PERFORMANCE COMPARISON OF GENETIC ALGORITHM AND A* IN PATH PLANNING FOR MOBILE ROBOTS

ADEM TUNCER

Computer Engineering Department, Yalova University, Yalova, Turkey
E-mail: adem.tuncer@yalova.edu.tr

Abstract- This paper explainsa path planning of mobile robots based on Genetic Algorithm (GA) and a performance comparison with A* algorithm. The path planning is an important issue in mobile robots. The path planning of mobile robots can be defined as finding a suitable and feasible path given a starting location, a target location, and a set of obstacles in an environment. In this paper, path planning under different environments are carried out, and the comparative studies with GA and A* algorithm are presented.Experimental comparisons are examined in terms of both the solution time and path distance. The results show the effectiveness of the GA, including a comparison with the A* algorithm.

Keywords- A* Algorithm, Genetic Algorithm, Mobile Robots, Path Planning

I. INTRODUCTION

There are many studies in the literature about mobile robots. One of the main subjects on mobile robots is path planning. The path planning of mobile robots can be defined as finding a suitable and feasible path given a starting location, a target location, and a set of obstacles in an environment. Several methods and algorithms have been developed to overcome the path planning problems such as Dijkstra, A* algorithm, potential field method. When methods compared with each other, although each method has its own advantages, disadvantages also available, e.g., long calculation process, local optimum.Researchers still continuesearching alternative and more efficient methods to solve the path planning problems. These methodsused to solve the path planning problems are determinedaccording to two factors; first one is the environment type (i.e., static or dynamic), the second is the path planning algorithms (i.e., global or local). If the locations of obstacles in an environment do not change during the path planning process, it is defined as a static environment. However, if the locations of obstacles changes, it is defined as a dynamic environment. In global path planning, the environment has to be known and the environment should be static. On the other hand, local path planning means that path planning is done while the mobile robot is moving; in other words, the algorithm is capable of producing a new path in response to environmental changes.

One of the main problems of mobile robots is to find the shortest path from a starting to a target node. The path planning problem can be addressed as an optimization problem when it is considered as the shortest or the least-cost path. GA has been recognized as one of the most robust search algorithms for complex and ill-behaved optimization problems. In recent years, the path planning problems have been widely solved using the GA, which isa heuristic algorithm. Proposed by Holland, GA is an optimization technique based on genetic science and has been applied to many hard optimization problems. It is able to reach a better solution quickly, since it searches simultaneously all the search space in a parallel manner.One of the main difficulties in finding an optimal path, is the size and complexity of the environment. It has widely been recognized that GA works even for complex problems such that traditional algorithms cannot find an acceptable solution within a reasonable amount of time.

II. OVERVIEW OF A*

A* algorithm is used to find the least-cost path between a starting node and a target node. This algorithm uses mainly an evaluation functionf(n) is defined as following;

$$f(n) = g(n) + h(n) \quad (1)$$

The distance function g(n) is defined as the path length between the starting node and the current node. h(n) is an admissible “heuristic estimate” of the distance from starting node to the target node. It might represent the straight-line distance to the target, since that is the smallest possible distance between any two nodes. There are two lists as open and closed in this algorithm. The open list contains all new nodes that could be visited in the next step from the already visited nodes. The closed list contains all nodes that were already visited. The open list is initialized with the start node. The algorithm finds the shortest path once the target node is added to the open list. The closed list starts empty.

III. PATH PLANNING BASED ON GENETIC ALGORITHM

GA starts an optimization with a population consisting of a set of random solutions. Each candidate solution in the population is named as a chromosome.After fitness function values of all the chromosomes find, selection operation is carried out according to these

values. Then, selected chromosomes are subjected to crossover and mutation operations. After a new generation is generated, each new chromosome is tested with a fitness function.GA operators used for path planning problems are described below.

A. Representation of Environment and Chromosome

In order to find a collision free path usingGA, the first step is to define a suitable representation environment and chromosome. A grid based model is commonly used to represent the environment in many path planning methods. In this method, the environment is divided into a number of grids in a two-dimensional coordinate plane. It has been determined that calculation of distance and representation of obstacle is easier with this model.

Fig.1 shows an example environment, which has 10x10 grids and a suitable path from starting (S) node to target (T) node. Free grids indicate suitable areas which mobile robot goes over them and gray grids indicate obstacles.

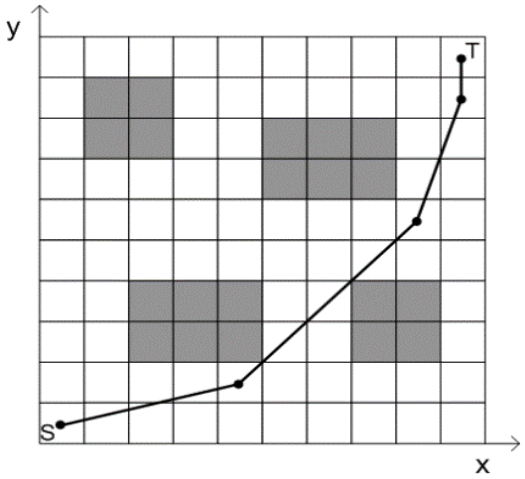


Fig.1. Grid-based environment representation

A path for mobile robots in their environment is a chromosome. Every chromosome can be represented as coordinates of the path. Different methods are used for coding chromosome, depending on the representation method of the environment such as binary, decimal. In this paper, the decimal coding method is used. Fig.2 shows an example, decimal coded chromosome as a two-dimensional coordinate plane.

$$\{(x_{start}, y_{start}), (x_1, y_1), (x_2, y_2), \dots, (x_{target}, y_{target})\}$$

Fig.2. Decimal coded chromosome

For example, the feasible path in Fig. 1 can be encoded as a {(0,0),(4,1),(8,5),(9,8),(9,9)}.

B. Fitness Function

One of the necessary steps inGA is the determination of the fitness function. The objective in path planning is to find a path without any collision. Optimal path may be the shortest, the least time and energy requiring path to trip on it. In the path planning problems, generally, the fitness function is considered as a shortest path. Fitness function used in this study, f is defined as following (2):

$$f = \begin{cases} \sum_{i=1}^{n-1} d(p_i, p_{i+1}) & \text{, for feasible paths} \\ \sum_{i=1}^{n-1} d(p_i, p_{i+1}) + \sum_{j=1}^{m} penalty & \text{, for infeasible paths} \end{cases} \quad (2)$$

Where; p_i is the i^{th} gene of the chromosome, n is the length of the chromosome, d is the Euclidean distance of the two nodes, and m is number of obstacles between two nodes. The fitness function is defined as the length of the path. If there is an obstacle between two nodes, a penalty is added to the fitness function value. The penalty value should be greater than the maximum path length on the environment. In order to find an optimal path, the algorithm searches for a chromosome whose penalty is eliminated.

C. Selection Operation

The main principle of the GA is that the best genes on the chromosomes should be survived and transferred to new generations. A selection operation needs to be done to determine the best chromosomes. In this operation, chromosomes are selected according to fitness values and then put into a mating pool to produce new chromosomes. The roulette wheel selection method is used in this study and elitism is applied.

D. Crossover Operation

A crossover operation is a major object to generate a new population in GA. The crossover operator crosses two selected parent chromosomes to generate two new offspring. Single-point crossover is used in this study. The function of the crossover operator is illustrated in Fig. 3.

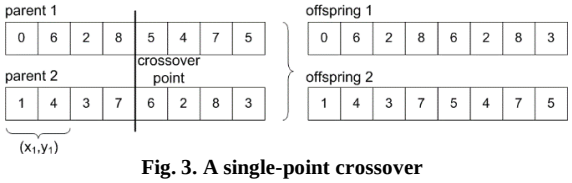


Fig. 3. A single-point crossover

E. Mutation Operation

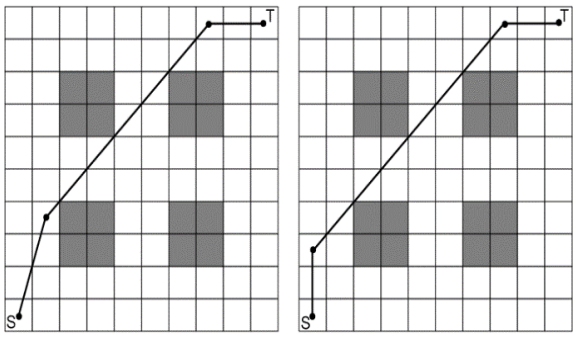
Mutation operation changes certain genes randomly in chromosome to introduce new individuals, and it increases the diversity of the population. The mutation operation expands the search space to regions that may not be close to the current population, thus ensuring a global search. How many of the genes in the population will be modified is controlled by the mutation rate. The mutation rate usually should be in the interval between 0.001 and 0.1. There are different mutation operations in the literature. In this study random mutation operation is used.

IV. EXPERIMENTAL AND PERFORMANCE RESULTS

For this study, environments with obstacles are represented by grids with 10x10, 50x50, 100x100, and 200x200, respectively. GA and A* algorithm are applied to these environments and compared in terms of solution times and path distances.

The parameters of the GA are as follows; the population size is taken as 20, crossover probability is taken as 1 and mutation probability is taken as 0.1.

Fig.4 (a) shows the path found with GA, Fig. 4 (b) shows the path found withthe A* algorithm for 10x10 grids environment with obstacles.



(a) **(b)**

Fig.4. 10x10 grids environment with obstacles

Fig. 5 shows the path both the GA and A* algorithm for50x50 grids environment with obstacles.

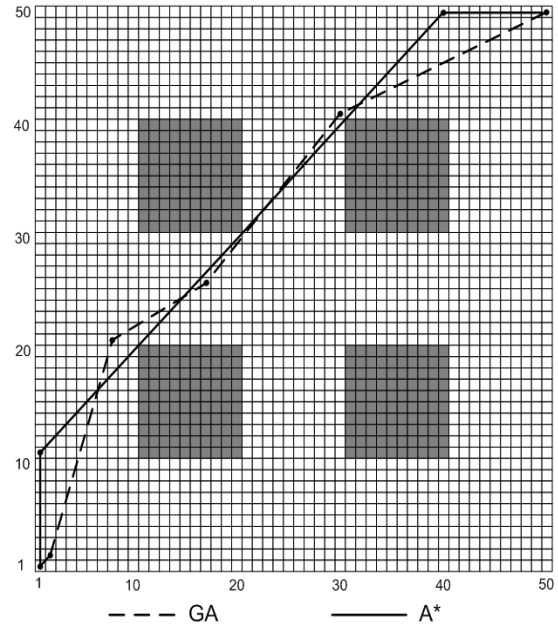


Fig. 5. 50x50 grids environment with obstacles

According to the figures above, it is clearly seen that the paths found withGA are more optimal thanA* algorithm.Table I gives the fitness function values under different sized environment separately forGA andA* algorithm.

Table I. Fitness Functions of theGA and A* Algorithm

Environment Size	GA	A*
10 x 10	13.6476	13.8995
50 x 50	73.4158	75.1543
100 x 100	161.6145	163.4386
200 x 200	324.112	328.2914

According to Table I, while the fitness value is 13.6476 found withGA for 10x10 grids environment, the fitness value is 13.8995 found withA* algorithm. As seen from the results on the table, GA found shorter paths. If the path planning problems are taken as a finding the shortest path, it is clearly seen that the GA is more effective according to A* algorithm.

Table II gives the average solution times in 10 trials for GA andA* algorithm, respectively.

Table II. Solution times of the GA and A* Algorithm

Environment Size	GA (sec.)	A* (sec.)
10 x 10	0.47	0.11
50 x 50	1.3	0.66
100 x 100	2.78	7.49
200 x 200	5.23	98.66

According to the table II, while the solution time of GA increases 11 times, the solution times ofA* increases 997 times, between 10x10 and 200x200 grids. These results show that A* algorithm is useful in small environments, but the solution time of this algorithm increasesrapidly for larger environments. GA always takes acceptable solution times for all environments. Fig. 5 shows that the solution times graphics of two algorithms. It is observed that the GA increases linearly, while the A* algorithm increases exponentially.

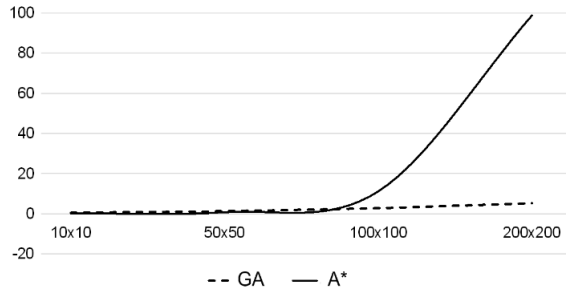


Fig. 5. The Solution Times Graphicof GA and A* algorithm

CONCLUSION

In this paper, path planning under different environments are carried out, and the comparative studies with GA andA* algorithm are presented. The

simulation results showed that the A* algorithm finds solution faster than the GA in smaller areas. However, when the environment gets larger, it takes longer time exponentially for A* algorithm to find a solution compared with the GA. In addition to solution times,comparisons are also examined in terms of path distance. The GA has much better performance in finding shorter path. Experimental results show that the GA is much more efficient than the A* algorithm.

REFERENCES

[1] E. W. Dijkstra, "A Note on Two Problems in Connexion with Graphs," NumerischeMathematik, vol. 4, no. 1, pp. 269-271, 1959.

[2] P. E. Hart, N. J. Nilsson, B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," IEEE Trans. Syst. Science and Cybern., vol. 4, no. 2, pp. 100-107, 1968.

[3] E. Rimon, D. E. Koditschek, "Exact robot navigation using artificial potential functions," IEEE Trans. Robot. Autom., vol. 8, no. 5, pp. 501-518, 1992.

[4] A. Ghorbani, S. Shiry, A. Nodehi, "Using Genetic Algorithm for a Mobile Robot Path Planning," International Conference on Future Computer and Communication (ICFCC 2009), pp. 164-166, 3-5 April 2009.

[5] K. H. Sedighi, K. Ashenayi, T. W. Manikas, R. L. Wainwright, H. M. Tai, "Autonomous Local Path Planning for a Mobile Robot Using a Genetic Algorithm," Congress on Evolutionary Computation, CEC2004, vol. 2, pp. 1338-1345, Portland, Oregon, USA, 19-23 June 2004.

[6] Y. Zhang, L. Zhang, X. Zhang, "Mobile Robot Path Planning base on the Hybrid Genetic Algorithm in Unknown Environment," ISDA '08. Eighth International Conference on Intelligent Systems Design and Applications, vol. 2, pp. 661-665, Kaohsiung, China, 26-28 November 2008.

[7] Z. Yao, L. Ma, "A Static Environment-Based Path Planning Method by Using Genetic Algorithm," 2010 International Conference on Computing, Control and Industrial Engineering (CCIE), vol. 2, pp. 405-407, Wuhan, China, 5-6 June 2010.

[8] A. Tuncer, M. Yildirim, "Dynamic path planning of mobile robots with improved genetic algorithm," Computers & Electrical Engineering, vol. 38, no. 6, pp. 1564-1572, 2012.

[9] J. H. Holland, "Adaptation in Natural and Artificial Systems," University of Michigan Press, Ann Arbor, Michigan, 1975.

[10] O. Hachour, "The Proposed Genetic FPGA Implementation For Path Planning of Autonomous Mobile Robot," International Journal of Circuits, Systems and Signal Processing, vol. 2, no. 2, pp. 151-167, 2008.

[11] T. W. Manikas, K. Ashenayi, R. L. Wainwright, "Genetic Algorithms for Autonomous Robot Navigation," IEEE Instrumentation & Measurement Magazine, vol. 10, no. 6, pp. 26-31, 2007.

[12] A. Tuncer, M. Yildirim, "Chromosome Coding Methods in Genetic Algorithm for Path Planning of Mobile Robots," The 26th International Symposium on Computer and Information Sciences (ISCIS 2011), pp. 377-383, London, England, 26-28 September 2011.

[13] Y. Zhao, J. Gu, "Robot Path Planning Based on Improved Genetic Algorithm," Proceeding of the IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 2515-2522, Shenzhen, China, December 2013.

[14] Q. Li, W. Zhang, Y. Yin, Z. Wang and G. Liu, "An improved genetic algorithm of optimum path planning for mobile robots," Intelligent Systems Design and Applications, ISDA '06. Sixth International Conference on, vol. 2, pp. 637-642, 2006.

★★★