

An Overview of Path Planning and Obstacle Avoidance Algorithms in Mobile Robots

¹Basavanna M, ²Dr. Shivakumar. M

¹Assistant Professor, Dept of TE, GSSSIETW,Mysuru ,

²Professor , Dept. of E&IE, GSSSIETW, Mysuru

Abstract:- Main goal of autonomous robot is to reach the destination by traversing through optimized path defined according to some criteria without any collision with the obstacle which may come in its path. Hence, Path planning and obstacle avoidance is the backbone of Autonomous Mobile Robot. Several algorithms have been proposed by various researchers for path planning and obstacle avoidance having several advantages and limitations. In this survey paper, we have mainly discussed about different algorithms for robot navigation with obstacle avoidance. We have also compared various algorithms and by indicating their characteristics; advantages and disadvantages along with simulation results.

Index Terms – Mobile robots, obstacle avoidance, path planning

I INTRODUCTION

A mobile robot is a vehicle which is capable of an autonomous motion. With the development of robotic technologies over the years, mobile robots have been widely employed in indoor environments such Cleaning large buildings ,Transportation in industry, Surveillance in large buildings and outdoor environments such as Agriculture ,Forest, Space, Underwater ,Military ,Fire fighting Sewage tubes and Mining.. The Autonomous mobile robots are being developed for numerous applications where long term capabilities would be beneficial. The main aim of mobile robotics is to create completely autonomous robots in the sense that they must be able to complete their tasks without human intervention. The Industrial and technical applications of mobile robots are continuously gaining importance in particular under considerations of reliability (uninterrupted and reliable execution of monotonous tasks such as surveillance), accessibility (inspection of sites that are inaccessible to humans, e.g. tight spaces, hazardous environments or remote sites) or cost (transportation systems based on autonomous mobile robots can be cheaper than standard track-bound systems)..

II. OBSTACLE AVOIDANCE METHODS

An autonomous mobile system should be able to plan its route towards the destination, detect and avoid obstacle in the path to reach the destination with an acceptable

accuracy. In every mobile robotic system, the skill to avoid obstacles is crucial. The robot must be trusted to complete its tasks without being a hazard to itself or to others. The requirements for better obstacle avoidance algorithm are it should be fast, robust and not dependent on prior information about the environment.

The Path Planning approaches in mobile robot can be classified into traditional or conventional method and Soft Computing method. The traditional method does not enforce intelligence into the path planning and it includes Graph Searching Techniques, Artificial Potential Field, cell decomposition method, Vector Field method and Road Map method. The soft computing methods introduce intelligence into the path planning and that includes Genetic Algorithm [16], Ant colony Optimization, Swarm algorithm, neural networks, and Fuzzy logic [6]. These two approaches are briefly discussed in the following sections along with their merits and drawbacks :

1. The Bug Algorithms

The action of mobile robots in finding a collision free path or avoiding the obstacle can be derived from the intelligence of an animal or bird behaviour. In Bug algorithm, a sensor is used to detect an obstacle and to minimize the outer perimeter of an obstacle usage without the need of a map and response to the output based on a sensor data.

The Bug's algorithms has three assumptions about the mobile robot: i) the robot is a point, ii) it has a perfect localization, and iii) its sensors are precise. The bug algorithm is the simplest obstacle avoidance algorithm ever described [1]. According to it, when an obstacle is encountered, the robot fully circles the object in order to find the point with the shortest distance to the goal, then leaves the boundary of the obstacle from this point (see figure 1 & 2).

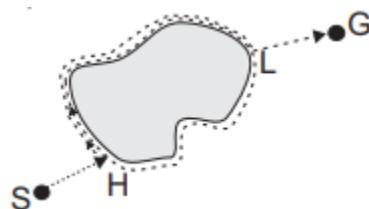


Figure1: Illustration of Bug1 algorithm.

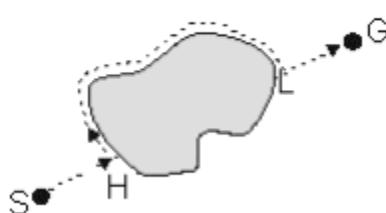


Figure 2: Bug2 algorithm

In Bug1 algorithm, the mobile robot moves towards the goal directly, unless it encounters an obstacle, in which case the robot explores the external lines of the obstacle until the motion to the goal is available again [22]. Whenever the mobile robot faces the unknown obstacle while travelling to the goal, it goes around the obstacle in clockwise sense (default), and then determines the leave point by calculating the distance between the current position and the goal position G during travelling around the object. The leave point is the closest point around the obstacle to the goal. Then, the robot determines the shortest path to this closest point in order to reach the leaving point, and changes or keeps its direction of the wall following according to the shortest path to return to the leave point. Next, the robot goes to the leave point and departs from the obstacle towards the goal position G along a new line. When it faces the second obstacle, the same procedure is applied. This method is inefficient but guarantees that the mobile robot is able to arrive any reachable goal point

The performance of Bug2 is better than the performance of Bug1. But in some cases, for instance, a maze searching; Bug2 loses its advantage upon Bug1. The mobile robot follows the slope as possible as it is able to be on it.

Advantages: Their simplicity is a major advantage of this algorithm.

Disadvantages: The bug-type algorithms have some significant shortcomings: they do not consider the actual kinematics of the robot which is important instead they consider only the most recent sensor readings, and the sensor noise will affects the overall performance of the robot, and becomes slow.

2. Probabilistic Road map (PRM)

PRM is a simplified sample-based algorithm which works by constructing probabilistic road map comprising networks of connected nodes in a given map based on free and occupied spaces to find an obstacle free path from start to end point [18], [33]. Once the roadmap is constructed, a path connecting source to destination is extracted using graph searching techniques. The Algorithm begins road map with a node at the initial configuration which is then expanded randomly by adding edges and nodes further [14], [34].

3. Artificial Potential Fields

The Potential field method was first proposed by Khatib [37]. This algorithm is based on the principle of Potential field force of attraction or repulsion in which robot and obstacle act as a positive charge where as goal act as a negative charge. Thus, obstacles repel from robot by generating repulsive force and goal attracts robot due to opposite charge results in attractive force. Final force on robot is the vector sum of all repulsive and attractive force.

Advantages of APF are

- For static obstacles with well-known atmosphere, potential is estimated offline such that the robot's velocity is within the energy field from start to destination point.
- Applicable for online or real-time environment as well with added obstacle avoidance component.

Disadvantages: Potential fields method suffers from Local minima problem and high complexity due to its bi-operational path model. Koren and Borenstein [17] identified some problems that are inherent to potential fields, i.e. they exist in all implementations of the method. The problems are:

i. Trap situations due to local minima: The local minima problem may occur when all the vector field from obstacles and the goal point cancel each other so the path never reaches the goal.

The local minima problem can be overcome by backtracking from original position or by doing some random movements from current location or by using any one of the Bug Algorithm [22] for obstacle avoidance. This local minimum problem can also be solved using some soft computing techniques. Hosseini et al [4] have worked on path planning for mobile robot using APF in which the workspace for mobile robot is divided into rectangular cells of grids where each cell is considered as an obstacle or non-obstacle. They have evaluated the potential function for each cell based on the distance estimated from the source, obstacles and destination by an autonomous mobile robot.

ii. No passage between closely spaced obstacles. If two obstacles are placed close to each other like a doorframe, the repulsive forces from each obstacle is combined into a single repulsive force that points away from the opening between the obstacles. The robot will then turn away from the opening even if the goal is on the other side.

iii. Oscillations in the presence of obstacles and in narrow passages: Borenstein and Koren[17] also showed that the robot sometimes would oscillate in the presence of obstacles and in narrow corridors. When the robot is travelling along the centreline between the two walls, the motion is stable. However, the robot would drift closer to one of the walls as it would be pushed back across the

centreline by the repulsive force from the closest wall. The robot would now experience a repulsive force from the other wall and be pushed back across the centreline. This behaviour would then continue and the robot would oscillate between the two walls. These problems can be overcome by soft computing techniques.

The APF algorithm is simulated using Matlab and results are shown below

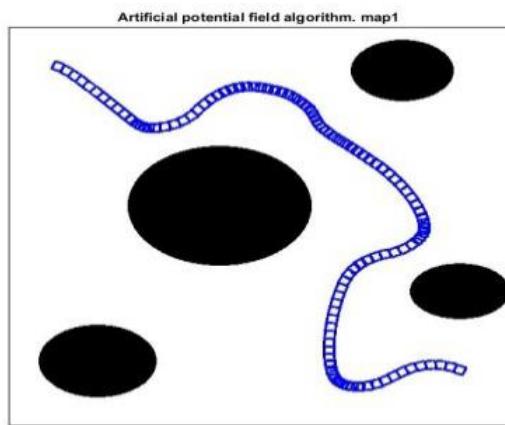


Fig 3: APF. Map1

Readings:

Processing time=9.182928e+001sec
Path Length=8.393897e+02mm

3 .RAPIDLY-EXPLORING RANDOM TREES (RRT)

RRT is a data structure and sampling scheme to quickly search high dimensional spaces that have both algebraic constraints and differential constraints [25]. In RRT algorithm, a source node or tree is created at the start point which expands or extends randomly further to branches of the tree within the workspace finally leading to optimal collision free path which is near to goal or destination node.

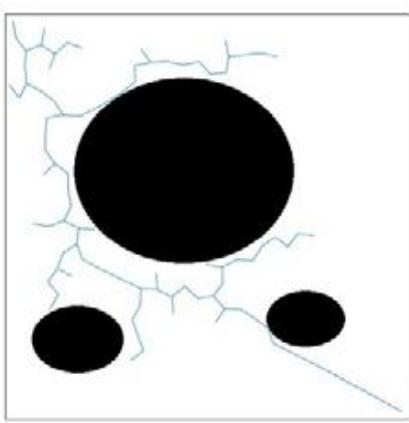


Fig.4: Map1 Roadmap using RRT

Readings:

Processing time=7.236812e+00sec

Path Length=8.725659e+02mm

4. Bidirectional RRT

BRRT is an extension of RRT algorithm where it explores optimal collision free path using search space method. Difference between RRT and BRRT is former approach RRT uses single trees to explore the path from source to destination point while BRRT uses two trees [7]. In BRRT, one tree starts from source location and extends towards destination locations while the second tree starts from destination location and extends towards source location. The point at which they meet is the optimal collision free path achieved [7]. The algorithm is simulated using MATLAB platform and result is shown in Fig4.

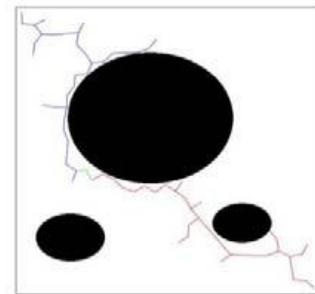


Fig.5: Map1 Roadmap using RRT

Readings:

Processing time=1.962829e+01sec
Path Length=1.1838272008e+03mm.

5. Vector Field Histogram (VFH)

Koren and Borenstein [6] proposed VFH method that deals with the shortcomings of potential fields.. The VFH method uses the histogram grid to represent the environment in which the robot travels. The contents of each cell in the histogram grid is treated as an obstacle vector. The direction of the vector is the direction from the current position of the robot to the obstacle. After that, the area surrounding the robot is divided into a number of equal sectors and in each sector, the obstacle vectors are summed together to give the polar obstacle density (POD).

Some severe problems with VFH method are **Dynamic constraints:** VFH method is augmented by taking the dynamics into consideration. The dynamic constraints imposed on the robot will block some of the sectors that were marked as free for travel .

Parameter settings: Another severe problem in VFH method is that the threshold that determines when a sector is **safe for travel** defines the behaviour of the robot. A too low threshold will make the robot unable to move in the presence of obstacles, whereas a too large threshold will make the robot bump into the obstacles.

Jianjun et al [23] proposed an improved VFH for path planning of real-time robot when the environment is unknown and dynamic. The ratio parameter for the area to be covered has been proposed by considering the size of the robot and obstacles. The authors also used Fuzzy Logic module to effectively avoid the obstacles in the dynamic environment The proposed technique rectifies the

accumulated error, need of large grid space for storage and also overcomes the local minima problem.

6. FUZZY APPROACH

Fuzzy logic deals with situations that are neither completely true nor completely false, it represents a partial solution when a perfect solution cannot be predicted and used to solve when pattern recognition problems arise in robotic tasks with more robustness. In this algorithm, whole logic is divided into simpler blocks composed of set of fuzzy logic rule statements intended to achieve desired objective. Decision of motion is prepared only on the basis of input parameters not on real-time situation. The advantages in Fuzzy logic is that it produces better result than a human can produce in a short period of time. It is well suited for implementing a solution in the complex autonomous mobile system, but it is difficult for simple control system.

7. GENETIC ALGORITHM

It was introduced by John Holland. In path planning and obstacle avoidance algorithm, genetic algorithm is used to move in a dynamic environment with predictable and unpredictable obstacles [16][20].

The concept is shown in Figure 3. Let a path be characterized by a fixed number of points in the robotic map. In order to make some path from this set of points, we start from the source and connect it to the first point by a straight line. The first point is connected to the second point by a straight line, and so on. At the end, the last point is connected to the goal. The objective function is the length of this path. A heavy penalty is added if any part of the path lies inside the obstacle, while the penalty is proportional to the length of the path inside the obstacle. The locations of each of these fixed number of points (both x and y axis positions) are the optimization variables. The variable bounds are such that the point lies inside the map (lower bound 1 and upper bound as the length/width of the map for the x/y axis). All points (both x and y axis values) put one by one make the genetic individual used for optimization (Fig 5).

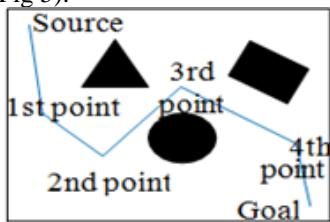


Fig 5: GA Individual representation

This algorithm can be implemented as follows.

- Generation of initial population.
- Defining the crossover operator.
- Defining the mutation operator.
- Defining the elitism operator.
- Choosing the selection method.
- Defining the fitness function.
- Choosing the algorithm termination condition.
- Defining the overall loop.

Shibata et al. [24] has developed a path planner using the concept of free convex areas introduced by Habib and Asama [25]. They used two gene-based searching algorithms to solve two easier subparts of the probem: one to find a set of optimal trajectories for each robot under selfish planning and another to select a candidate from the set of trajectories for each robot so as to avoid collisions when all robots work simultaneously. This approach does not take full advantage of the free space available at a certain time.

The Genetic algorithm is simulated using Matlab and result is as shown below

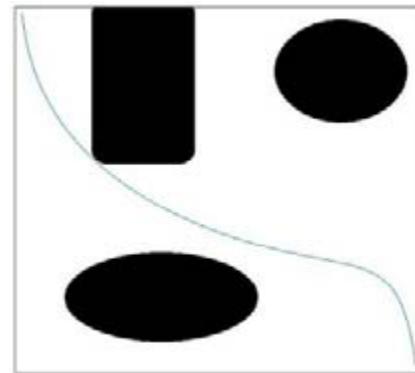


Fig. 6: GA map 1 Path Computed

Readings

Processing time=7.919825e+01

Path Length=938 e+02

7. A-STAR ALGORITHM

It is the most efficient free-space searching algorithm for path planning and obstacle avoidance [20], [21]. Initially, graph I is taken as an input which is then explored step by step till all the nodes are evaluated to find the shortest obstacle free path. Possible and impossible movements are labeled as 1 and 0, respectively [21]. If the number of 1 is more, higher are the chances of optimal path with more number of flexible turns.

This algorithm uses combination of heuristic searching and searching based on the shortest path. A* algorithm is defined as best-first algorithm, because each cell in the configuration space is evaluated by the value:

$$f(v)=h(v)+g(v)$$

Where $h(v)$ is heuristic distance (Manhattan, Euclidean or Chebyshev) of the cell to the goal state and $g(v)$ is the length of the path from the initial state to the goal state through the selected sequence of cells.

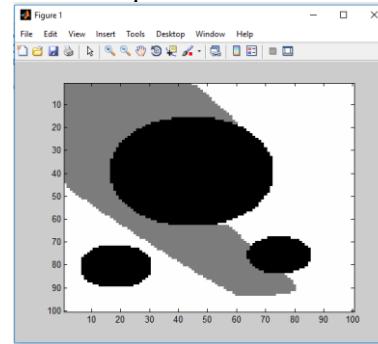


Fig. 7 (a) A* Map1 Possible path or connections

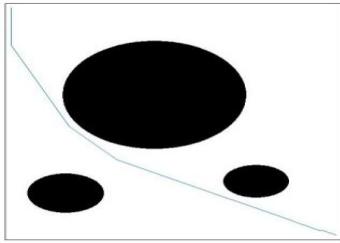


Fig7(b): Final Computed path

Readings:

Processing Time=1.003114e+02

Path Length=7.318316e+02

III. CONCLUSION

An overview of different path planning and obstacle avoidance algorithms for AMR, their strengths and weakness are presented and discussed. This review paper discusses different the robot path planning algorithms and **their simulation results** are also shown in this paper giving an insight into the positive and negative points of every algorithm. From study of different algorithms it can be concluded that the algorithm should be generic in respect to different maps and it should be capable of resulting a collision free path in less computation time and less path length thereby **saving cost** and energy.

The mobile robot path planning techniques based on predefined mapping becomes more challenging **due to unpredictable**. Heuristic approaches like genetic algorithms, Fuzzy logic, neural networks, neuro-fuzzy approach, ant colony approach provides appropriate and efficient results for an mobile robot navigation in dynamic environment. Thus heuristic approach the autonomous mobile robots can steer safely along with the all type of obstacles without colliding with them. These techniques provide optimized solution to hard problems.

REFERENCES

- [1] A. Yufka and O. Parlaktuna, "Performance Comparison of Bug Algorithms for Mobile Robots", 5th International Advanced Technologies Symposium, pp. 1-5, 2009
- [2] M.S. Ganeshmurthy, and G.R. Suresh, "Path planning algorithm for autonomous mobile robot in dynamic environment", *International Conference on Signal Processing, Communication and Networking*, vol. 3, pp. 1-6, 2015
- [3] K. M. Hasan, A. Al-Nahid, and K.J. Reza, "Path planning algorithm development for autonomous vacuum cleaner robots," *In Proceedings International Conference on Informatics, Electronics & Vision*, pp. 1-6, 2014
- [4] D. Dolgov, S. Thrun, M. Montemerlo, and J. Diebel, "Practical search techniques in path planning for autonomous driving", *AAAI Workshop - Technical Report*, 2008.
- [5] L. Huang, "Velocity Planning for a Mobile Robot to Track a Moving Target - A Potential Field Approach", *Robotics and Autonomous Systems*, vol. 57, pp. 55-63, 2009
- [6] Y.W. Chen, and W.Y. Chiu, "Optimal Robot Path Planning System by Using a Neural Network-Based Approach", *Proceedings of IEEE International Automatic Control Conference (CACS)*, pp. 85-90, 2015
- [7] E.H. Shan, B. Dai, J. Song, and Z.P. Sun, "A dynamic RRT path planning algorithm based on B-spline," *In Proc. International Symposium on Computational Intelligence and Design*, pp. 25-29, 2009
- [8] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning," *in Proc. IEEE International Conf. on Robotics and Automation*, pp. 995-1001, 2000
- [9] A. Stentz, "Optimal and efficient path planning for partially-known environments," *In Proc. IEEE International Conf. on Robotics and Automation*, pp. 3310-3317, 1994
- [10] Z.J. Du, D.K. Qu, F. Xu, and D.G. Xu, "A hybrid approach for mobile robot path planning in dynamic environments," *In Proc. IEEE International Conf. on Robotics and Biomimetics*, pp. 1058-1063, 2012
- [11] M. Samuel, M. Hussein and M.B. Mohamad, "A Review of some Pure- Pursuit based Path Tracking Techniques for Control of Autonomous Vehicle", *International Journal of Computer Applications*, vol. 135, no.1, pp. 35- 38, 2016
- [12] C. Wong, H. Wang, and S. Li, "PSO-based Motion Fuzzy Controller Design for Mobile Robots", *International Journal of Fuzzy Systems*, vol. 10, issue no. 1, pp. 284-292, 2008
- [13] H.Y. Chung, C.C. Hou and S.C. Liu, "Automatic Navigation of a heeled Mobile Robot using Particle Swarm Optimization and Fuzzy Control", *IEEE International Symposium on Industrial Electronics, Taiwan*, pp. 1-6, 2013
- [14] Y. Zhang, D.W. Gong, J.H. Zhang, "Robot Path Planning in Uncertain Environment using Multi-Objective Particle Swarm Optimization", *Elsevier Neurocomputing*, vol. 103, pp. 172-185, 2013
- [15] L.E. Zarate, M. Becker, B.D.M. Garrido, and H.S.C. Rocha, "An artificial neural network structure able to obstacle avoidance behavior used in mobile robots". *IEEE Annual Conference of the Industrial Electronics Society*, vol. 28, pp. 2457-246, 2002
- [16] S.X. Yang and C. Luo, "A neural network approach to complete coverage path planning", *IEEE Transactions on Systems, Man and Cybernetics*, vol. 34, issue 1, pp. 718-724, 2004
- [17] K. Sedighi, K. Ashenayi, R. Wainwright and H. Tai, "Autonomous local path planning for a mobile robot using a genetic algorithm", *IEEE Congress on Evolutionary Computation*, vol. 2, pp. 1338-1345, 2004
- [18] P.O. Pettersson and P. Doherty, "Probabilistic roadmap based path planning for an autonomous unmanned helicopter", *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, vol. 17, no. 4, pp. 395-405, 2005
- [19] H. Kristo, "Path Planning and Learning Strategies for Mobile Robots in Dynamic Partially Unknown Environments", *Faculty of Mathematics and Computer Science*, pp. 19-32, 2006
- [20] R. Kala, A. Shukla, R. Tiwari, S. Roongta and R.R. Janghel, "Mobile Robot Navigation Control in Moving Obstacle Environment using Genetic Algorithm, Artificial Neural Networks and A* Algorithm", *Proceedings of the IEEE World Congress on Computer Science and Information Engineering*, pp. 705-713, 2009.
- [21] A.K. Gurjri, H. Agarwal, D.K. Parsadiya, and A. Madhav, A. "Time- Efficient A* Algorithm for Robot Path Planning", *International Conference on Innovations in Automation and Mechatronics Engineering*, vol. 3, pp. 144 – 149, 2016.
- [22] Alpaslan Yufka, Osman PARLAKTUNA, "Performance Comparison Of Bug Algorithms For Mobile Robots", 5th International Advanced Technologies Symposium (IATS'09), May 13-15, 2009, Karabuk, Turkey.
- [23] Jianjun Ni, An Improved VFF Approach for Robot Path Planning in Unknown and Dynamic Environments, *Mathematical Problems in Engineering* 2014(12) · March 2014.
- [24] Stevari.M,James.J",Rapidly-Exploring Random trees : Progress and prospects.
- [25] M. Hahib and H. Asama, "Efficient Method to Generate Collision Free Paths for Autonomous Mobile Robot Based on New Free Space Structuring Approach", *Proceeding of IEEE/RSJ International Workshop on Intelligent Robotics and Systems*, 1991, pp. 563-567.