

# Implementation of Cognitive Mapping Algorithm for Mobile Robot Navigation System

Amelia Ritahani Ismail, Recky Desia, Muhammad Fuad Riza Zuhri and Raja Muhamad Daniel  
Department of Computer Science  
Kulliyah of Information Communication Technology, IIUM  
P.O Box 10, 50728 Kuala Lumpur, Malaysia.  
amelia@iium.edu.my, recky.qie@gmail.com, fuad.zuhri@gmail.com, rajamhddaniel@gmail.com

**Abstract**—Mobile robot navigation has become as a challenging problem over the last few decades. Despite the significant advances in this field, mobile robot researchers have yet to reach a comfortable level of satisfaction and yet still finding an improved algorithm for robot's navigation. In this paper, we proposed investigate the *cognitive mapping algorithm* that is been implemented to the mobile robot. This mobile robot will be given some inputs which are horizontal and vertical value and start and goal point and a task which is to move from start point to goal point. Some obstacles are also located in the arena and the mobile robot must be able to avoid any given obstacles to reach the goal point. The performance of the cognitive mapping algorithm is compared with an edge follower navigation algorithm, which is an algorithm for a mobile robot to navigate by following the edges or the wall. Based on the experiments that have been performed the *cognitive mapping algorithm* can be an alternative way for the robot's navigation systems.

**Keywords**-Mobile Robot; Cognitive Mapping Algorithm; Robot Navigation System;

## I. INTRODUCTION

Mobile robot is a re-programmable, multi-functional product designed to move materials, parts, tools or specialized devices through variable programmed motions for the performance of variety of tasks [6], [10], [12]. They can be divided into types by their application and their locomotion and kinematics. For example, types of robot according to their application can be grouped as industrial robot, domestic or household robot, medical robot, service robot, military robot, entertainment robot and space robot. Another example, types of robot according to their locomotion and kinematics which are stationary robot, mobile robot, legged robot, swimming robot and flying robot. Nowadays, mobile robot has become popular in military, as they can be used as bomb detector robot, in house, mobile robot can be used as vacuuming or gardening and in industry, mobile robot can be used to help people for factory purpose.

In this paper, we proposed a cognitive mapping algorithm for mobile robot for a semi-dynamic environment. For achieving the goal, the mobile robot must be able to construct a map, to localize itself in it and to move from the start point to the goal point and avoid the obstacles in

the environment. The map will be constructed is based on the input of the user sent via Bluetooth module of android device. This map will serve as a cognitive map [3], [9] to the mobile robot to assist its navigation. The algorithm also include the obstacle avoidance algorithm to avoid any shape of obstacle that exist in the environment.

For comparison the edge (wall) follower algorithm [7] are also been implemented to be compared with the cognitive mapping algorithm. The edge follower algorithm is inspired by the behavior of insect where they are always trying to find wall, tree or anything that can be the edge of its environment. For example, insect perches on the wall or tree, or they also perch on leaf. Imitating how the insect traveled from a point to another point has been an inspiration for the edge follower algorithm. Just imagine on how insects like the cockroach spend the vast majority of their time following along the edges of rooms, shelves and such. They will usually only dare to leave the safety of the edges to venture out in to the open for food or if it is dark. The reason for this is that the cockroach is usually safer in the edges. The function of the edge followers is to orient the mobile so that it travels parallel to an edge, explore the environment and learn about the shapes of the area.

The rest of this paper is organised as follows: Section II describes the literature review, Section III explains on the robot's design and the description of the algorithm, Section 4 shows our experimental setup and finally the discussion on the results are presented in Section V.

## II. LITERATURE REVIEW

Mobile robot navigation is an approach of how a robot can reach the desired destination from an initialised starting point and avoiding the obstacle at the same time [2]. It can be seen with a sort of mobile robot algorithm where the implementation of robot movement has a fundamental problem on how to avoid the obstacle while the navigation algorithm can adapt on the current position of a robot. There are numbers of navigation algorithms proposed by the mobile robots' researchers. For example,[4] proposed a segment based mapping algorithm is along with an Extended Kalman filter driven by measurement taken by ultrasonic

sensors on the robot. The performance of proposed algorithm has been evaluated in both numerical and experimental tests which obtains good mapping and localization. However, the algorithm execution in the simulation shows the time is low enough to avoid the obstacles in the real time. [5] in his paper combines RaoBlackwellized particle filtering and scan matching in robot's navigation. Based on [5], the scan matching is used to minimize the odometric errors during mapping. The residual error of scan matching process is used for re-sampling steps. The required sample is reduced and at the same time it reduces the particle depletion problem. Other than these techniques, a real time algorithm for acquiring compact 3D maps of indoor environment by using range and imaging sensor have been introduced by [14]. Maps obtained by its algorithm consist of compact sets or textured polygons that can be visualized interactively. The result shows compact and accurate maps can be acquired in real time and in a fully automated fashion. Furthermore [13] presented a new online method for robust mapping and localization in indoor environments. The approach combines ideas from incremental mapping (maximum likelihood, incremental map construction) with ideas of more powerful, non-incremental approaches (posterior estimation, backwards correction). The result is a fast and robust algorithm for real-time mapping of indoor environments, which extends to multi-robot mapping and mapping in 3D. A fast algorithm was employed to generate compact 3D models of indoor environments.[1] in his paper implement a localization and navigation techniques to perform general task in an indoor office-like environment. The robot uses the camera to recognize the position based on the light fixtures as natural landmarks and the ultrasound sensor to avoid the obstacle. For the path planning, the robot uses the Wavefront algorithm. Finally, this system maintains a correct estimate of the robot's pose in real time, being able to correct the error accumulation of intrinsic sensors and estimate its own pose if this information is not provided. For indoor environment, [11] proposed a robot navigation system using a Partially Observable Markov Decision Process (POMDP) based on WiFi signal strength and ultrasound observations. Some comparison of applying the SLAM-EM algorithm using Ultrasound observation, WiFi observation and WiFi + Ultrasound observations. For Ultrasound observation experiment, the algorithm converges with an 85% of true locations but is not able to recover from lost of state. For the WiFi observation experiment (WSLAM), the algorithm converges with a 100% of true locations. For the algorithm with WiFi and Ultrasound observation, the result shows the best performance compared with the other two experiments. Hence, from this experiment, the WiFi and Ultrasound environment map with minimum effort is obtained.

### III. IMPLEMENTATION

This section begins with the description of the robot design followed by the algorithm flow and the description of the cognitive mapping algorithm.

#### A. Robot Design

There are two parts of robot body which are the upper part and the lower part. In the upper part, the components are breadboard, Arduino Mega 2560 board, Arduino sensor shield and Arduino Bluetooth module as shown in figure 1. Meanwhile In the lower part, the components are ultrasonic sensor, driver stepper motor, stepper motor, battery box and wheel as shown in figure 2.

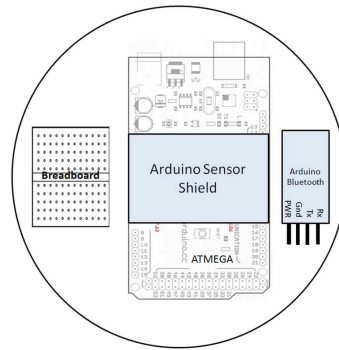


Figure 1. The upper robot body

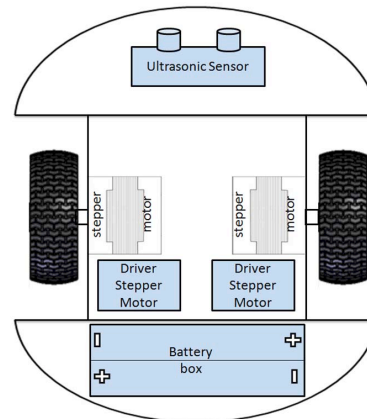


Figure 2. The lower robot body

#### B. Algorithm Flow

Generally, the scenario are divided into four cases which are right up, left up, right down and left down. All these scenarios are implemented on the cognitive mapping algorithm that will be described later in this section and considered to reflect the other scenario. For example, right up might reflect up right, left down might reflect down left, left up

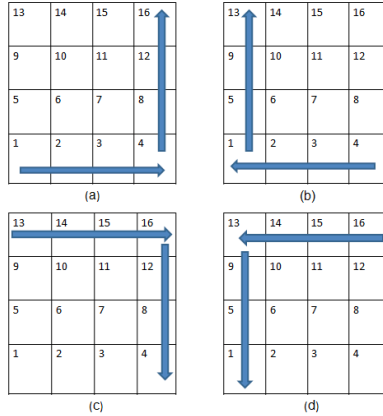


Figure 3. Four scenario of mobile robot: (a) Right Up Case, (b) Left Up Case, (c) Right Down Case and (d) Left Down Case

might reflect up left and right down might reflect down right. Hence, to make it as simple as it is, only these four scenarios would be considered. The figure 3 shows the four scenarios.

Fig. 3.(a) shows a case where the start point(1) is less than goal point(16) and the goal point coordinate on the horizontal view(4) is greater than start point coordinate on the horizontal view (1).

Fig. 3.(b) shows a case where the start point(4) is less than goal point(13) and the goal point coordinate on the horizontal view(1) is less than start point coordinate on the horizontal view(4).

Fig. 3.(c) shows a scenario where the start point(13) is greater than goal point(4) and the goal point coordinate on the horizontal view(16) is greater than start point coordinate on the horizontal view(13).

Fig. 3.(d) shows a scenario where the start point(16) is greater than goal point(13) and the goal point coordinate on the horizontal view(13) is less than start point coordinate on the horizontal view(16).

### C. The Cognitive Mapping Algorithm

The cognitive algorithm allow the robot to move from th start point to goal point based on the calculation executed in the beginning of the algorithm as explained in sec III-B. Basically, the calculation uses four inputs mentioned before which are start and goal point as well as vertical and horizontal value. Furthermore, the robot will determine the first step should be taken to start the work. The first step is taken based on the starting position of the robot and the final destination. The goal coordinate point is an axis that robot should move to.

#### Algorithm for Cognitive Mapping

```
StartNew := start point
GoalCoor := goal point % horizontal value
SPHor := start point % horizontal value
```

```
if GoalCoor == 0 then
GoalCoor := GoalCoor + horizontal value
else if SPHor == 0 then
SPHor := SPHor + horizontal value
endif
```

```
if start point % horizontal value == 0 then
GPHor := (GoalCoor + (horizontal value *
(start point / horizontal value)))
- horizontal value
else
GPHor := GoalCoor + (horizontal value *
(start point / horizontal value))
endif
```

From the above algorithm, it can be seen that in the beginning process of the algorithm, goal and start point modulus by horizontal point. The result of this process is very important since it takes a big role to the robot during the movement. The result will be a pivot point of the robot which will be explained later.

Besides that, the robot can get the current position point (*StartNew*), goal point coordinate (*GPHor*) and start point coordinate (*SPHor*) which will be used during the robot's work as well. These algorithms are used by robot to move toward another side(place) horizontally. On this function, to do path planning method, mostly current position (*Start-New*) compare to goal point coordinate by horizontal view (*GPHor*). The value of *UpVer* is also checked to know the situation around the robot. It shows that if *UpVer* is equal to 1 means there is an obstacle on the north side of the environment and 0 means there is no obstacle. It is also applicable to *DownVer* (south), *RightHor* (east) and *LeftHor* (west).

Figure 4 depicts the process based on the described algorithm. In this scenario, the robot will start from point 1 and stop at point 18. The third column is colored with yellow. The color yellow is a pivot for robot on this scenario. It means that when the robot start from point 1 and the destination at point 18, the robot will recognize that on the first row, point 3 is a pivot for robot to go to point 18. If the robot go up one step (on the second row), the robot will know that point 8 is a pivot that robot should direct to. On the third row, the robot will know point 13 is a pivot too which is the same as point 8 and 3. Here, the robot will assume that point 3, 8 and 13 are the points that robot should go first when the robot on the first, second and third row respectively and those points are called goal point coordinate (*GPHor*).

After the robot finds the pivot on each row, the robot will start to move toward the goal point. If there is an obstacle in front of it, the robot will avoid it and the avoidance step should direct the robot to move forward to the upper row.

While the robot on the next row, the first step robot should do is to find its pivot on that row. If the robot has found it, the robot has to continue the same procedure as before.

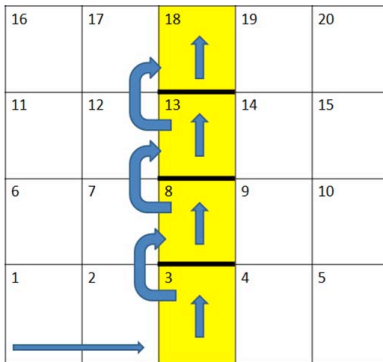


Figure 4. General concept of robot mapping

After the previous algorithm is executed, the next four algorithms will be implemented are these below algorithms.

#### Algorithm for Robot Path Planning

```
if start point != goal point then
direct the robot to goal point
endif
```

#### Algorithm for Obstacle Avoidance

```
if obstacle is detected then
change direction
else
continue the movement
endif
```

#### Algorithm for Robot Localization

```
if robot go forward then
update current position
endif
```

#### Algorithm for Robot Movement

```
if three previous algorithms
have been done then
move the robot
else
waiting for unexecuted algorithm
endif
```

### IV. EXPERIMENTAL SETUP

A grid of 4 cells x 8 cells are used to do the experiment and each cell has a measurement of 30 cm x 30 cm. The length and height of the obstacle are around 30 cm and 15 cm respectively. The horizontal and vertical point is provided to be processed by the robot to construct its cognitive map.

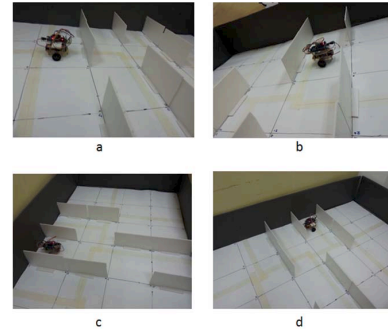


Figure 5. Robot mapping scenario: (a) robot mapping at start point, (b) robot mapping avoids the small obstacle, (c) robot mapping avoids the big obstacle and (d) robot mapping arrives at goal point

Start point and goal point node information is also given by user via Bluetooth communication. The inputs to the central station include:

- 1) Horizontal value
- 2) Vertical value
- 3) Start point
- 4) Goal point

Figure 5 shows the experimental arena for the implementation of cognitive mapping algorithm for mobile robot. The environment is still categorized as a known semi-dynamic environment. In figure 5(a), the robot is starting its task. In figure 5(b), the robot is avoiding the obstacle in front of it which is categorized as a small obstacle. In figure 5(c), the robot is avoiding obstacle placed in front, right and left of it which is categorized as a big obstacle. Finally, in figure 5(d), the robot arrives at the goal point and stop its work.

### V. RESULT AND ANALYSIS

The cognitive mapping algorithm is compared with the edge (wall) follower algorithm, where the robot in the environment navigates from the start point to the goal point by moving near the edges or the wall in its environment. For comparison, the robot's start point is 1 and the goal point is 14. The length of time taken is recorded for each algorithm. The result can be seen in figure 6 that shows the path that is taken by each algorithm and figure 7 that depicts the comparison of each algorithm based on time takes by the robot to navigate to the goal point.

### VI. CONCLUSION AND FUTURE ENHANCEMENT

It can be concluded that the cognitive mapping algorithm is more efficient and effective in term of time and distance than the edge follower algorithm based on the experiments that we have conducted.

For the next development of this mobile robot, some improvement should be added regarding to the obstacle avoidance so eventually the robot will be able to work in any given environment and avoid any given obstacle. The

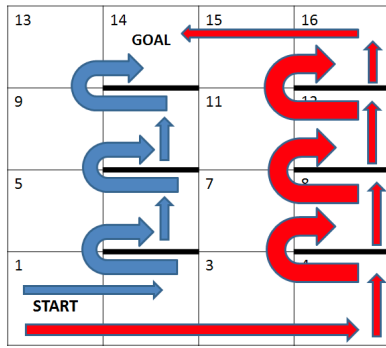


Figure 6. The comparison of path taken for both algorithms

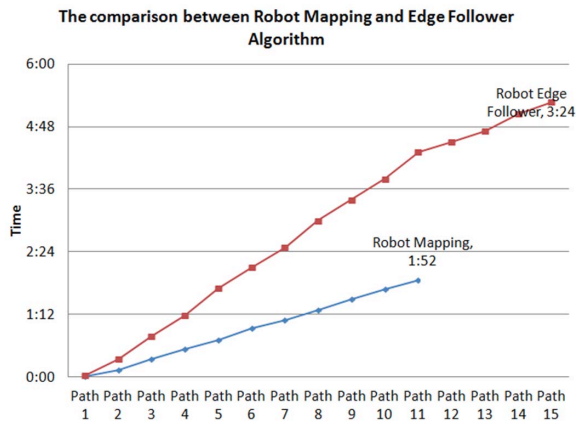


Figure 7. The comparison of time consumed and path taken for both algorithms

robot should be able to find the shortest path in maze and unknown environment.

## VII. ACKNOWLEDGEMENT

This research is funded by the Ministry of Higher Education under the Exploratory Research Grant Scheme (ERGS): ERGS 13- 017-0050.

## REFERENCES

- [1] Alves, P., Costelha, H. and Neves, C.: Localization and Navigation of a Mobile Robot in an Office-like Environment. *13th International Conference on Autonomous Robot Systems (Robotica)*. pg. 1-6. (2013)
- [2] Batalin, M. and Gaurav, S.: Mobile Robot Navigation Using Sensor Network. *IEEE International Conference on Robotics and Automation*, pp. 636-42. (2004)
- [3] Cognitive map. (2014, September 7). In Wikipedia, The Free Encyclopedia. [Online]. Available: [http://en.wikipedia.org/wiki/Cognitive\\_map](http://en.wikipedia.org/wiki/Cognitive_map)
- [4] Dalfonso, L., et al.: A SLAM Algorithm for Indoor Mobile Robot Localization Using An Extended Kalman Filter and A Segment based Environment Mapping. *16th International Conference on Advanced Robotics (ICAR)*, pg. 1-6. (2013)

- [5] Hahnel, D., et al.: An Efficient FastSLAM Algorithm for Generating Maps of Large-Scale Cyclic Environments from Raw Laser Range Measurement. *International Conference on Intelligent Robots and Systems*. pg. 206-211. (2003)
- [6] Madsen, D.: Manufacturing Materials and Process. *Engineering Drawing and Design*. Retrieved from <http://books.google.com.my/>. pg. 163. (2011)
- [7] Maze solving algorithm. (2014, April 16). In Wikipedia, The Free Encyclopedia. [Online]. Available: [http://en.wikipedia.org/wiki/Maze\\_solving\\_algorithm#Wall\\_follower](http://en.wikipedia.org/wiki/Maze_solving_algorithm#Wall_follower)
- [8] Mobile robot. (2014, September 16). In Wikipedia, The Free Encyclopedia. [Online]. Available: [http://en.wikipedia.org/wiki/Mobile\\_robot](http://en.wikipedia.org/wiki/Mobile_robot)
- [9] Montello, D. R.: Cognitive map-design research in the twentieth century: Theoretical and empirical approaches. *Cartography and Geographic Information Science*. pg. 283-304. (2002)
- [10] Nof, S. Y.: Cad and Graphic Simulators/Emulators of Robotic Systems. *Handbook of Industrial Robotics*. [Online]. Available: <http://books.google.com.my/>. pg. 755. (1999)
- [11] Ocana, M., et al.: Indoor robot navigation using a POMDP based on WiFi and ultrasound observations. *IEEE/RSJ International Conference on Intelligent Robots and Systems*. pg. 2592 - 2597. (2005)
- [12] Robotics. Robotics - Encyclopedia - Business Terms—Inc.com. (September 23, 2014). [Online]. Available: <http://www.inc.com/encyclopedia/robotics.html>
- [13] Thrun, S., Burgard, W. and Fox, D.: A Real-Time Algorithm for Mobile Robot Mapping with Application to Multi-Robot and 3D Mapping. *IEEE International Conference on Robotics and Automation*. pg. 321-328. (2000)
- [14] Thrun, S., et al.: A real-time expectation-maximization algorithm for acquiring multiplanar maps of indoor environments with mobile robots. *IEEE Transactions on Robotics and Automation*. pg. 433-443. (2004)