Modified A* Algorithm for Mobile Robot Path Planning

Anshika Pal, Ritu Tiwari, and Anupam Shukla

Abstract. Robot path planning is about finding a collision free motion from one position to another. Efficient algorithms for solving problems of this type have important applications in areas such as: industrial robotics, computer animation, drug design, and automated surveillance. In this paper, a modified A* algorithm is used for optimizing the path. Different from the approaches that only choose the shortest routes, this method estimates the energy consumption and chooses the most energy efficient routes. As mobile robots are powered by batteries, their energy is limited. Therefore, how to minimize energy consumption is an important problem. The basic idea is to minimize unnecessary stops and turns for mobile robots that cause acceleration and deceleration and consumes significant energy. Simulation results are presented on various environments with different levels of complexity depending on the density of the obstacles. The effectiveness of the proposed approach is evaluated in terms of number of movement steps, path length, energy consumption, number of turns and time. The experimental results show that our approach can provide effective path by reducing the number of turns compared to A*, thus saving energy. All paths generated were optimal in terms of length and smoothness.

Keywords: Path Planning, Mobile Robot, A* Algorithm, Energy Consumption, Optimized Paths, Robotics.

1 Introduction

Robotic Path Planning is one of the problems in the field of robotics that tries to find and optimize the path from the initial position to the final position [1]. Commonly, there are many paths for robot to accomplish the task, but in fact the

Anshika Pal · Ritu Tiwari · Anupam Shukla
Soft Computing and Expert System Laboratory
ABV-Indian Institute of Information Technology and Management Gwalior, India
e-mail: {anshika,ritutiwari,anupamshukla}@iiitm.ac.in

S. Patnaik & Y.-M. Yang (Eds.): Soft Computing Techniques in Vision Sci., SCI 395, pp. 183–193.
springerlink.com

best path is selected according to some guide line. These guide lines are : shortest path, least energy consuming or shortest time. So, the robot path planning is a constrained optimization problem [25]. Besides optimization, it needs to be ensured that the robot moves without any collision in the entire path it follows from the source to the destination. This would mean that the algorithm avoids all obstacles and reaches the destination starting from the source. This is also referred to as the navigation plan of the robot.

The algorithms for path planning must be evaluated in terms of completeness and computational complexity. An algorithm is said to be complete if it returns a valid solution to the path-planning problem if one exists and returns failure if and only if the problem is not feasible: This is what we will call a correct termination for a path-planning algorithm. The computational complexity of the algorithm is formulated by various problem specific performance metrics that are used for evaluation purpose [26].

The problem has been solved using numerous statistical, soft computing and other approaches [2, 20]. In this paper, a new path planning method using A* algorithm, for mobile robots is presented. The algorithm is used to find out the most optimal path of the robot [2]. This path is the final path that is used for the purpose of robotic navigation. The algorithm returns the complete path if one exists between the source and the destination. If however, no path is possible between the source and the destination, the algorithm returns null. The proposed model algorithm is computationally simple. Simulation results show that the proposed mothod is capable of planning collision-free path in various environments.

The paper is organized as follows. Section 2 summarizes the related work. The proposed approach is described in section 3. Experimental results are presented in section 4. Finally conclusion is given in section 5.

2 Related Works

The problem of path planning has been a very active area of research. The problem has seen numerous methods and means that cater to the needs of the problems. A lot of work exists to model the entire problem [1-7]. There exist good algorithms to scan the environment and represent all the obstacles in form of a grid [3]. Also various algorithms have been proposed to plan the movement of the robot using various conditions.

The research on robot path planning can be categorized into two models, based on different assumptions about the information available for planning: (1) path planning with complete information; and (2) path planning with incomplete information. The first model assumes that a robot has perfect information about itself and its environment. Information that fully describes the sizes, shapes, positions, and orientations of all obstacles in two dimensional (2D) or three-dimensional (3D) space is known. Because full information is assumed, the path planning is a one-time and off-line operation [8, 9].

In the second model, an element of uncertainty is present, and the missing data is typically provided in real time by some source of local information through sensory feedback using an ultrasound range or a vision module. A robot has no information on its environment except a start position and a target position. The sensory information is used to build a global model for path planning in real time. The path planning is a continuous on-line process. The concept of path planning algorithms in this category mostly comes from [10, 11, 12, 13].

In static environment, many algorithms have been implemented and results verified [14, 15 16, 17]. In planning dynamic environment the steps are a little different, as the environment continuously changes.

We also have various works of research in which people have tried to solve the navigation problem using genetic algorithm [14 15, 16, 18, 27, 28]. The basic principles in all these have been to take a fixed solution length and find the solutions by using genetic operators. Also similar work exists in neural network [6, 16, 19]. Here neural network has been applied mainly on static data. Genetic Algorithms are computationally much complex even for simple environments. Neural Networks are inapplicable if the path to be traced is very complex. Swarm optimization based work are presented in [29][30].

Theoretical foundations of soft computing are covered in [20]. It supplies a concise explanation of various models, principles, algorithms, tools, and techniques, including artificial neural networks, fuzzy systems, evolutionary algorithms, and hybrid algorithms.

The approach proposed in [21] finds the nearly most optimal path of the robot using Genetic, ANN and A* algorithms at each instant of time of robot travel. In this paper a heuristic function is used that optimizes the path, at the same time resolves the conflicts when two paths may have same heuristic values by considering the rotational factor as well.

In [22] MNHS based Robot Path Planning method is proposed. The motivation is to make the problem robust against the uncertainties that might arise like the sudden discovery that the path being followed does not lead to the goal.

Paper [23] solves the problem of robotic path planning using a combination of A* algorithm and Fuzzy Inference. The A* algorithm does the higher level planning by working on a lower detail map. The algorithm finds the shortest path at the same time generating the result in a finite time. The A* algorithm is used on a probability based map. The lower level planning is done by the Fuzzy Inference System (FIS).

In this paper we have modified the A* algorithm for better path planning, in terms of saving in energy.

3 Proposed Approach

3.1 Assumptions

The assumptions are divided into two parts: (1) the geometry of the environment; and (2) the characteristics and capabilities of a mobile robot.

1) Environment Assumptions

The environment is a 2D plane and is occupied by stationary obstacles. The environment includes a Start (S) and a Target (T) point. It is assumed that there are only a finite number of obstacles in the environment. The working space is partitioned into a grid of square cells, and a M X N board is gotten. If there is no obstacle in a cell, the cell is called free, otherwise called obstacle cell.

2) Mobile Robot Assumptions

The mobile robot is given the coordinates of the start, the target, and its current position. Thus, it can always calculate its direction and its distance from the target. The mobile robot has a memory to store position data and intermediate results. We assume the robot uses 45^0 as the unit for turning, since we only allow the robot to move from one cell to one of its eight neighbors.

3.2 Algorithm of Path Planning

We solve the problem using A* algorithm with additionally including an energy factor, and call it Energy-efficient A* (EA*) algorithm. In A* algorithm [24] the path scoring function uses an 'exact + estimate' cost heuristic, which is given by

$$F = G + H \tag{1}$$

Where G is cost of getting from source to current node; and H is estimated cost from the current node to target. Here cost is the Euclidean distance of two 2D points. The A* algorithm tries to minimize the path travelled and the path which is left to be travelled [18]. It hence tries to optimize the total path length to be travelled by the robot. A* generates good results for the problem of path planning, where shortest route is only a criteria of goodness. In the real world, however only shortest route might not give a very realistic picture of the problem, energy conservation also play a vital role. For example, figure 1 shows the two routes from source 'S' to target 'T'.

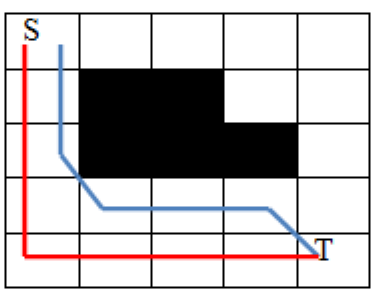

Fig. 1 Energy Efficient Route Example

Route shown by 'blue' color has four line segments, while route shown by 'red' color has only two line segments. The second route is longer but has less stops and turnings than first route.

Stops and turnings cause acceleration and deceleration that consumes significant energy. First route may be shorter but consume more energy. Hence, an energy efficient route with moderate loss in distance is always preferable. This is the motivation of the work presented in this paper.

The most essential part of the A* algorithm is a good heuristic estimate function. This can improve the efficiency and performance of the algorithm, and depends on the specific state space being explored. Our EA* algorithm is slightly different from A*, where selection of next node is not only based on the distance, it also includes robot movement direction. Robot's state is represented as its location (x, y) and direction θ; state = < x, y, θ >. We assume the robot uses 45^0 as the unit for turning. We consider the energy for stops and turns if the two states have different directions. But if two states have same directions, robot does not stop or turn, energy consumption is zero according to our assumption, the scoring function only add a movement cost which may be 1 or 1.41. Table 1 shows the energy consumption rate for different turns and stops. In EA* algorithm the path scoring function is given by

$$F = G + E + H \tag{2}$$

Where parameter G and H are same as A* algorithm. E is the energy consumption from source to current node. Results shows that, the path generated by EA* algorithm is more smooth compared to A*.

Table 1 Energy Consumption rate for stops and turns

Turns/Stops	**Energy Consumption**
Stop	0.5
45^0	0.4
90^0	0.6
135^0	0.8
180^0	1.0

4 Results and Discussions

The proposed algorithm is simulated in Java. Figure 2 shows four different workspace on which the experiments have been carried out. The workspace was taken as an input in form of an image. The image depicted the obstacles as black regions and the accessible area as the white region. The size of all the maps is 100X100. The robot was supposed to move from the top left corner to the bottom right corner. The two algorithms are run on all four maps. The objective of the experiments is to find out the influence of the energy parameter on the path followed by the robots. Performance of each of the algorithm is measured in terms of (a) number of movement steps; (b) path length; (c) energy consumption; (d) number of turns; and (e) time. One movement step is the robot's moving from one cell to one of its neighbor cells.

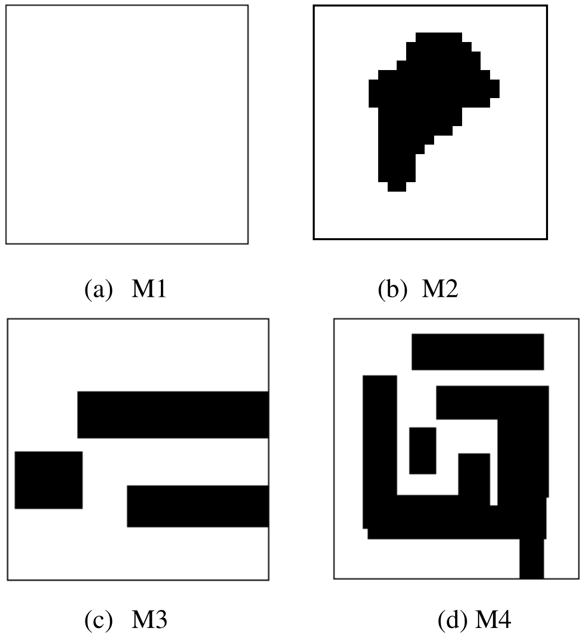

Fig. 2 Different environments in which algorithm were tested

4.1 Case I: Obstacle Free Environment

In this case experiments are performed on map M1, which is an obstacle free environment. Both EA* and A* algorithm was made to run on a completely blank map. We observed that both algorithms traced the path from the source to the destination following a straight line path. This was the possible optimal path. Figure 3 shows the snapshots of the path traversed by the robot.

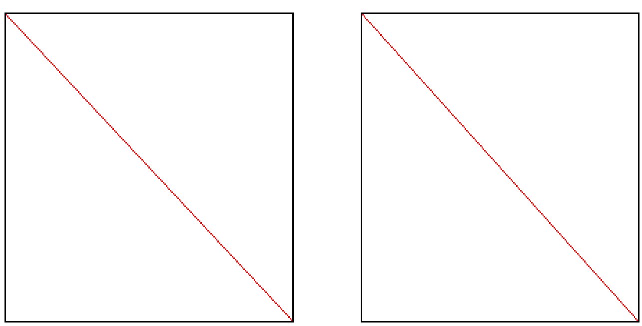

Fig. 3 Path traversed by the robot using EA* (left) and A* (right) in map M1

Table 2 presents the computed results. The analysis shows that the behavior of both algorithm is same in obstacle free environment.

Table 2 Computed results of map M1

Performance Metric	**EA***	**A***	**Analysis (EA* vs. A*)**
# Movement Steps	100	100	same
Path Length	138.6	138.6	same
Energy Consumption	1.3	1.3	same
# Turns	1	1	same
Time (in sec.)	1	1	same

4.2 Case II: Obstacle Avoidance

In this case experiments are performed on map M2, M3, and M4. Figure 4, 5 and 6 shows that, the robot easily avoided the obstacle and marched towards the target position. Computed results are presented in table 3, 4 and 5 respectively.

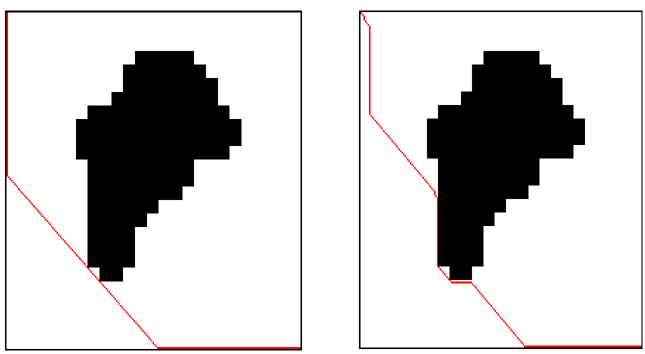

Fig. 4 Path traversed by the robot using EA* (left) and A* (right) in map M2

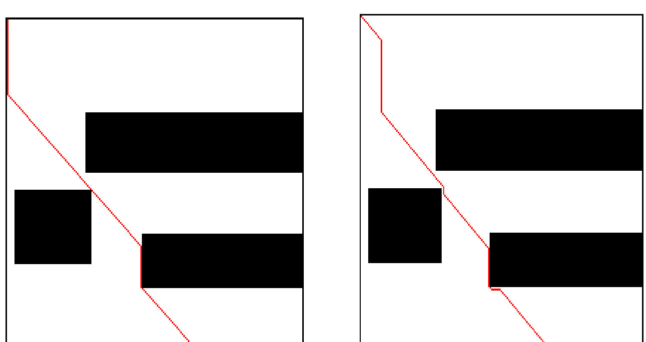

Fig. 5 Path traversed by the robot using EA* (left) and A* (right) in map M3

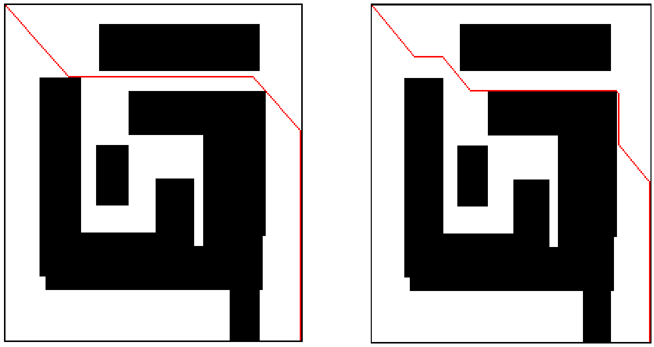

Fig. 6 Path traversed by the robot using EA* (left) and A* (right) in map M4

Table 3 Computed results of map M2

Performance Metric	EA*	A*	Analysis (EA* vs. A*)
# Movement Steps	148	148	same
Path Length	167.4	167.4	same
Energy Consumption	3.3	11.2	70.5% less
# Turns	3	12	75% less
Time (in sec.)	19.14	16.4	14.3% more

Table 4 Computed results of map M3

Performance Metric	EA*	A*	Analysis (EA* vs. A*)
# Movement Steps	134	134	same
Path Length	159	159	same
Energy Consumption	5.1	10.3	50.5% less
# Turns	5	11	54.5% less
Time (in sec.)	18.6	15	19.4% more

The results presented in table 3, 4, and 5 shows that EA* algorithm consumes on an average 56.13% less energy compared to A* algorithm. Number of turns is also 59.8% less. Although the time is on an average 14% more compared to A*. Moreover the number of movement steps and path length is same in all the maps.

Table 5 Computed results of map M4

Performance Metric	**EA***	**A***	**Analysis (EA* vs. A*)**
# Movement Steps	162	162	same
Path Length	175.8	175.8	same
Energy Consumption	4	7.6	47.4% less
# Turns	4	8	50% less
Time (in sec.)	12	11	8.3% more

5 Conclusion

In this paper, an energy saving method for mobile robot path planning based on A* algorithm has been presented. Energy Conservation is more crucial in any robotic mission, because energy is limited in robots, and in order to complete the mission it is necessary robots never die due to low energy. Various parameters have been defined for the purpose of evaluating the method. The simulation results that have been taken showed; the mobile robot travels successfully from starting point and reaching its target point. All obstacles that are located in its way have been avoided. Experiments show that great improvements are to be found in terms of energy consumption and number of turns. The discovered path is smoother. It is worth noting that A* strategy yields better results than EA* technique only in terms of time, although the improvement is not significant. The proposed algorithm is effective and efficient, because if a path takes quite more time but if the energy consumption is very less, so this path is always preferable.
Table 6, summarizes the analysis that have been done.

Table 6 Results' Summary (EA* vs. A*)

Parameter	**Conclusion (on average basis)**
Energy Consumption	56.13% less
# Turns	59.8% less
Time	14% more
# Movement Steps & Path Length	same

References

[1] Hutchinson, S.A., Kak, A.C.: Planning sensing strategies in a robot work cell with Multi-sensor capabilities. IEEE Trans. on Robotics and Automation 5(6) (1989)

[2] Rich, E., Knight, K.: Artificial Intelligence, pp. 29–98. McGraw- Hill, New York (1991)

[3] Takahashi, O., Schilling, R.J.: Motion planning in a plane using generalized voronoi diagrams. IEEE Trans. On Robotics and Automation 5(2) (1989)

[4] Borenstain, J., Everett, H.R., Feng, L.: Navigating Mobile Robots: Systems and Techniques. A. K. Peters, Wellesley (1996)
[5] Matlin, W.M.: Cognition, Hault Sounders, printed and circulated. Prism books, India (1996)
[6] Konar, A., Pal, S.: Modeling cognition with fuzzy neural nets. In: Leondes, C.T. (ed.) Fuzzy Systems Theory: Techniques and Applications. Academic Press, New York (1999)
[7] Pagac, D., Nebot, E.M., Durrant, W.H.: An evidential approach to map building for autonomous robots. IEEE Trans. On Robotics and Automation 14(2), 623–629 (1998)
[8] Lumelsky, V.J., Skewis, T.: A paradigm for incorporating vision in the robot navigation function. In: Proc. of IEEE Int. Conf. on Robotic Automation, pp. 734–739. Institute of Electrical and Electronics Engineers, New York (1988)
[9] Lumelsky, V.J., Stepanov, A.A.: Path-planning strategies for a point mobile automation moving amidst unknown obstacles of arbitrary shape. Algorithmica 2(4), 403–430 (1987)
[10] Lumelsky, V.J., Tiwari, S.: An algorithm for maze searching with azimuth input. In: Proc. IEEE Int. Conf. on Robotic Automation, pp. 111–116. Institute of Electrical and Electronics Engineers, New York (1994)
[11] Lee, S., Adams, T.M., Byoo, B.: A fuzzy navigation system for mobile construction robot. Autom. Constr. 6(2), 97–107 (1997)
[12] Lee, S.: Spatial model and decentralized path planning for construction automation. PhD thesis, University of Wisconsin–Madison, Madison, Wis. (2000)
[13] Kamon, I., Rivlin, E.: Sensory-based motion planning with global proofs. IEEE Trans. Rob. Autom. 13(6), 812–814 (1997)
[14] Ayala-Ramirez, V., Perez-Garcia, A., Montecillo- Puente, E.J., Sanchez-Yanez, R.E.: Path planning using genetic algorithms for mini-robotic tasks. In: 2004 IEEE International Conference on Systems, Man and Cybernetics (2004)
[15] Manikas, T.W., Ashenayi, K., Wainwright, R.L.: Genetic Algorithms for Autonomous Robot Navigation. IEEE Instrumentation & Measurement Magazine (December 2007)
[16] Du, X., Chen, H.-H., Gu, W.-K.: Neural network and genetic algorithm based global path planning in a static environment. Journal of Zhejiang University Science 6A(6), 549–554 (2005)
[17] Behnke, S.: Local Multiresolution Path Planning. In: Polani, D., Browning, B., Bonarini, A., Yoshida, K. (eds.) RoboCup 2003. LNCS (LNAI), vol. 3020, pp. 332–343. Springer, Heidelberg (2004)
[18] Mahmoudi, S.E., Bitaghsir, A.A., Forouzandeh, B., Marandi, A.R.: A New Genetic Method for Mobile Robot Navigation. In: 10th IEEE International Conference on Methods and Models in Automation and Robotics, Miedzyzdroje, Poland, August 30 - September 2 (2004)
[19] Na, Y.-K., Oh, S.-Y.: Hybrid Control for Autonomous Mobile Robot Navigation Using Neural Network Based Behavior Modules and Environment Classification. Autonomous Robots 15(2), 193–206
[20] Shukla, A., Tiwari, R., Kala, R.: Real Life Applications of Soft Computing, May 21. CRC Press Taylor & Fransis Group (2010)

[21] Rahul, K., et al.: Mobile Robot Navigation Control in Moving Obstacle Environment using Genetic Algorithm, Artificial Neural Networks and A* Algorithm. In: Proceedings of the IEEE World Congress on Computer Science and Information Engineering (CSIE 2009), IEEE Explore, Los Angeles/Anaheim, USA, pp. 705–713 (April 2009)

[22] Rahul, K., Anupam, S., Ritu, T.: Robotic Path Planning using Multi Neuron Heuristic Search. In: Proceedings of the ACM 2009 International Conference on Computer Sciences and Convergence Information Technology, ICCIT 2009, Seoul, Korea, pp. 1318–1323

[23] Rahul, K., Anupam, S., Ritu, T.: Fusion of probabilistic A* algorithm and fuzzy inference system for robotic path planning. Artificial Intelligence Review 33(4), 275–306 (2010)

[24] Hart, P.E., Nilsson, N.J., Raphael, B.: A formal basis for the heuristic determination of minimum cost paths. IEEE Transactions on Systems Science and Cybernetics SSC4(2), 100–107 (1968)

[25] Liu, G., et al.: The Ant Algorithm for Solving Robot Path Planning Problem. In: Third International Conference on Information Technology and Applications (ICITA), pp. 25–27 (2005)

[26] Frazzoli, E., et al.: Real-Time Motion Planning for Agile Autonomous Vehicles. Journal of Guidance, Control and Dynamics 25(1) (January–February 2002)

[27] Liang, Y., Xu, L.: Global path planning for mobile robot based genetic algorithm and modified simulated annealing algorithm. In: Proceedings of the first ACM/SIGEVO Summit on Genetic and Evolutionary Computation, pp. 303–308 (2009)

[28] Dai, S., et al.: Path Planning for Mobile Robot Based On Rough Set Genetic Algorithm. In: Second International Conference on Intelligent Networks and Intelligent Systems, pp. 278–281 (2009)

[29] Masehian, E., Sedighizadeh, D.: A multi-objective PSO-based algorithm for robot path planning. In: 2010 IEEE International Conference on Industrial Technology (ICIT), pp. 465–470 (2010)

[30] Kim, S.-H., et al.: New Potential Functions for Multi robot path planning: SWARM or SPREAD. In: 2nd International Conference on Computer and Automation Engineering (ICCAE), pp. 557–561 (2010)