detection has been addressed in previous studies by the group [17], [19].

4 Conclusions and future works

The proposed method for multi-robot unknown environment exploration has been implemented and experimented in the test plans. The robots are able to cooperate and create a shared topological map of the unknown environment. Cooperation between the robots is done by sharing information in the shared map. The algorithm has been tested against several different configurations in Player/Stage simulation program. The exploration algorithm is merged with fire source detection algorithm and has been tested in the real world. The effect of the number of the robots on exploration in different type of environment has been discussed. The results show the efficiency and reliability of this method.

In terms of implementation, more accurate sonar sensors should be installed on the robots. The visual localization system provides many constraints for testing, and is thus being replaced by a non-centralized localization system that integrates odometry with gyroscopes and accelerometers as well as an inbuilt compass.

The system works only in a maze like environment, in future it should work in a real environment with less restrictions. For this purpose, feature extraction should be more developed, range sensors should be more accurate and the topological map should contain some geometric data. Perhaps a mixture of topological and geometric maps will be useful in this case.

References

1. D. Fox, J. Ko, K. Konolige, B. Limketkai, and B. Stewart, "Distributed multi-robot exploration and mapping," *Proc. of the IEEE, Special Issue on Multi-Robot Systems*, p. 1325, 2006.
2. W. Burgard, M. Moors, D. Fox, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," *Proc. of the IEEE, ICRA*, 2000.
3. W. Burgard, M. Moors, C. Stachniss, and F. Schneider, "Coordinated multi-robot exploration," *IEEE Trans. on Robotics*, vol. 21, no. 3, pp. 376–386, 2005.
4. G. Dedeoglu and G. Sukhatme, "Landmark-based matching algorithm for cooperative mapping by autonomous robots," *Proc. of 5th Int. Symp. on Distributed Autonomous Robotic Systems (DARS)*, 2000.
5. B. Yamauchi, "Frontier-based exploration using multiple robots," *Proc. of 2nd Int. Conf. on Autonomous Agents*, 1998.
6. J. Stipes, R. Hawthorne, D. Scheidt, and D. Pacifico, "Cooperative Localization and Mapping," *Proc. of the IEEE on Networking, Sensing and Control*, 2006.
7. D. Scheidt, J. Stipes, and T. Neighoff, "Cooperating Unmanned Vehicles," *Proc. of the IEEE on Networking, Sensing and Control*, 2005.
8. S. Koenig, C. Tovey, and W. Halliburton, "Greedy mapping of terrain," *Proc. of the IEEE, ICRA*, 2001.
9. R. Zlot, A. Stentz, M. Bernardine Dias, and S. Thayer, "Multi-robot exploration controlled by a market economy," *Proc. of the IEEE, ICRA*, 2002.
10. M. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (SLAM) problem," *IEEE Trans. on Robotics and Automation*, vol. 17, no. 3, 2001.
11. M. Pfingsthorn and A. Birk, "Efficiently communicating map updates with the pose graph," *Proc. of the IEEE/RSJ, IROS*, pp. 2519–2524, 2008.
12. Y. Liu and K. Passino, "Stable Social Foraging Swarms in a Noisy Environment," *IEEE Trans. on Automatic Control*, vol. 49, no. 1, 2004.
13. Y. Liu and k. Passino, "Biomimicry of social foraging behavior for distributed optimization: Models, Principles and emergent behaviors," *Journal of Optimization Theory and Applications*, vol. 115, no. 3, pp. 603–628, 2002.
14. K. Singh and K. Fujimura, "Map making by cooperating mobile robots," *Proc. of the IEEE, ICRA*, 1993.
15. L. Marques, N. Almeida, and A. Almeida, "Olfactory sensory system for odour-plume tracking and localization," *Proc. of IEEE on Sensors*, vol. 1, 2003.
16. L. Marques and A. Almeida, "Electronic nose-based odour source localization," *6th Int. Workshop on Advanced Motion Control*, 2000.
17. L. Marques, N. Almeida, and A. Almeida, "Particle swarm-based olfactory guided search," *Springer journal of Autonomous Robots*, vol. 20, no. 3, pp. 277–287, 2006.
18. L. Marques, U. Nunes, and A. Almeida, "Olfaction-based mobile robot navigation," *Elsevier journal of Thin Solid Films*, vol. 418, no. 1, pp. 51–58, 2002.
19. J. Pascoal, P. Sousa, and L. Marques, "KheNose - A smart transducer for gas sensing," *Proc. of the 11th Int. Conf. on Climbing and Walking robots and the support technologies for mobile machines, CLAWAR*, 2008.
20. B. Gerkey, R. Vaughan, and A. Howard, "The player/stage project: Tools for multi-robot and distributed sensor systems," pp. 317–323, 2003.