International Journal of Trend in Scientific Research and Development (IJTSRD)
International Open Access Journal | www.ijtsrd.com
ISSN No: 2456 - 6470 | Volume - 2 | Issue – 6 | Sep – Oct 2018

Homogeneous Swarm Robots Exploration

W. A. F. W. Othman, M. A. Rosli, A. A. A. Wahab, S. S. N. Alhady
School of Electrical & Electronic Engineering, Universiti Sains Malaysia,
14300 Nibong Tebal, Penang, Malaysia

ABSTRACT

Exploration process in a real application seems had attracts the interest from researchers and academics nowadays. The usage of robots rather than human themselves give many benefit in term of security and safety, also in time and energy consumption. This project aims to generate a code represent the exploration technique in an unknown environment for homogeneous multi-agents systems. In multi-agents system, there are many variable involve in the result of the exploration process. The number of the agents, network connectivity, type of hardware, the environment and other aspects must be considered in multi-agents system exploration investigation. However, due to time constraint, this project only studies the effect of the number of robots or agents in the exploration task. The technique chose for the task must have a certain rules and assumptions to make the exploration process run smoothly and efficient. The code has been developed by using NetLogo due to the availability of sample models and toolboxes. Recording of the data was also done through the use of NetLogo. Finally, the data is arranged by using Microsoft Excel to extract the recorded data for analysis process. The usage of Minitab software is used to plot the graph. Hopefully, this system can be used in real application with the appropriate hardware.

Keyword: *component; multi-agent system; exploration technique; path; obstacle; explored; unexplored*

INTRODUCTION

Since the human have been made, the Earth have been explored by human themselves. The task to explore the unexplored environment is a must in order to develop new information for human’s benefit. However there are regions under dangerous or difficult circumstances such as underwater exploration, outer space or radioactive distractive remainders. The sacrifice is needed for those kinds of tasks, and to prevent this situation human have made a machine as a new agent to do this work. The main reasons of the usage of these intelligent machines, or simply name it as robot; are to protect the human life and to help human in a daily routine life. The challenge for a robot in exploring the environment is a fundamental problem. The example of exploration applications like cleaning [1, 2], underwater exploration, space exploration [3], or rescue belongs to the common parts of robotic mission nowadays.

Over the past decades, the research and development of computer robotic systems has been actively pursued. One best example of the use of the applications is automatic vacuum cleaning robots. However the commercial of the vacuum cleaning robots is still lacking in several aspects. The most problematic is the inefficiency of the exploration technique of this machine. The vacuum cleaner might clean the same region for many times, leading to time-wasting problem, expendable of cost and power consumption. Therefore to decrease time, a new method had been implied by using more than one vacuum cleaner in the environment. But this method lead to another problems, that are;

a) the vacuum cleaners do not know the area explored by another vacuum cleaner b) the vacuum cleaners detect each other as an obstacle

The problems have studied by researcher and they conclude; the robots need to update the knowledge about the environment and share their information at same time. Nowadays, the commercial like vacuum cleaner is still not perfect, because of the technique for exploration and mapping is still in development process. These techniques are not easy to implement in commercial robots because it use imprecise sensors for economic reasons. Moreover, the commercial robots might not be able to process the technique because the requirement of large amount of memory.

A good exploration strategy is needed in order to make multi-robot accomplish their task in less time. The exploration strategy must permit the complete coverage of an environment, even in a situation in which new knowledge about the environment is incrementally acquired by mobile robots during the exploration. A requirement for an exploration strategy is it must be able to be implemented to mobile robots in actual exploration mission.

The exploration task always comes with mapping. Mapping the environment is a basic challenge in mobile robots where it leads to benefit in future plan. Previously, most papers in exploration and mapping only dealt with a single agent systems, which we have already knew that the systems is less advantage than multi-agent systems. While solving the problems that appear in this project, an approach that had been performed is quite well.

In this paper, the problem of exploration technique using homogeneous multi-agent in an unknown environment is considered. The usage of multi-agent systems has many advantages over single agent systems [4 - 11]. The main advantage by using multi-agent is the task can be accomplished faster than using a single agent. Besides that, the usage of multi-agent is more fault-tolerant than a single agent [12]. Another advantage of using multi-agent is the merging of agents' knowledge, which leads to avoid exploring the same area in a large amount of time. However, there are also some disadvantages by using multi-agent. One of the disadvantage is the usage of multi-agent require more power consumption. *N* number of multi-agent need *N* times the power used by a single agent. Although multi-agent complete the task in less time than a single robot, the power consumption must be considered also because more power used, more cost needed to do the task in real application. Moreover, longer path may be needed to avoid collision between the agents [12].

II. Related work

After a review of different approaches for multi-robot exploration, we conclude that the solutions are subscribed depend on the strategy chose for robots movement. Basically there are two types of movement or strategies [11]. The first type of strategies uses non-structured trajectories [13 - 17], where the navigation of the robots depends on the search of the best next point of view that drives in the elimination of borders of the unknown world, or, by means of probabilistic methods. The other type involves structured trajectories [7, 8, 18 - 21], where the movement is basically following *zig-zag* or spirals like paths.

A. Coordinates and Grids System

Yamauchi's approach [4] generally improved the coordination between robots. The Frontier-based exploration allowed the robots to gain and share new information about the environment. The strategy used in the paper is called frontier-based exploration. The robots explore and increase the knowledge by moving to successive frontiers. The evidence grid is used as their spatial representation. From the practice, the team has decided to use laser-limited sonar rather than raw laser because laser-limited sonar reduces the error specula reflection from the large obstacles such as wall.

Each robot has its own global evidence grid that represents its knowledge about the environment. When a robot arrives at a new frontier, it will construct an evidence grid representing its current surroundings. This local grid is integrated with the robot's global grid, that's make the knowledge can be shared to all of the other robots. Each robot will have other robots' local grids. This approach make robot use the information from other robots to help in their own exploration path. Therefore, the robots can explore more effective and most important is the time exploration is reduced successfully because the robot didn't wasting the time for exploring the wrong path or environment. However the position of the path can be error if the mapping is not accurate especially for larger environment. To overcome the problem, localization is useful for building accurate maps.

Vazquez and Malcolm [22] pressed that an exploration algorithm is based on certain objectives: to avoid obstacles, to maintain communication between robots and to explore around the frontier. By using this approach, the connectivity of the network is taken into account. To achieve this, each robot must analyse the topology of the network. The environment is represented by means of a global probabilistic grid map. Each robot will share their information about their position and their current heading movement. The other robots receive the information from their neighbors directly and from the rest of the team. Form this is formation, the robots can identify the topology of the network. This approach is very helpful in short range connectivity, but the authors main concern is the capability for a large amount of robots and environment because the connection network might have an error in data transmission and to make it worst, the network is disconnect.

The technique presented by Burgard *et al...* [6] extends the Simmons' efforts in several ways. First, the approach described here distributes the computation, to a large extent. This enables the robots' "bids" to be calculated in parallel, which facilitates scaling to larger numbers of robots and enables the robots to construct bids based on their own capabilities (travel cost, sensor range, etc.). Second, the current method uses a more sophisticated notion of expected information gain that takes current map knowledge and the robots' individual capabilities into account. This allows for more subtle types of coordination, for example, allowing the robots to remain near one another if the map shows that they are separated by a solid wall. In this approach, the team use occupancy grid to maps the environment. The explored area will kept in memory to identify next possible target location. Since the robots did not know the knowledge about the environment, the area is estimated which is explored by the robot's sensors when it reaches the target. Based on this information, different target positions are chose for other robots.

B. Probability Strategy

In Yamauchi approach [4], the evidence grid is used as their spatial representation. Each cell in the grid is differentiated by comparing its probability assigned to all cells. There are three classes in the cell, which are open, unknown and occupied. Those classes depend on the probability of occupancy and prior probability. Based from the probability result, the mobile robot will move to the chosen grid or cell.

- open: occupancy probability < prior probability
- unknown: occupancy probability = prior probability
- occupied: occupancy probability > prior probability

Burgard *et al.*. [6] state that the assumption that had been made is the robots only knows their related positions when exploring the environment. While the robots explore, they constructs the map of the environment at the same time. The robot will estimate the expected area that will be explored on the next step. To determine the cost of reaching the current frontier cells, the optimal path is computed from the current position to the frontiers. The computation is based on deterministic variant of value iteration. The cost for traversing a grid cell is proportional to occupancy value. The minimum-cost path is computed using the following two steps, those are initialization and update loop.

C. Network Connectivity and Knowledge Sharing

Simmons *et al.* [23] was the first team to introduce the concept of information gain for exploration algorithm. The approach prevents other mobile robots to select the same target location by coordinating the explored path of the mobile robot, which facilitates the reduction of the exploration time and interference among the robots. The target point of exploration is chosen based on the path length from the mobile robot's current position to the target point and it's utility to obtain the new information after reaching the target point. However, the approach always assigns that target location to a robot that has the trade-off between the amount of new information of the location and the travelling cost for the robot to reach this location, which is greedy and could result in overall inefficiency of the mission.

Rooker and Birk [5] improved Simmons approach. They proposed a centralized coordination ensuring that, during the exploration, no robot will lose the connection with the rest of the robots. To achieve this goal, a central entity collects the current positions of all robots and generates a next possible positions if the robots. But due to high number of robots used, maximize the memory of each robot; all configurations cannot be considered but only a limited number of them. Among this number of generated configurations, the central entity chooses the best one according to utility function. For the worst case scenario, when the central entity fails or disconnections occurred. Moreover, when considering large environment, the use of a central entity might have a problem in finding a central point to concentrates the data from all robots.

III. Homogeneous multi-agent exploration technique

A. Methodologies

The project consist three different environments or arenas that used in analysis. As can be seen in Figure 1, the first arena, Arena 1 represents the blank environment where there is no obstacle at the centre. The second arena has a cross-shape obstacle at the centre while the third arena has an n-shape obstacle. All arenas are bounded. The environments for agents to explore have the same length and width, which is 29 x 32 making the total boxes or we call patches in the arena is 928 patches. The patches then are divided to two items that are path and obstacle. From the figure, there are two colours that can be seen in the arena. The black colour represents the path while the blue colour represents the obstacle. Each patch that been categorized as path can be explored on it by the agents while the obstacle act like a wall where the agents cannot go pass through it. The agents only can detect the obstacle ahead, and then have to avoid it since there is no way to go through.

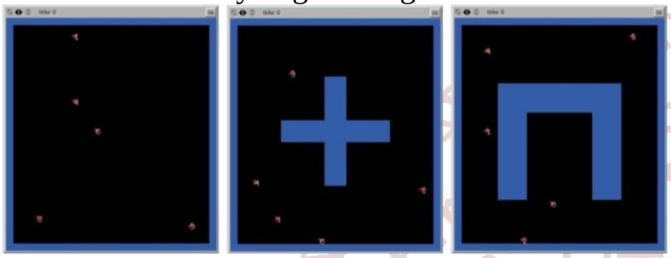


Figure 1: Arena-1 (left), Arena-2 (centre) and Arena 3 (right)

B. Problems and challenges

From other previous papers, there are some problems regarding multi-agent systems. The challenges to overcome those problems are still in research. The goal to be achieved in this task is to complete explore entire environment in minimum time. To achieved this, there are five problems that need to be solved in this project that are;

1. Technique to used

The strategy used in this model is by using coordinates system; and the technique for the agents' movement is random where the agents will turn randomly from -45° to +45° and move step forward until it meet the obstacle where the agents will turn 15° that had been set by the observer.

2. Agents' memorization

The usage of a list of the paths for each agent had solved this problem. In setup procedure, before the model is running, each agent will have their own list named as "pathlist". The list is used to store the paths explored by the agents. The agents actually declared the patches that had been explored as paths or obstacle by changing the colour of the patches. For explored paths, the agents will changed the colour from black to green while the explored obstacles will be yellow in colour compared to unexplored obstacles that are blue in colour. At the same time, the agents will label the path according to number starting from "1" to the total number of paths in the environment. The numbering labelled only done for paths in order to save the memory of the agents. Every time the agents label the path, it will memorize the path and keep it in their list. Each agent had been set to have their own list of the path detected, named as "pathlist".

3. Knowledge sharing

The agents have their network coverage range, called as "territory". Each time two or more agents in the other agent's coverage, the agents will share their knowledge. The agents will add up the other agents' "pathlist" in their "territory" so that both of the agents will have a same "pathlist". For example, let say the first agent, named as "turtle 0" have a list of path as [1 2 3 4 5] and the second agent, "turtle 1" have a list of path as [6 7 8]. When both agents meet each other, they will add up each path in the list, meaning that "turtle 0" and "turtle 1" now have a same "pathlist", that is [1 2 3 4 5 6 7]. After both agents sharing their knowledge, they will separate to different way.

4. Avoid collision

When the agents enter other agent's "territory", both agents will intend to separate each other after they share the knowledge so that they will not explore the same region. This is made to reduce time wasting. The result is made to avoid the collision between two agents. When the agents are entering each other regions, one of the agents will turn facing the back of the agent. This is done by use the calculation where the final angle of heading is the initial angle deducts 180°. Then the agent will move forward if there are no obstacle ahead. While the first agent has to turn and move one step ahead, the other agent only has to turn in random angle.

5. Avoid revisiting

This is the only problem that couldn't be solved in this model. The situation example is shown in Figure 2. The agents have no preferences in choosing the way to unexplored patch unless if the patch is in agent's network coverage. This means that if only the agent exploring near the unexplored area within their territory, the agent will prefer to explore that area. If not, the agents will explore randomly without thinking it should go to the area that has not been explored yet. This solution still has not solved the problem efficiently because if the agents did not explore near to the unexplored area, the area will still remain unexplored. This problem actually leads to revisiting where the agents keep exploring the same area, besides increasing the time taken to complete the task. The problem hopefully can be solved in future work.

Figure 2: Problem

C. Rules, Assumptions and Strategy Used

The technique or algorithm for agents to explore requires some assumptions and rules have been made. The assumptions that have been made are:

1. The agent will only sense one patch around it from its centre
2. The agents know nothing about the environment that they are going to explore

While the rules that have been discussed are:

1. The agents will prefer to avoid entering other agents area
   - For this rule, the agents have been set to have an area of fixed patch in radius
2. The agents prefer to go to unknown environment rather than explored area
   - This is to prevent a time wasted for circling around at same region

IV. Results and discussion

For the experiment, the test involve 25 runs or readings for each number of agents used in exploration process that are 5, 8 and 11 agents while the area coverage by each agents was fixed to 3 in radius. From data that been collected, the table is tabulated and the average time taken to complete explore for each number of agents is measured. Then the closest reading to average data is taken to plot the exploration graph for comparison between each number of agents used. The procedure is similar to those three arenas.

A. Results

Figure 3a-3c Shows Progress View of The Model at 3 Different Ticks Represent The Initial, Running and Final View of Arena-1, Arena-2 and Arena-3.

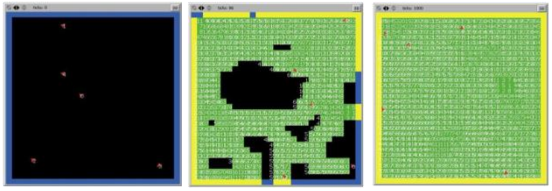


Figure 3a: Arena-1 exploration progress

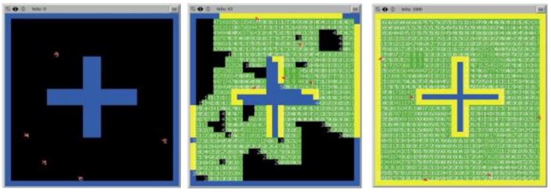


Figure 3b: Arena-2 exploration progress

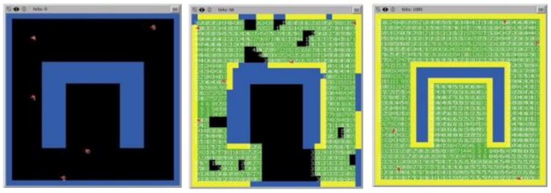
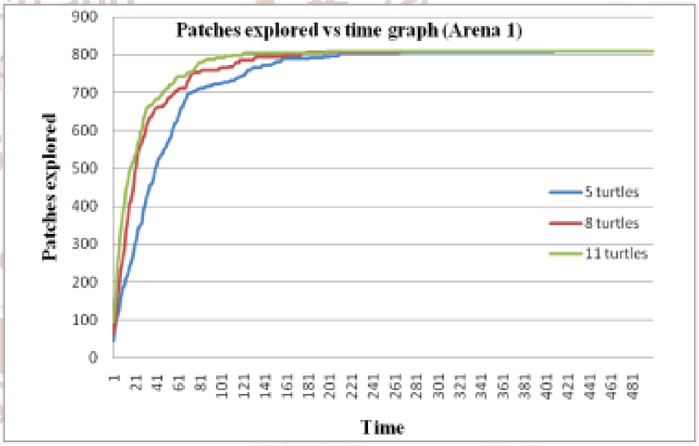
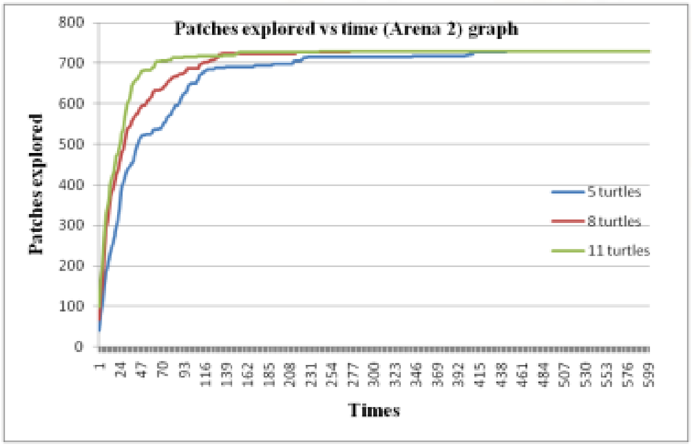


Figure 3c: Arena-3 exploration progress

B. Analysis

As can be seen in Figure 4, the green line that represents exploration using eleven agents has a highest steepness, followed by red line and blue line represent eight agents and 5 agents respectively. This means that by using eleven agents, the number of patches explored at certain time is larger than using eight and five agents. The green line also is the first line to reach the peak of the graph compared to other lines, which tell us that by using eleven agents, the time taken to complete explore all area is lesser than using smaller number of agents. This situation is same to those three arenas.





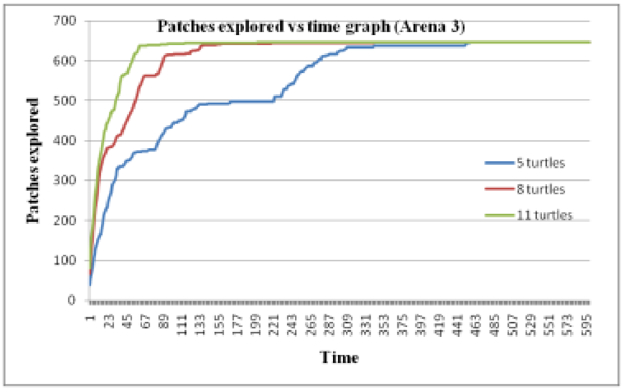


Figure 4: Exploration time of different number of agents for Arena-1 (top left), Arena-2 (top right) and Arena-3 (bottom)

C. Discussion

As can be seen in the final view for Arena-1 from Figure 3, there is no black or blue colour left, meaning that all patches or boxes in the environment already explored by the agents. Compared to Arena-1, the observer can see there is no black colour but there are still blue colours left in final view for Arena-2. This is because the cross-shape obstacle at the centre of Arena-2 has a thickness of three patches but the agents can only sense one patch around it, means that 25 patches or boxes located at the centre of the cross-shape obstacle can't be detected by the agents. The rule has already been discussed in previous section where the agent can only sense one patch around it from its centre. The situation also appears in Arena-3, where the n-shape obstacle at the centre has a thickness of four patches, leaving 78 patches in between unexplored. From a total 928 patches in each arena, Arena-2 and Arena-3 will finish with some unexplored obstacles that cannot be sensed by the agents. Table 1 below shows the condition of the arenas for exploration process.

Table 1: Arenas' exploration conclusion

Environment	Detected patches		Undetected patches
	Paths	Obstacles	
Arena-1	810	118	0
Arena-2	729	174	25
Arena-3	646	204	78

The result obtained from the simulation actually is not good because if we do the analysis, the results have a large standard deviation, meaning that the data is not precise. That's why the data is taken 25 times to increase the precision and accuracy. Each environment has a different type of obstacles, making the time for agents to explore is not same for each arena. As we can see in Figure 4, the variation in number of agents affects the time taken to complete the task. By using smallest number of agents, that is five, the time to complete explore whole environment is longer. This is because each agent has more area to explore individually. This can be proved mathematically to show how this situation happens.

For each agent, the number of patches to explore, *x* is:

$$x = \frac{\text{Total patches to explore}}{\text{Number of agents}}$$

The Table 2 to Table 4 below show the mathematic evidence of the affect of using variable number of agents which lead to the different number of paths to explore for all arenas.

Arena : Arena-1
Total patches to explore : 928

Table 2: Arena-1 mathematical theory

Number of agents	5	8	11
Patches per agent	185.6	116	84.36

Arena : Arena-2
Total patches to explore : 903

Table 3: Arena-2 mathematical theory

Number of agents	5	8	11
Patches per agent	180.6	112.88	82.09

Arena : Arena-3
Total patches to explore : 850

Table 4: Arena-3 mathematical theory

Number of agents	5	8	11
Patches per agent	170	106.25	77.27

Based on those statistics, we can conclude that larger number of agents used in exploration task leads to shorter time to complete the task because there are less number of paths to explore by each agents.

V. CONCLUSION AND SUGGESTION

The study has succeeded in investigating the exploration technique using homogeneous multi-agent system. Basically, the main contribution of this project has been to provide the smooth exploration technique so that the agents are able to complete the task in minimum time frame. From the analysis, it can be seen that by using more agents in exploration process, the task can be accomplished quicker. Based on problems and challenges mentioned in section III, the solutions that have been made are quite well.

The technique is based upon the coordinates system where the robots will memorize and detect the patches as paths or obstacles. Robots share the information about their perception whenever they arrive at new region or area, and they integrate the information from other robots into their own global map. In this way, robots cooperate and use the information from other robots to guide their own exploration.

In this project, the robots prefer to avoid entering other robots' territory and to prevent the exploration at same area. Those solutions have already been explained in section III. However, the sophisticated technique still can be improved to make the exploration process complete in minimum of time.

Future work includes a complete code generated for the application to avoid revisiting by agents. Although the strategy has been included in this project, the revisiting still appears when the agent's did not sense the unexplored area in its network coverage. By completing this problem, the robots will explore the environment smoothly and more efficient, within their own area, sharing the information of the current updated knowledge about the environment until the robots finish exploring the entire world.

ACKNOWLEDGMENT

The authors wish to thank the Universiti Sains Malaysia for partly supported the work by Fundamental Research Grant of MoE Malaysia (Grant number: USM/PELECT/6071239).

REFERENCES

1. M. Jager and B. Nebel, "Dynamic Decentralized Area Partitioning for Cooperating Cleaning Robots". In Proceeding of the IEEE International Conference on Intelligent and Automation (ICRA), pg 3577-3582, 2002.
2. M. Simoncelli, G. Zunino, H. Christensen, and K. Lange. "Autonomous Pool Cleaning: Self Localization and Autonomous Navigation for Cleaning". Journal of Autonomous Robots, Vol. 9, No. 3 , pg 261-270, 2000.
3. D. Apostolopoulos, L. Pederson, B. Shamah, K. Shillcutt, M. Wagner, and W. Whittaker. "Robotic Antarctic Meteorite Search: Outcomes". In Proceeding of the IEEE International Conference on Intelligent and Automation (ICRA), pg 4174-4179, 2001.
4. B. Yamauchi. "Frontier-Based Exploration Using Multiple Robots". In Proceeding of the 2nd International Conference on Autonomous Agents, pages 47-53, 1998.
5. M. N. Rooker and A. Birk. "Multi-Robot Exploration Under the Constraints of Wireless Networking". Control Engineering Practice, pages 435-445, 2007.
6. W. Burgard, D. Fox, M. Moors, R. Simmons and S. Thrun. "Collaborative Multi-Robot Exploration". In Proc. International Conference on Robotics and Automation, 2000.
7. N. Bouraqadi and A. Doniec. "Flocking-Based Multi-Robot Exploration". In Proceeding the 4th National Conference on Control Architectures of Robots, April 2009.
8. D. Kurabayashi, J. Ota, T. Arai and E. Yoshida. "Cooperative Sweeping by Multiple Mobile Robots". In Proceeding of the IEEE International Conference on Robotics and Automation, pages 1744-1749, 1996.
9. S. Tisue and U. Wilensky. "Netlogo: Design and implementation of a multi-agent modeling Environment". In Proceedings of the Agent 2004 Conference, 2004.
10. W. Sheng, Q. Yang, J. Tan, and N. Xi. "Distributed multi-robot coordination in area exploration". Robotics and Autonomous Systems, pages 945–955, 2006.
11. E. Gonzalez, E. Gerlein. "A Multi-Robot Coverage Algorithm". International Joint Conference on Web Intelligence and Intelligent Agent Technology, 2009.
12. W. Burgard, M. Moors, C. Stachniss and F. Scheider. "Collaborative Multi-Robot Exploration". In Proc. of the IEEE Transactions on Robotics, Vol. 21, No. 3, pg 376-378, 2005.
13. B. Yamauchi, A. Schultz and W. Adams. "Mobile Robot Exploration and Map-Building with Continuous Localization". In Proceeding of the IEEE International Conference on Robotics and Automation, pages 217-229, May 1998.
14. D. Maio and S. Rizzi. "Unsupervised Multi-Agent Exploration of Structured Environments". In Proceedings of the 1st International Conference on Multi-Agent Systems, pages 269-275, 1995.

15. B. Yamauchi, P. Langley, A. C. Schultz, J. Grefenstette and W. Adams. "Magellan: An Integrated Adaptive Architecture for Mobile Robots". Tech Report 98-2, Institute for the Study of Learning and Expertise, May 1998.

16. E. Rimon and D. Koditschek. "Exact Robot Navigation Using Artificial Potential Functions". In Proceeding of the IEEE International Conference on Robotics and Automation, pages 501-518, October 1992.

17. F. H. P. Fitzek and M. D. Katz. "Introduction to NetLogo: A Powerful Programming Tool for Modeling Cooperative Interactions in Wireless Networks". Cognitive Wireless Networks, pages 579–602, 2007.

18. A. Zelinsky. "A Mobile Robot Exploration Algorithm". In Proceeding of the IEEE International Conference on Robotics and Automation, pages 707-717, 1992.

19. K. Singh and K. Fujimura. "Map Making by Cooperating Mobile Robots". In Proceeding of the IEEE International Conference on Robotics and Automation, pages 254-259, 1993.

20. Y. Gabriely and E. Rimon. "Spiral-STC: An Online Coverage Algorithm of grid Environments by a Mobile Robot". In Proceeding of the IEEE International Conference on Robotics and Automation, pages 954-960, May 2002.

21. I. Rekleitis, G. Dudek, E. Milios. "Accurate Mapping of an Unknown World and Online Landmark Positioning". In Proceeding of Vision Interface, pages 455-461, 1997.

22. J. Vazquez and C. Malcolm. "Distributed Multirobot Exploration Maintaining a Mobile Network". In Proceeding of the 2nd International IEEE Conference on Intelligent Systems, pages 113-118, 2004.

23. R. Simmons, D. Apfelbaum, W. Burgard, D. Fox, M. Moors, S. Thrun and H. Younes. "Coordination for Multi-Robot Exploration and Mapping". In Proceeding of the National Conference on Artificial Intelligence AAAI, pages 851-858, 2000.

24. J. Ferber. "Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence". Ed Addison-Weley, page 84, 1999.

25. D. Kurabayashi, S. Koga, T. Arai, J. Ota, H. Asama and I. Endo. "Local Path Replanning for Unforeseen Obstacle Avoidance by an Autonomous Sweeping Robot". In Proceeding of the IEEE International Conference on Robotics and Automation, 1998.

26. U. Wilensky. http://ccl.northwestern.edu/netlogo. Center for Connected Learning and Computer-Based Modeling, 1999. Last viewed 29 march 2010.