I J C T A, 9(37) 2016, pp. 453-459

Path Planning Algorithm for a Mobile Robot – A Comparative Study in Different Environment

Debadatta Satapathy[*] and G. Joselin Retna Kumar[**]

Abstract: Industrial and technical applications of mobile robots are continuously attaining a great importance under the consideration of reliability, accessibility and cost factor. Mobile robots are already widely used for the purpose of surveillance, inspection and transportation tasks-a further emerging market with enormous potential and tremendous requirement. So it is obvious that a range of fundamental competences should be available for a mobile robot to operate it safely in unstructured and uncertain environment.

In this sense, the overview of the paper discusses about the path planning algorithm of the mobile robot in an amorphous area. For this designing, A* (A star) algorithm is studied in occupancy grid map (OGM) and voronoi diagram. Various measurement factors and constraints are taken into consideration and a comparative analysis is proposed here. After that the shortest path mechanism with optimized algorithm is derived as the primary significance.

The above designing idea is developed in LabVIEW environment and its robotics simulator. The main objective of this implementation aims towards the flexibility and tractability of LabVIEW in the field of Robotics and Artificial Intelligence.

Index Terms: A* (A Star), voronoi, OGM, LabVIEW.

1. INTRODUCTION

In these past few decades the importance of mobile robot in the field of industrial, technical and military application has a great impact. For the better performance of automobile robot the most important factor is the path planning. In case of path planning we have to decide what is the source or destination, which path is the optimal means to find feasible solutions as fast as possible or to find out which path is the shortest one between the source and destination. Among all the factors to design a better path planning the most important factor is obstacle detection and avoidance. Thus we can call the entire path planning as the collision avoidance problem [1]. The necessity of a collision free path is that it will minimize the costs such as time, energy and distance [2]. For the accuracy of every path planning the essential key is the environment mapping and the localization [3]. The algorithms are used so that the automobile robot can able to detect the U-shape obstacles [4].

There are basically two types of path planning. They are Global path planning and Local path planning. In case of global path planning the environment and situations are given. We can say that the environment is static in case of global path planning. The path for the automobile robot has been decided before the programme runs. The global path planning is also known as static path planning. But if the local path planning is adjusted synchronously it is known as dynamic path planning [1].

There are different path planning methods for automobile robot in both global path planning and the local path planning. They are grid based, road map, cell decomposition and artificial potential field [5]. Among these path planning, grid based path planning is the common method for the description of the environment. The grid based path planning also known as cell decomposition type path planning [6].

[*] Dept. of Electronics and Control, SRM University, Chennai, India. *Email: satapathy.debadatta1992@gmail.com*
[**] Asst. Prof., Dept. of Electronics and Instrumentation, SRM University, Chennai, India. *Email: Joselin_su@yahoo.com*

For a better optimal path planning or reduce path cost shortest finding path algorithms are used; such as A* (A star) algorithm [7, 3], Dijkstra algorithm, D* (D star algorithm) [8], Bezier method [9].

Among all the path planning for automobile robot Occupancy Grid (OG) and Voronoi mapping are used mostly. The occupancy grid algorithm basically uses histogram based approach for the detection of boundary. This is a location based representation by considering the obstacle occupied parts [10]. The occupancy grid calculation based on the Basyian approach [7, 11]. Voronoi facilities are efficient methods for piece wise approximation of the environment. The approximation used in voronoi is Herustic approximation [2]. For the detection of the objects for obstacle avoidance many sensors can be used. But mostly ultrasonic sensor, laser finder (LIDAR), stereo vision sensor are used. [11]

A* (A star) algorithm is widely used to find the optimal and the shortest path for the path planning of the mobile robot. Dijkstra can also be used to calculate the shortest and the optimal path but the problem is that Dijkstra iterates through all the nodes and both the efficiency and searching and calculation is poor compare to A* algorithm [12].

The rest part of the paper can be formulated as follows. In the section II types of grid method both the occupancy grid algorithm and the voronoi diagram mapping and the A* algorithm for the shortest and optimal path for the path planning is described. In the section III two of the grid methods designed in LabVIEW platform for the shortest path. In the section IV conclusion has been presented.

2. GRID MAPPING AND A* ALGORITHM

The most common method to describe the environment is the grid method. From many grid methods the occupancy grid and the voronoi diagram mapping are mostly used. The idea of the grid here is to ruled out the discrete environment into one or two dimension and to acquire the environment system modeling of the grid as limited area [6]. The grid method is the most common method because the overall cost can be minimized by partitioning into balanced subsets [14].

A. Occupancy Grid Mapping

Occupancy grid is first introduced by Elfes and Moravec [13]. Occupancy grid is nothing but a location based representation. Occupancy grid looks like a matrix of cells and depending on the sensor reading it calculates the probability of each cell. The basic difference between the tracking and the occupancy grid is that; occupancy grid not only contains the presence of the objects but also the absence of objects in grid cells. Based on the sensor reading the occupancy grid mapping designed such that the entire grid is divided into evenly space grid of cells and the probability of each cell is estimated according to the occupancy of obstacle [10]. In the mobile robotics a system is allowed by occupancy grid algorithm an internal map of the environment can be created so that the noise can be accounted the noise from the sensor reading [7]. In another point of view we can say that Occupancy grid subdivided into finite cells and every cell present inside the occupancy grid algorithm contains two information; the first one is the location of the cell in the environment and the second one is its probability of distribution which reflects a belief about presence of the object at that location [11]. Some of the major disadvantages of occupancy grid algorithm is as follows. One is the computational intensive in occupancy grid algorithm and another is the trade-off between the size and the resolution of the grid [10]. An example of the occupancy grid using LabVIEW is given below in Figure 1. In this Figure 1 the black spaced are obstacles and the white one are obstacle free spaces. The two red symbols are start and the goal. By using A* algorithm we can find out the shortest path between the start and the goal.

B. Voronoi Diagram Mapping

Mehlhorn and Erwig proposed the graph voronoi diagram [14].The basic definition of a voronoi diagram is that it is an important data structure which is used to describe the space relationship and also is an appropriate way for extracting topology of space connected domain [6]. The classical voronoi diagram is nothing but a distance based decomposition of a metric space relative to a discrete set, the Voronoi sites [14]. Both the local path planning and the global path planning can be constructed using voronoi diagram mapping. Basically the environment in the voronoi diagram mapping is a skeleton based mapping.

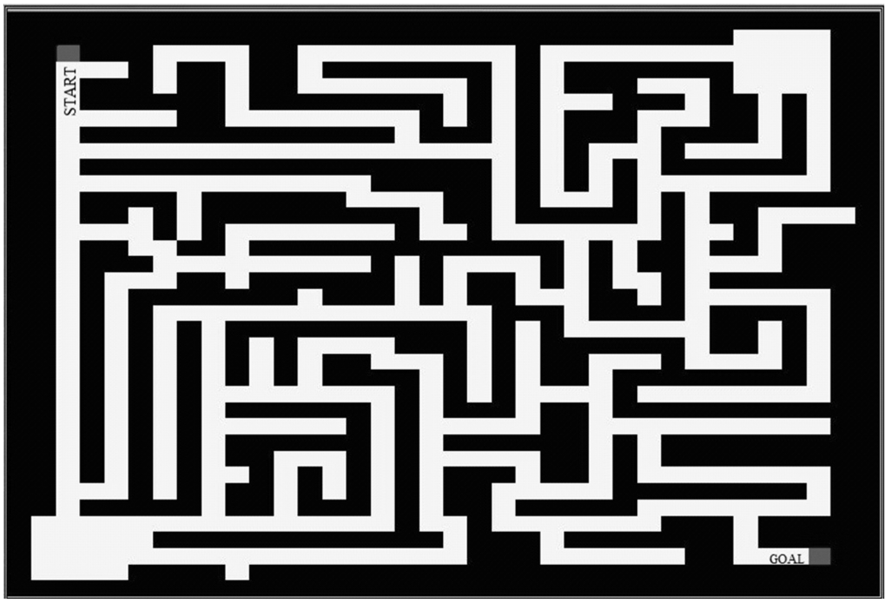

Figure 1: Example of occupancy grid algorithm in LabVIEW platform

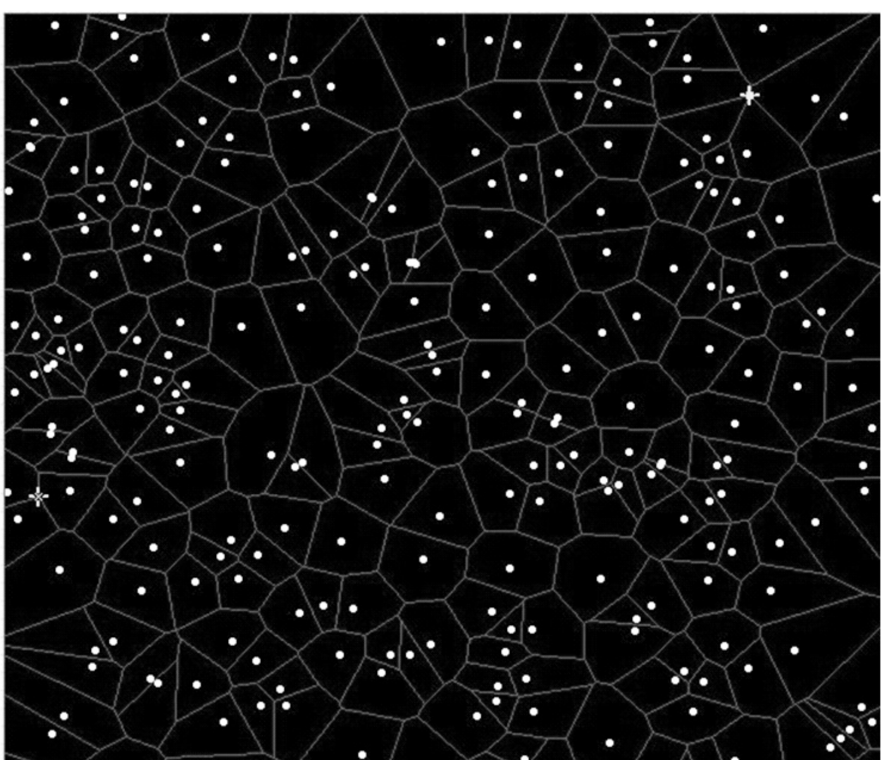

Figure 2: Example of vornoi diagram mapping in LabVIEW platform

An example of voronoi diagram mapping in LabVIEW platform is given in Figure 2. In this above figure all the white dots are the obstacles and the red lines are vertices. In the above figure all the obstacles are inside a vertex. The two yellow cross signals are start and goal. From these two points means from start to goal there are many paths. So to determine the optimal path and the shortest path we have to apply the A* algorithm in the voronoi diagram. In case of voronoi diagram every site objects are independent of each other [15] as shown in Figure 2.

C. A* (A star) algorithm [3][16]

The A* algorithm is a combination of Dijkstra and Herustic algorithm. It is one of the algorithms vastly used to find out the shortest distance between two nodes. The Herustic function is a method of calculating distance between the current expanding cell and the goal. The A* start with expanding from the start node and all neighbors are placed in a stack mode. The stack cell is ordered according to the low cost value of $f(x)$ and including the Heuristic functions $h(x)$. By finding the lowest cost cell it is extracted and expanded. The evaluation function used in this algorithm is given by $f(x) = g(x) + h(x)$.

x = (Xx, Yx) is recent expanding node or cell

$f(x)$ = Estimated minimum cost of the cells between all the paths and the start node.

$g(x)$ = Actual cost function from the start node.

$h(x)$ = Herustic estimation of minimum cost of a path form the current node.

3. LABVIEW SIMULATION AND RESULTS

A. Occupancy Grid Map design using A* algorithm

Occupancy grid map is developed here by using the Intensity graphs, which depends on the preset values to modify the graph into black and white regions. The intensity value for white is 1 and for black it is 100. The grid in the form of 1 and 100 are given in Figure 3. These black regions mention the occupancy of obstacle and white regions represent the absence of object. After the designing of grid map, main objective is to find out the optimal and shortest path between starting goal and final goal. For this purpose A* algorithm is taken into consideration. And the main advantage of this type of path planning approach is online decision making ability to find out the shortest path, if an object comes on the previously calculated optimal path. So, it is coming under the category of dynamic path planning system. This path planning system is designed and tested in LabVIEW platform by using Robotics tool box, which performs mainly three functions like starting position of robot, ending position of the robot and marking of obstacle in currently decided optimal path. The simulated system front panel is shown in Figure 4.

B. Voronoi Diagram Map design using A* algorithm

Voronoi diagram map design is developed by using voronoi map in LabVIEW. In this vornoi design in LabVIEW platform the obstacles are given as white dots and the vertices are in red colour. The yellow cross lines are the start and goal. The green line represents the optimal and the shortest path between the start and goal. To find the shortest path algorithm A* algorithm is used. Like the occupancy grid mapping this voronoi is also has the capability of decision making to find the shortest path.

This type of voronoi mapping comes under the dynamic path planning. There are 3 position starting node, end node and the shortest and no of obstacles to design the graph. The shortest and optimal path in LabVIEW platform in Figure 5.

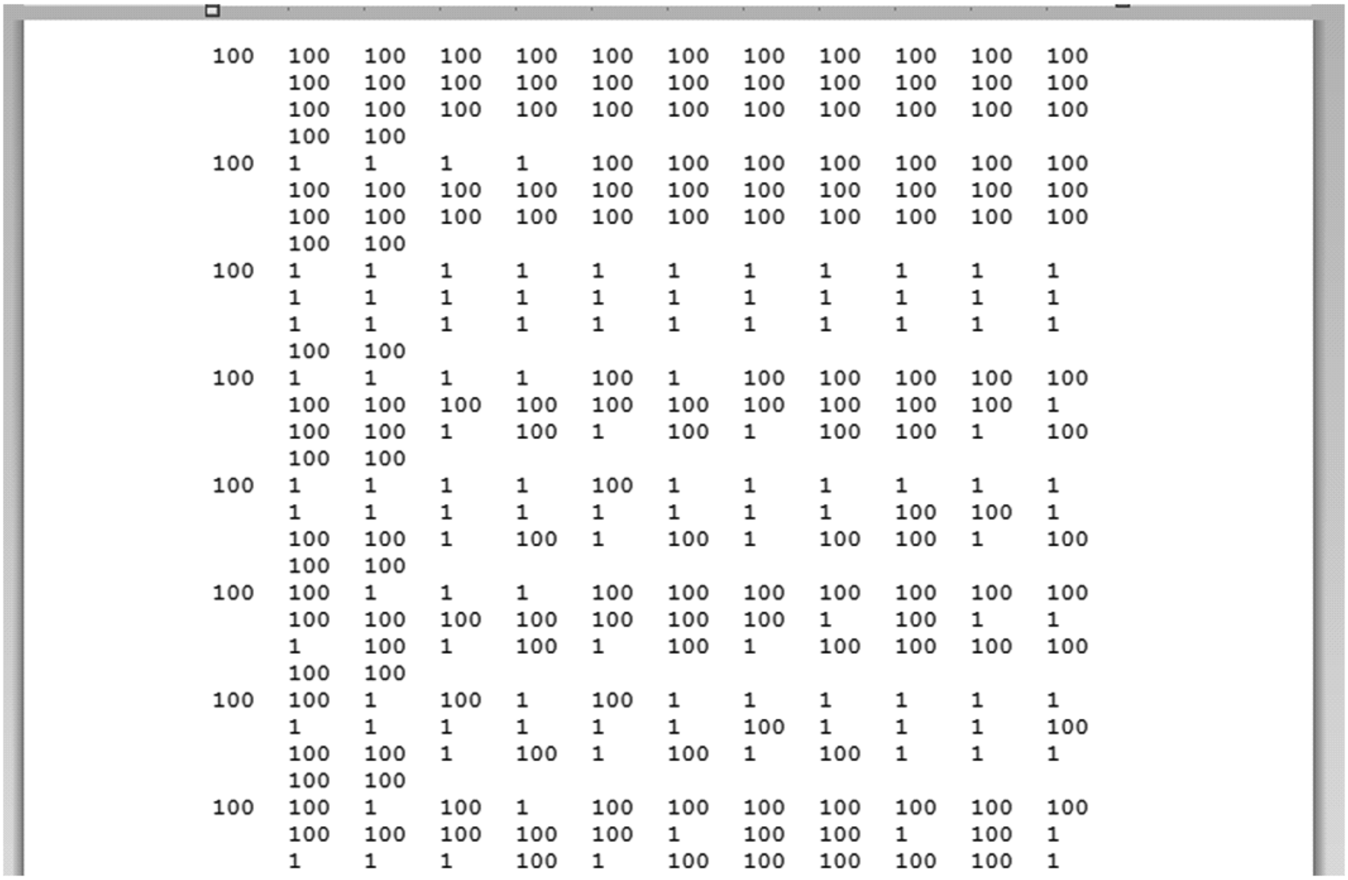

Figure 3: Grid formation in the form of 1 and 100

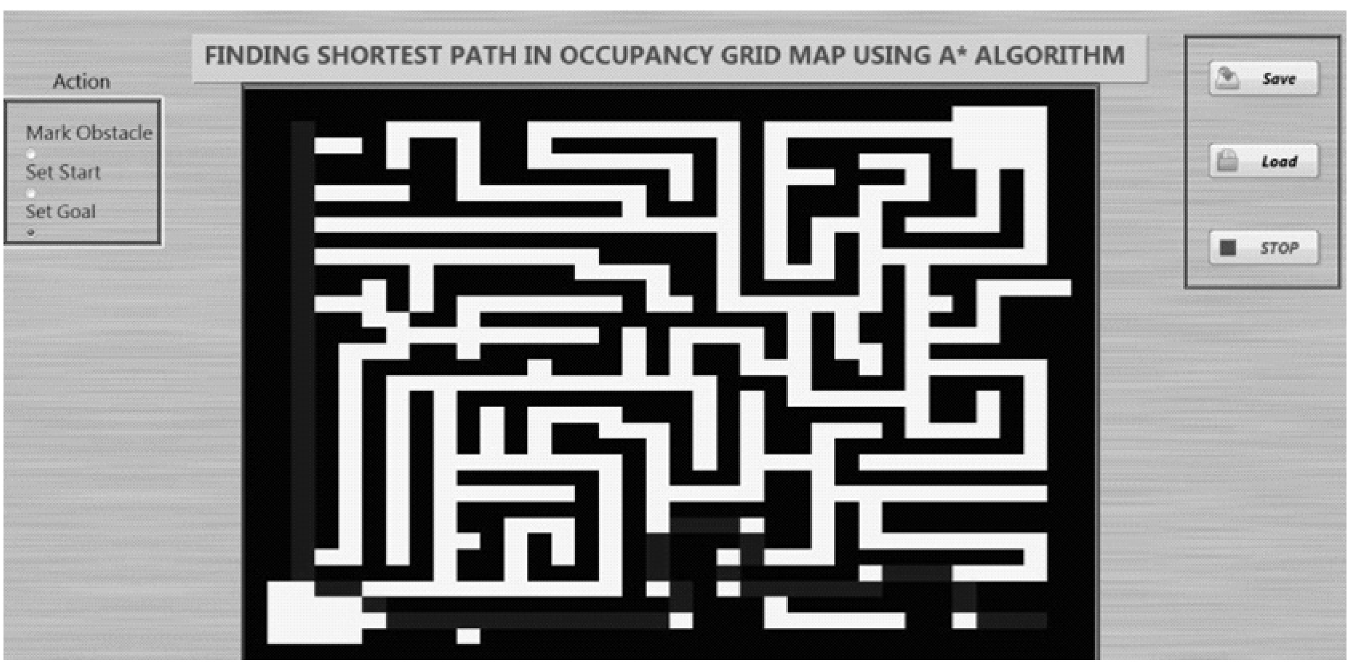

Figure 4: LabVIEW simulation for the Occupancy Grid Algorithm using A* algorithm

4. CONCLUSION AND FUTURE WORK

In the current paper two types of approach has been made to find out the shortest and the optimal path between the start node and the goal using LabVIEW platform. A comparative study has been implemented

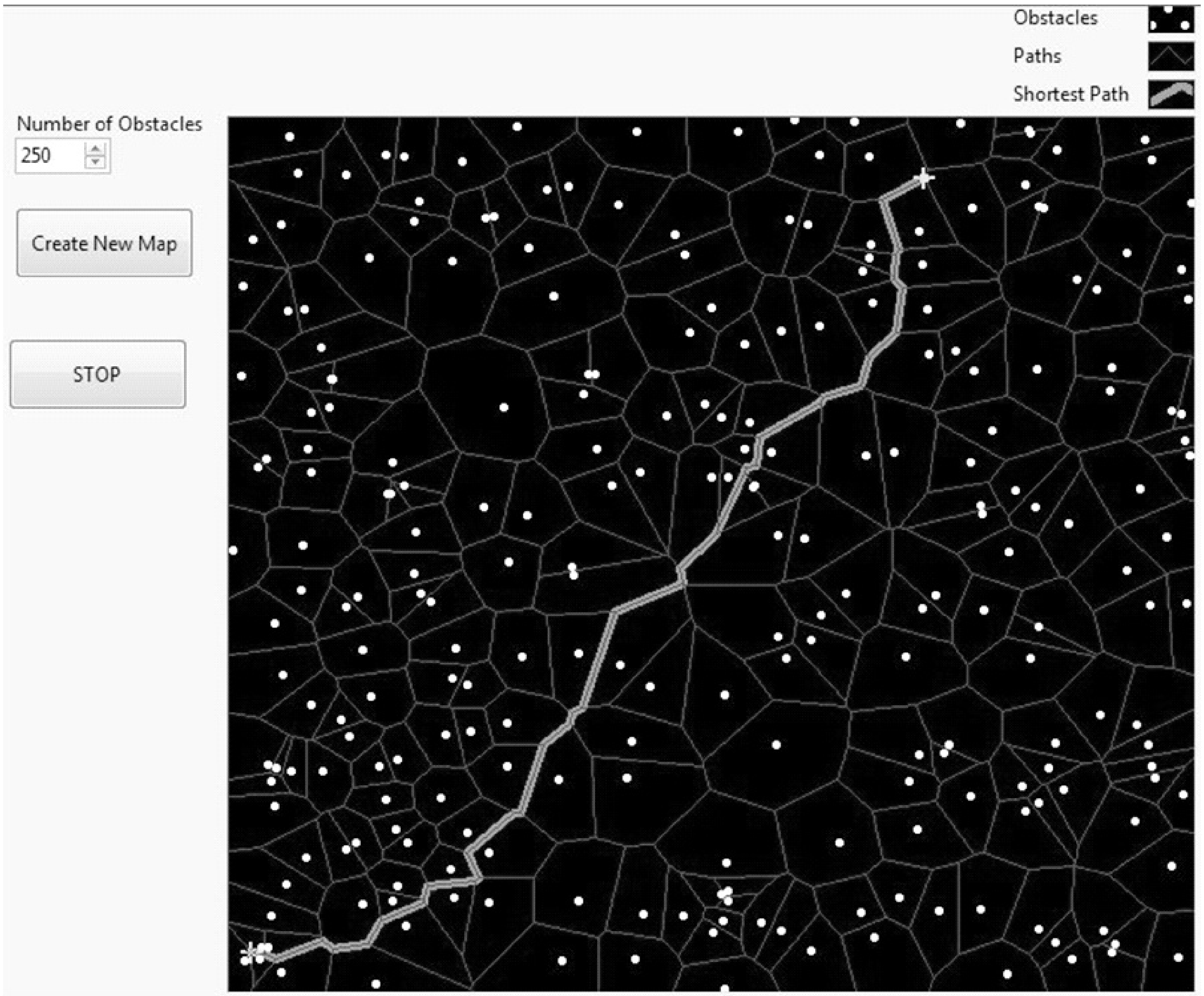

Figure 5: LabVIEW simulation for the Occupancy Grid Algorithm using A* algorithm

between the occupancy grid mapping and the voronoi diagram mapping. In both of the environment to find out the optimal and the shortest path between start and goal node A* (A star) algorithm has been approached. These two algorithms has been implemented and a successful simulation has been made by using LabVIEW by the help of robotic tool kits present in LabVIEW platform. By using robotics starter kit 1.0 or 2.0 or NI MY RIO tool these two types of algorithm can be implemented in the real world application by the help of ultrasonic sensor.

References

1. Faming Gong and Xin Wang "*Robot Path-planning based on Triangulation Tracing*" International Symposium on Intelligent Information Technology Application Workshops, IEEE, pp. 713-716, 21-22 Dec. 2008
2. Mr. Rabindra Kumar Shial and Mr. K Hemant Ku.Reddy "*Navigational Approach for Mobile Robots*", Emerging Trends and Applications in Computer Science (NCETACS), 2011 2nd National Conference on, IEEE, pp. 1-5, 4-5 March 2011.
3. Jitin Kumar Goyal and K.S. Anglia "*New Approach of Path Planning for Mobile Robots*", Advances in Computing, Communications and Informatics (ICACCI, 2014 International Conference on, IEEE, pp. 863-867, 24-27 Sept. 2014.
4. Joao Dinis and Margarida Mamede "*Updates on Voronoi Diagrams*", Voronoi Diagrams in Science and Engineering (ISVD), 2011 Eighth International Symposium, pp. 192-199, 28-30 June 2011.
5. Allan R. Willms and Simon X. Yang "*An Efficient Dynamic Real System for Real Time Robot Planning*", Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on Vol. 36, Issue: 4, pp. 755-766, Aug. 2006.

6. Qian Zhang, Ming Li, Xuesong Wang, "*Global Path Planning Method of Mobile Robot in Uncertain Enviroment*", Control and Decision Conference (CCDC), 2010 Chinese, pp. 4320-4324, 26-28 May 2010.

7. John Stamford, Arjab Singh Khuman, Jenny Carter and Samad Ahmadi, "*Path finding in partially explored games environments, The application of the A* Algorithm with Occupancy Grids in Unity 3D*," Computational Intelligence (UKCI), 2014 14th UK Workshop on , IEEE, pp. 1-6, 8-10 Sept. 2014.

8. Chen-Chien Hsu, Yuan-Jun Chen, Ming-Chih Lu and Shih-An Li "*Optimal Path Planning Incorporating Global and Local Search for Mobile Robots*" Consumer Electronics (GCCE), 2012 IEEE 1st Global Conference on, pp. 668-671, 2-5 Oct. 2012.

9. Yi-Ju Ho and Jing-Sin Liu, Member, "*Collison free Curvature-bounded Smooth Path Planning using Composite Bezier Curve based on Voronoi Diagram*," Computational Intelligence in Robotics and Automation (CIRA), 2009 IEEE International Symposium on , IEEE Transl. pp. 463-468 15-18 Dec. 2009

10. F. Homm, N. Kaempchen, J.Ota and D. Burschka "*Efficient occupancy grid computation on the GPU with Lidar and Rader for road boundary detection*", IEEE Intelligent vehicles symposium university of Callfornia, Vol. 2, pp. 1006-1013, 16-17 Dec 2010.

11. S.E. Hadji, T.H. Hing, M.S.M. Ali, M.A. Khattak, S. Kazi "*2D occupancy grid with inverse range sensor model*" IEEE 10^{th} Asian Control Conference, pp. 1-6, May 31-2015-june 3 2015.

12. Liping Cheng ,Chuanxi Liu and "*Improved Hierarchical A-star Algorithm for Optimal Parking Path Planning of the Large Parking Lot*" Information and Automation (ICIA), 2014 IEEE International Conference on, IEEE, pp. 695-698, 28-30 July 2014.

13. H. Moravec and A. Elfes, "*High resolution maps from wide anglesonar*," in Proc. IEEE Inter. Conf. on Robotics and Automation (ICRA), 1985.

14. Shinichi Honiden, Michael E. Houle, and Christian SommerR. Nicole, "*Balancing Graph Voronoi Diagram*" 2009 Sixth International Symposium on Voronoi Diagrams, pp. 183-191, 23-26 June 2009.

15. Chen, D.Z.,Ziyun Huang ,Yangwei Liu and Jinhui Xu "*On Clustering Induced Voronoi Diagrams*" Foundation of Computer Science (FOCS), 2013 IEEE 54^{th} Annual Symposium, pp. 390-399, 26-29 Oct. 2013.

16. Junfeng Yao, Chao Lin , Xiaobiao Xie , Andy JuAn Wang and Chih-Cheng Hung, "*Path Planning for Virtual Human Motion Using Improved A* Algorithm*", 2010 Seventh International Conference on Information Technology, IEEE, pp. 1154-1158, 12-14 April 2010.