Comparison of optimal path planning algorithms

Conference Paper · February 2018

DOI: 10.1109/TCSET.2018.8336197

2 authors:

Mehmet Korkmaz
Aksaray Üniversitesi

Akif Durdu
Konya Technical University

Comparison of optimal path planning algorithms

Mehmet Korkmaz
Department of Electrical and Electronics Engineering
Aksaray University
Aksaray, Turkey
mkorkmazeem@gmail.com

Akif Durdu
Department of Electrical and Electronics Engineering
Selcuk University
Konya, Turkey
akifdurdu@gmail.com

***Abstract*— This work is concerned with path planning algorithms which have an important place in robotic navigation. Mobile robots must be moved to the relevant task point in order to be able to fulfill the tasks assigned to them. However, the movements planned in a frame or random may affect the duty time and even in some situations, the duty might be failed. When such problems are taken into consideration, it is expected that the robots should go to the task point and complete their tasks within the shortest time and most suitable way. It is aimed to give results about a comparison of some known algorithms. With this thought, a map for a real time environment has been created and the appropriateness of the algorithms are investigated with respect to the described starting/end points. According to the results, the shortest path is found by the A* algorithm. However, it is observed that the time efficiency of this algorithm very low. On the other hand, PRM algorithm is the most suitable method in terms of elapsed time. In addition to this, algorithm path length is closer to the A* algorithm. The results are analyzed and commented according to the statistical analysis methods.**

Keywords—path planning methods, SLAM, statistical analysis

I. INTRODUCTION

Mobile robots take more and more parts in our daily life. As a conclusion of this increasing role, the tasks normally carried out by humans are assigned to the robots. This thought sometimes makes easier the human life. However in some cases, it plays a vital role in which the use of robots instead of humans. The Fukushima nuclear disaster [1] is a dramatic instance of this situation. It was highly possible to prevent most deaths by using the robots instead of the kamikazes. From this point of view, it is obvious that the mobile robots will be an indispensable part of our future life [2].

In order for the mobile robots to able to perform the assigned tasks, they have to know their locations within the environment and comprehend what the world around them looks like. These problems are investigated within the scope of localization and mapping phenomenon in the robotic literature [3]. On the other hand, even if the robots have the information about their location and environment, they also have to choose an optimal path from the start to the end point in order to reach the target. The path that the robot follow has to be specified, but this fact remains that what criteria should be considered for the choice of the evaluation of the optimal path. This problem is handled as a path or route planning problem in the literature [4, 5, 6, 16].

The constraints such as the obstacles within the environment, the dangerous fields have to be specified in order to obtain the optimal route. It is benefited from a variety of methods like graph theory, probabilistic or heuristic-based optimization methods to find a solution to this problem [4, 7, 15]. In addition to this, online and offline solutions are sought in some situations [6].

The paper is organized as following parts. Section II mentions briefly the general optimal path problem definitions and the methods which are executed in this study. Section III gives the implementation scheme and results which belongs the real time experiment. Finally, conclusions and statistical analysis are presented and discussed in section IV.

II. PATH PLANNING PROBLEM & ALGORITHMS

A. *Path Planning Problem*

Path or route planning is defined as a problem that an agent moves from starting to end point by avoiding obstacles in order to reach the target at minimum cost. The starting and end points are might be same (loop closure) or different. The cost to find the optimal path varies under the different circumstances (time, distance, energy etc.). It is generally desired to have the shortest distance between the start/end points. However, the definition of the optimality changes in some situations. For example, if the tasks are successively assigned to the robots, the elapsed time to calculate the optimal path also matters. When the computation time is too long to obtain the optimal path, it would be hard to provide the task continuity. Because of this reason, the most suitable algorithm has to be chosen according to the desired optimality criterion. In some cases, it might be useful to combine the criteria [4, 11, 14, 16].

B. *Algorithms*

There are several approaches described in the literature to obtain the optimal path. The core output of these heuristic or probabilistic based algorithms is the optimal path.

One of the well-known and basic heuristic algorithm is *A star search* (*A** or *A-star* or *A* search*). It is a combination of Dijkstra's algorithm. The algorithm tries to minimize the function formulized as $f(n) = g(n) + h(n)$ heuristically considering the relationship between the nodes and the edges. $g(n)$ refers to the cost of starting point or node and $h(n)$ implies a heuristic cost estimation related to the remaining path. $h(n)$ hereby constitute the heuristic base of the algorithm [8, 16].

The other popular heuristic approach is the *genetic algorithm (GA)*. This method imitates the evolutionary process and attempts to acquire the best individuals (chromosome) which

978-1-5386-2556-9/18/$31.00 ©2018 IEEE

represent the optimal path for this type of problems with regard to defined objective function [9]. The crossover strategy is applied to the parents in order to create new individuals. In addition to this, the mutation process which prevents algorithm to converge on local minima is one of the crucial parts, so that the variety of the solution space is always preserved. *GA* and similar heuristic methods are an alternative way for the solution of the optimal path problem and they are often utilized [10].

Rapidly-Exploring Random Tree – RRT is another probabilistic based algorithm which is improved by Lavalle and Kuffner [11, 12]. It is aimed at the solution of the path planning problems which have nonholonomic constraints. The algorithm generally gives effective results in non-convex and high-dimensional spaces. The tree which implies the solution space is built incrementally and two different functions which are the creation and the expansion of the tree are used for all over the scheme. The expansion of the tree is one-way which is from starting to end point. On the other hand, the idea of the expansion of the tree in a bidirectional way is improved the time efficiency of the whole solution. This technique is called *bidirectional RRT (bRRT or B-RRT)* and the tree is established in both starting and target points and then expanded into the unsearched parts. The search process is terminated when the junction point of these two trees are found and this path which is the output of the algorithm is an optimal route [4, 5, 13].

One of the probabilistic-based and last path planning algorithm used in this study is *Probabilistic Roadmap (PRM).* The optimal path is acquired by the distance calculation of the edges which are the connections between randomly created nodes on the map. First of all, the nodes are generated by sampling from the non-obstacle points on the map and then these nodes are linked to each other and lastly, the path cost is evaluated. The rapid random cluster strategy may provide faster results than the other algorithms but not guarantee the shortest path [14, 15, 16].

III. Implementation

The map of the real time environment illustrated in Figure 1 is obtained by *gmapping* algorithm which works on *ROS (Robot Operating System)* for the purpose of the implementation of the proposed comparisons of the path planning algorithms. *ROS* is a framework that has been recently used in many robotic applications because of the fact that it allows the users to implement algorithms easier in both real-time and simulation. *gmapping* is a type of SLAM (Simultaneous Localization and Mapping) algorithm which is a widespread and powerful way to build a map of the environment. The inputs of the algorithm are robot odometry and sensor data acquired from the environment and the output of it is the environment map. The map is in the form of an occupancy grid in which the white fields represent the free areas where the robot is able to move and the black regions reflect the obstacles. In some cases, it is possible to see gray parts in this kind of algorithms which means that these areas are not explored and unknown whether free or occupied while mapping process.

The starting and end points are defined for the map of the environment and the algorithms are operated from these points, for a fair comparison. The main objective of all algorithms employed in the study is to find the optimal path within the shortest time and distance.

The related starting and end points on the map are as follows:

- Starting point [X, Y] : [89, 237]
- End point [X, Y] : [353, 152]

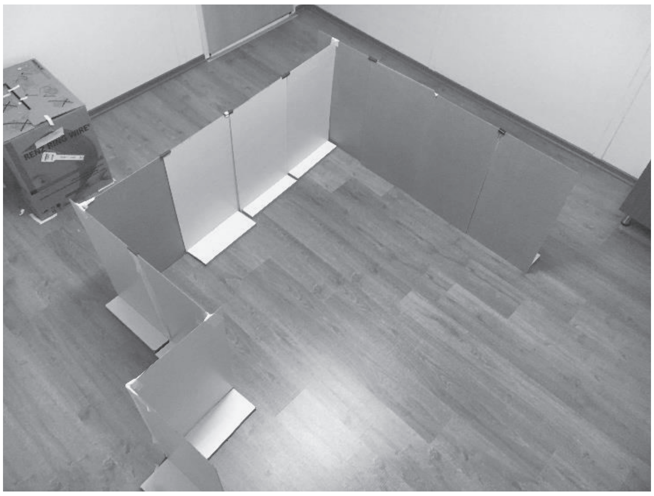

Fig. 1. Real-time environment

Occupancy grid based maps generally fit well with the *A** algorithm. This algorithm is utilized from the graph theory to get optimal path on this kind of discrete maps. The cost from a node to target is computed considering the correlation or cost matrix (Figure 2).

Fig. 2. Optimal path of the A* search algorithm

The shortest distance between the starting and target points are thought as an objective function of the *Genetic Algorithm*. The optimal path is desired to find within the free regions and the obstacles are penalized with a high penalty score (Figure 3).

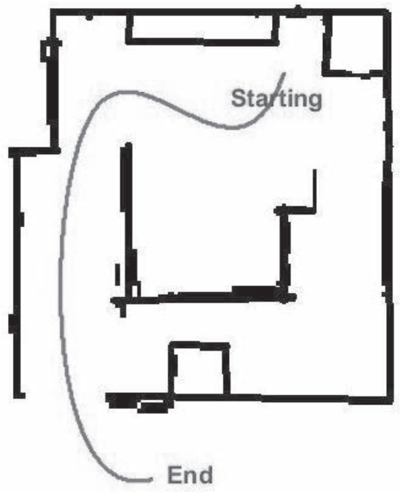

Fig. 3. Optimal path of the GA algorithm

Figure 4 and 5 are the results of the best trees of the *RRT* and *bRRT* algorithms.

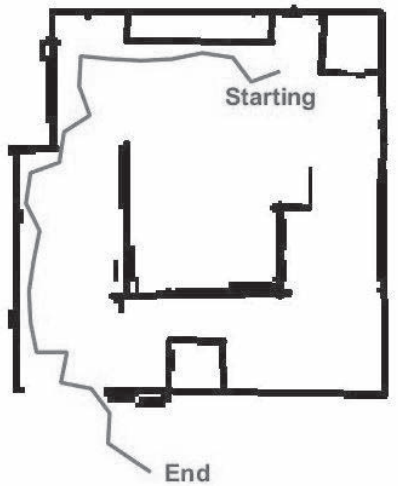

Fig. 4. Optimal path of the RRT algorithm

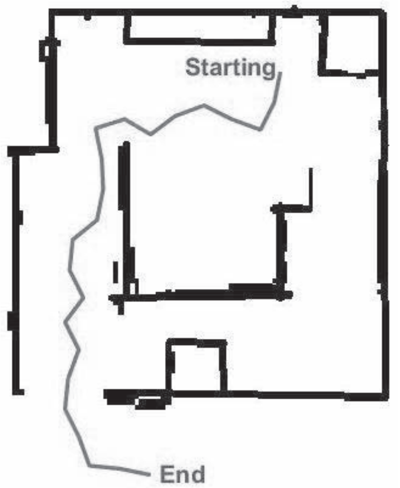

Fig. 5. Optimal path of the bRRT algorithm

In addition to these algorithms, the optimal path of the *PRM* algorithm is illustrated in Figure 6.

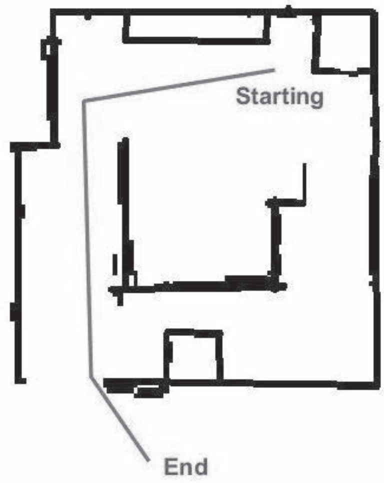

Fig. 6. Optimal path of the PRM algorithm

All of these results are one solution to the mentioned problem within the solution space. Apart from *A** algorithm, the other ones are highly likely to produce different solution sets in each operation. Due to this reason, the results have to be interpreted from the point of statistical analysis methods. With this aim, all algorithms are run 10 times and path distance and consumption time of them are saved. One-way *ANOVA (Analysis of Variance)* [17] is used to determine whether there are any statistically significant differences between the groups (Table 1 to 4).

TABLE I. ONE-WAY ANOVA: TIME VERSUS ALGORITHM

Source	DF	SS	MS	F	P
Algorithm	4	42753,73	10688,43	9252,04	0,000
Error	45	51,99	1,16		
Total	49	42805,72			

S = 1,075; R-Sq = 99,88%; R-Sq(adj) = 99,87%

TABLE II. INDIVIDUAL 95% CIS FOR MEAN BASED ON POOLED STDEV AND GROUPING INFORMATION USING FISHER METHOD

Level	N	Mean	StDev	Grouping
A*	10	76,960	0,000	A
bRRT	10	1,848	0,320	C , D
GA	10	25,886	2,335	B
PRM	10	1,009	0,070	D
RRT	10	2,154	0,464	C

Pooled StDev = 1,075

According to the analysis results, *bRRT* and *PRM* gives the best scores from the point of time.

TABLE III. ONE-WAY ANOVA: DISTANCE VERSUS ALGORITHM

Source	DF	SS	MS	F	P
Algorithm	4	252422	63106	10,80	0,000
Error	45	262926	5843		
Total	49	515349			

S = 76,44; R-Sq = 48,98%; R-Sq(adj) = 44,45

According to the analysis results, *A** and *PRM* gives the best scores from the point of distance.

All these analyses reveal that *PRM* algorithm is involved in the best groups both in shortest time and distance. It is clearly seen that *PRM* algorithm gives the best results in terms of both shortest time and distance parameters for a problem like this.

TABLE IV. INDIVIDUAL 95% CIS FOR MEAN BASED ON POOLED STDEV AND GROUPING INFORMATION USING FISHER METHOD

Level	N	Mean	StDev	Grouping
A*	10	368,00	0,000	D
bRRT	10	457,50	37,26	B
GA	10	568,90	135,44	A
PRM	10	383,30	14,18	C, D
RRT	10	443,90	96,34	B, C

Pooled StDev = 76,44

On the other hand, in some cases it might be necessary to quick assessments of the algorithms in the sense of both in optimal path and time with a combined criteria which explicitly divulge the best one. The statistical analysis might be tiresome for these situations. Because of this reason, we have developed a comparison criterion that shows the optimal method regarding the shortest distance and time (Eq. 1 and 2).

$$\varepsilon_i = t_i * x_i \quad (1)$$

$$\eta_i = inv(\varepsilon_i / max|\varepsilon_i|) \quad (2)$$

where, t_i represents an elapsed time and x_i is a distance of each algorithm. η_i is an efficiency factor of any algorithm. Table 5 shows the results of one operation within 10.

TABLE V. TIME / DISTANCE / EFFICIENCY COMPARISONS

Algorithm	Time (s)	Distance	Efficiency
A*	76.96	368	1
bRRT	1.98	434	32.95778
GA	24.62	524	2.195298
PRM	0.95	397	75.09288
RRT	2.38	446	26.68094

The algorithm we enhanced firstly calculates the total cost regarding the time and distance. After that, all costs are normalized with respect to the maximum cost value. Lastly, all these values are reversed to find the efficiency level of the algorithm. The higher efficiency value means that related algorithm is the best one in terms of shortest distance and time.

IV. RESULT AND DISCUSSION

In this study, it is aimed to find the optimal path of a robot. With this thought, the real time environment map is built with SLAM algorithm, *gmapping*. The optimal path is defined as a combined criteria which finds the shortest distance within the shortest time. Within this scope, time and distance efficiency of the algorithms described in the literature are sought and compared.

In the light of the study findings, it is clearly seen that *A** algorithm is the best one regarding the shortest distance. Nevertheless, this algorithm is in need of a long computational time in order to obtain the shortest distance. The algorithm will not be appropriate to the situations in which sequential tasks are assigned to the robot. On the other hand, *PRM* algorithm time result is impressive compared to the A*. Besides, the distance from the starting to the target points is not much different from the A*.

As a conclusion, it is important to find optimal path for the mobile robots in a shortest time and distance in order to fulfill the tasks assigned to them. Within this framework, the effectiveness of some algorithms defined in the literature is investigated and compared. It is observed that PRM algorithm is suitable for this kind of problems from the angle of time and distance. All comments related to the best algorithm are made based on the statistical analysis methods. In addition to this, it is improved a basic but effective combined criterion which involves both the shortest time and distance with the thought that it might be necessary a quick comparison of the trials. The improved criterion results match up with statistical analysis method ones and it can be used for a quick comparison.

REFERENCES

[1] Aoki, Masahiko, and Geoffrey Rothwell. "A comparative institutional analysis of the Fukushima nuclear disaster: Lessons and policy implications." Energy Policy 53 (2013): 240-247.

[2] Siegwart, Roland, Illah Reza Nourbakhsh, and Davide Scaramuzza. Introduction to autonomous mobile robots. MIT press, 2011.

[3] Durrant-Whyte, Hugh, and Tim Bailey. "Simultaneous localization and mapping: part I." IEEE robotics & automation magazine 13.2 (2006): 99-110.

[4] Berntorp, Karl. "Path planning and integrated collision avoidance for autonomous vehicles." American Control Conference (ACC), 2017. IEEE, 2017.

[5] Lacevic, Bakir, Paolo Rocco, and Morten Strandberg. "Safe motion planning for articulated robots using RRTs." Information, Communication and Automation Technologies (ICAT), 2011 XXIII International Symposium on. IEEE, 2011.

[6] Liu, Ming. "Robotic online path planning on point cloud." IEEE transactions on cybernetics 46.5 (2016): 1217-1228.

[7] W. Zhang; X. Gong; G. Han; Y. Zhao, "An Improved Ant Colony Algorithm for Path Planning in One Scenic Area with Many Spots," in IEEE Access , vol.PP, no.99, pp.1-1

[8] Yao, Junfeng, et al. "Path planning for virtual human motion using improved A* star algorithm." Information Technology: New Generations (ITNG), 2010 Seventh International Conference on. IEEE, 2010.

[9] Whitley, Darrell. "A genetic algorithm tutorial." Statistics and computing 4.2 (1994): 65-85.

[10] Alnasser, Sarah, and Hachemi Bennaceur. "An efficient Genetic Algorithm for the global robot path planning problem." Digital Information and Communication Technology and its Applications (DICTAP), 2016 Sixth International Conference on. IEEE, 2016.

[11] Kavraki, Lydia E., and Jean-Claude Latombe. "Probabilistic roadmaps for robot path planning." (1998).

[12] Baumann, Matthew, et al. "Path planning for improved visibility using a probabilistic road map." IEEE Transactions on Robotics 26.1 (2010): 195-200.

[13] Meng, Li, Song Qing, and Zhao Qin Jun. "UAV path re-planning based on improved bidirectional RRT algorithm in dynamic environment." Control, Automation and Robotics (ICCAR), 2017 3rd International Conference on. IEEE, 2017.

[14] Kavraki, Lydia E., and Jean-Claude Latombe. "Probabilistic roadmaps for robot path planning." (1998).

[15] Baumann, Matthew, et al. "Path planning for improved visibility using a probabilistic road map." IEEE Transactions on Robotics 26.1 (2010): 195-200.

[16] Tiwari, Ritu, ed. Intelligent planning for mobile robotics: algorithmic approaches: algorithmic approaches. IGI Global, 2012.

[17] Yarnold, Paul R. "Novometrics vs. ODA vs. One-Way ANOVA: Evaluating comparative effectiveness of sales training programs, and the importance of conducting LOO with small samples." *Analysis* 4 (2016): 27-28.