

Mobile Robot Path Planning based on Improved Genetic Algorithm With A-star Heuristic Method

Yibo Li¹, Dingguang Dong¹, Xiaonan Guo²

1. College of Automation, Shenyang Aerospace University, Shenyang, China

2. Shenyang Aviation Xinxing Electromechanical Co., Ltd. Shenyang, China

465710931@qq.com, ddgwork@163.com

Corresponding Author: Yibo Li Email:465710931@qq.com

Abstract—This paper proposes an improved genetic algorithm to achieve efficient searching capabilities of path planning in complicated maps for mobile robot. The improved genetic algorithm uses the evaluation function of A-Star (A*) algorithm. Firstly, the grid environment model is constructed. The evaluation function of A* algorithm and the bending suppression operator are introduced to improve the heuristic information of the genetic algorithm, which accelerates the convergence speed during the search. Secondly, the insertion operators and deletion operators are introduced into the traditional genetic operators, meanwhile, the consistency of path planning is considered in fitness function, which calculating the fitness values of each path. Output the path with the highest fitness value as the optimal path. The simulation results show that the improved genetic algorithm has less iteration number and can get a better solution than the traditional genetic algorithm.

Keywords—A-Star; Genetic Algorithm(GA); Mobile Robot; Path planning

I. INTRODUCTION

The autonomous navigation process of mobile robot can be divided into three processes: environment perception, path planning and motion control. Path planning is a key technology for autonomous navigation of mobile robot [1]. Path planning refers to finding a non-collision path from the starting point to the target point according to some evaluation criteria (such as distance, time, energy consumption, etc.) in an environment with obstacles [2]. Usually, the path planning methods only focus on the length and the smoothness of single path planning, but ignore the importance of the path continuity in multiple plans for mobile robots. If the path nodes in the two planning processes are too far apart, the mobile robot will turn unnaturally, which will lead to a large roll of the mobile robot and a potential safety hazard [3].

According to the degree of grasping environmental information, mobile robot path planning can be divided into two types: global path planning based on environmental prior information and local path planning based on real-time sensor information. Global path planning is usually based on digital maps, the routes are selected by the surrounding road network model [4,5]. Only the global path planning is carried out, the smooth driving of the mobile robot can be realized by making the

mobile robot drive in strict accordance with the planned path. But the mobile robot's working environment is usually dynamic and it is difficult to obtain accurate environmental information. Therefore, path planning can only be divided into several sub-planning processes, that is, local path planning. Local path planning relies on the surrounding environment information detected by sensors in real time to plan a local path to make mobile robots avoid obstacles [6].

A* algorithm and Dijkstra algorithm are the most commonly used path planning methods based on graph search [7]. But A* algorithm and Dijkstra algorithm are both based on raster octave search, there are many path turning points and the result of path smoothness is poor, which is not conducive to the practical application of robot [8,9]. The planning methods by curve interpolation are based on sampling near the reference path, then using polynomial curve or other forms of connect the sampling points, generating candidate paths, and set up the evaluation function to evaluate the candidate path, choose the path with minimum cost and no crash [10,11]. Among the swarm intelligence optimization planning methods, genetic algorithms, particle swarm optimization, and ant colony optimization are representative. Genetic algorithm has high efficiency in solving combinatorial optimization problems and can find the optimal solution in a reasonable time. According to the paper [12], in this paper genetic algorithm used to optimize the path length and the turning angle at the same time, the path optimized was shorter and smoother. But this algorithm can only achieve global path smoothing, and can't guarantee local path coherence.

In this paper, an effective version of genetic algorithm is achieved. It utilizes the evaluation function of A* algorithm to improve the heuristic information of genetic algorithm, which accelerates the convergence speed during the search. Then the insertion operators and deletion operators are introduced into the traditional genetic operators, meanwhile, the consistency of path planning is considered in fitness function, which calculating the fitness values of each paths. Output the path with the highest fitness value as the optimal path, which solve the problem of inconsistency and unevenness in the path planning, improving the path planning ability of mobile robot.

II. ENVIRONMENT MODEL

The work environment is built by using the grid model, which divides the robot working space into $N \times N$ squares. As shown in Figure 1, the black grids are represented as obstacles, the white grids are represented as free grid squares.

In order to identify obstacles, the white grid cell is represented by 0 and the gray grid unit is represented by 1. The grid method is simple and effective to create and maintain grid model. The raster model was placed in two-dimensional coordinate system, In the grid mapping of $N \times N$, the starting grid named as Start and the target grid named as Goal, The position coordinates x , y corresponding to any grid whose grid number is R as follow:

$$\begin{cases} x = \begin{cases} \text{mod}(R, N) - 0.5 & \text{mod}(R, N) \neq 0 \\ N + \text{mod}(R, N) - 0.5 & \text{mod}(R, N) = 0 \end{cases} \\ y = N + 0.5 - \text{ceil}(\frac{R}{N}) \end{cases} \quad (1)$$

Where mod is the surplus operation, ceil rounds the elements to the nearest integers toward infinity.

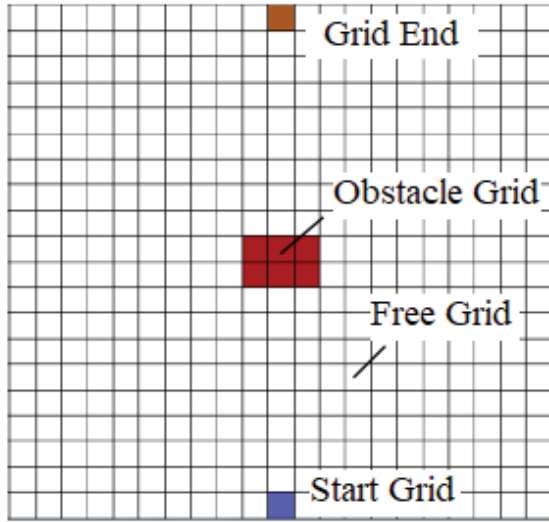


Fig. 1. Environment model

III. PATH PLANNING ALGORITHM

Genetic algorithm is an optimization algorithm based on biological evolution mechanism, which is usually divided into such steps as generating initial population, using genetic operators for genetic operation, calculating fitness value of individuals, and iterating to obtain the optimal individuals. In order to decrease the path length of the mobile robot move and improve the stability of the mobile robot, this paper introduces the delete operator and

the insert operator. On the basis of conventional genetic algorithm, using the value evaluation method in A* algorithm to choose the first population, it can keep the population abundantly.

A. Chromosome coding

In this paper, each path is represented as a chromosome, which is described by grid number. It has less counts and convenient for operation in this way. Arrange the grid number of the path passed, so

$$P = \{p_1, p_2, p_3 \dots p_n\}$$

can be represented as a path.

B. Population Initialization

The quality of genetic algorithm's primary population determines the quality of the result. In this paper, A* algorithm is used as the heuristic information of path searching to improve the convergence speed of the algorithm and obtain the better path.

The heuristic cost of A* algorithm is expressed by the estimated function [13], and the estimated function equation $f(n)$ is as follows:

$$\begin{cases} f(n) = g(n) + h(n) \\ h(n) = ((n_x - g_x)^2 + (n_y - g_y)^2)^{1/2} \\ g(n) = ((n_x - s_x)^2 + (n_y - s_y)^2)^{1/2} \end{cases} \quad (2)$$

where $g(n)$ is the minimum cost from the source node to the current node. $h(n)$ is the minimum cost from the current node to the destination node. n_x and n_y are the coordinates of the current node n , g_x and g_y are the coordinates of the target node g , s_x and s_y are the coordinates of the initial nodes.

C. Genetic operator

Genetic algorithm is mainly accomplished by a series of operations of various operators. Because of the path generation is random, it is possible that the path length is not the shortest, and there are too many unsmooth paths, which are not conducive to genetic operation. Therefore, a new operator is inserted to reconstruct the path and a new deletion operator is added to delete the redundant path nodes.

1) Selection operator

The tradition roulette selection operator used widely, but it is easy to increase the iteration numbers. In this paper, which choose the roulette selection operator with rank, is different from the tradition method on the calculation of selection probability [14]. The roulette selection operator with rank ranks the fitness value of the individual in order of size $x_1 \geq x_2 \geq \dots \geq x_n$, then calculates the selection probability according to the linear function as follows:

$$p_i = \frac{1}{n} \left(a - \frac{bi}{n+1} \right) \quad (3)$$

Where i is the rank number; n is the population numbers of individuals; a, b are constants, usually $a=1.1$. With the selection probability, the parent body is selected according to the roulette selection operator for genetic operation.

2) Crossover operator

In genetic algorithm, crossover operation is the basic mechanism of gene rearrangement, which is applied after selection operation. When the genetic operation works, the choice of the intersection is random, so the single-point crossover or the multi-point crossover can be used.

In this paper, single-point crossover method will be used. The specific crossover operation is to find all the same points in the two paths, and then select one of them to operate randomly, finally cross the following paths.

3) Mutation operator

The mutation method, which selects two grids in the path except the starting and ending points randomly, remove the path between the two grids, and then take the two grids as adjacent points, using the second step in the initialization path to operate the two points continuously. It is possible that a continuous path cannot be generated, so you need to reselect the two grids to perform the above operation until the mutation operation is complete.

In this operator, it needs to determine a mutation probability, pm , created a random number between 0~1, and compare it with the mutation probability. If the random number less than pm , the mutation operation will be carried out.

Random mutation was chosen in traditional genetic algorithms, choose a new path node instead of the original path node randomly. In this way, it creates the path with bad quality, so in this paper, the methods which was proposed in the paper [15] was used. Choosing the eight grids around the mutation node randomly with no obstacle instead of the original path node.

4) Insert operator

The insertion operator fills the gap in the path with the surrounding free grid to make it a feasible continuous path. According to equation (4), it can judge whether the two adjacent path points are discontinuous.

$$\Delta = \max \{ \text{abs}(x_{i+1} - x_i), \text{abs}(y_{i+1} - y_i) \} \quad (4)$$

x_i, y_i and x_{i+1}, y_{i+1} are the coordinates of the current path node p_i and p_{i+1} . If $\Delta=1$, p_i and p_{i+1} are continuous, otherwise, they are discontinuous. If they are discontinuous, use the averaging method to fill the path [16]. function equation is as follows:

$$\begin{cases} \dot{x}_{i+1} = (x_{i+1} + x_i) / 2 \\ \dot{y}_{i+1} = (y_{i+1} + y_i) / 2 \\ n_i = \dot{x}_{i+1} + N * \dot{y}_{i+1} \end{cases} \quad (5)$$

Where \dot{x}_{i+1} and \dot{y}_{i+1} are the coordinates of the alternate grid, N is the number of columns in the grid model. If n_i is an obstacle grid, the nearest grid of n_i used as an alternate.

5) Deletion operator

In order to shorten the length of each path and the calculation time of the algorithm, a new operator named deletion is proposed in this paper. The main idea is to start from the end point and traverse the path nodes in turn. If the current path nodes can be connected with the starting point without obstacles, the intermediate nodes between them are redundant. Remove these redundant nodes and recalculate the fitness of the new path.

D. Fitness Function

The fitness function can effectively evaluate the pros and cons of each path, which is usually proportional to the fitness value, and it has important influence on the convergence and stability of the genetic algorithm.

The purpose of this paper is to make the mobile robot avoid obstacles smoothly, reach the destination, and travel the shortest path. Therefore, the length of the path and the coherence degree of the path are optimized. The fitness function is designed as follows:

$$\text{fit}(p) = \text{Inf} - w_1 * f_1(p) - w_2 * f_2(p) \quad (6)$$

Where $f_1(p)$ is the function of path length, $f_2(p)$ is the path continuity function, w_1 and w_2 are the weights of $f_1(p), f_2(p)$ respectively. Inf is a number big enough. Path length function as follows:

$$f_1(p) = \sum_{i=0}^{L-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (7)$$

Where L is the number of the path nodes, x_i and y_i are the coordinates of the path node i . The closer the path is to the robot, the closer it will be to the position of the robot at the next moment. Therefore, the first path in the grid diagram is taken as the path the robot is about to travel.

Path coherence function as follows:

$$f_2(p) = |\theta_k - \theta_{k-1}| \quad (8)$$

Where θ_k represents the angle between the orientation of the mobile robot and the planned path at the current moment, θ_{k-1} represents the angle between the orientation of the mobile robot and the planned path at the last moment.

In this paper, the mobile robot was abstracted into a particle, can be expressed as the desired corner as follows:

$$\theta = \arctan\left(\frac{x_1 - x_s}{y_1 - y_s}\right) \quad (9)$$

Where (x_1, y_1) is the coordinate of first path node, (x_s, y_s) is the starting coordinate.

IV. RESULT

In order to verify the effectiveness of the improved ant colony algorithm, this paper uses MATLAB 2017 software to simulate. It is more convincing to use comparative method to carry out experiments under the same experimental conditions.

In this paper, comparing three different scenes which include the path length and path smooth, used A*

algorithm, genetic algorithm (GA), and improve genetic algorithm with deletion operator(D-IGA).Scene 1 has regular shape of obstacles and simple environment. Scene 2 has irregular shape and complex environment of obstacles. Scene 3 has "u-shaped" obstacles. Comprehensive consideration of algorithm running speed and accuracy of raster environment modeling, set the size of the grid diagram as 20*20 or 25*25, the edge length of each grid is equivalent to the actual scene distance of 0.2m.

The simulation results are shown in figure 2, figure 3 and figure 4.

Comparative Analysis of Path Planning Algorithms

Scene 1

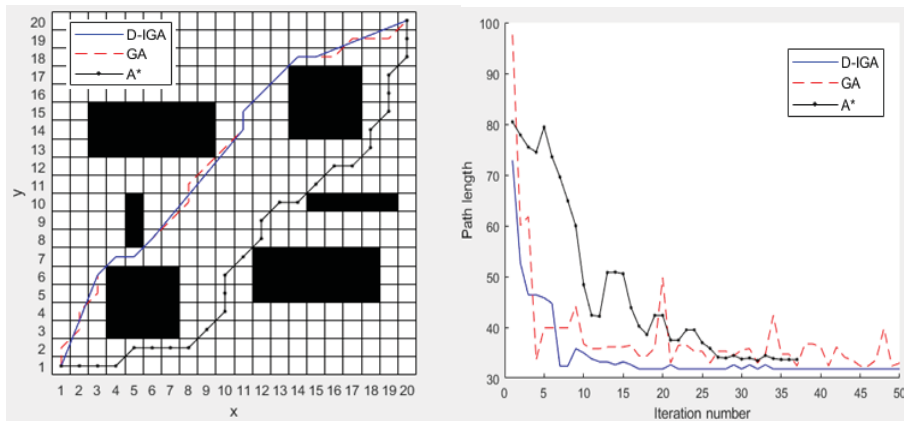


Fig. 2. Fig.2. The test results of three algorithms run on scene 1

Scene 2

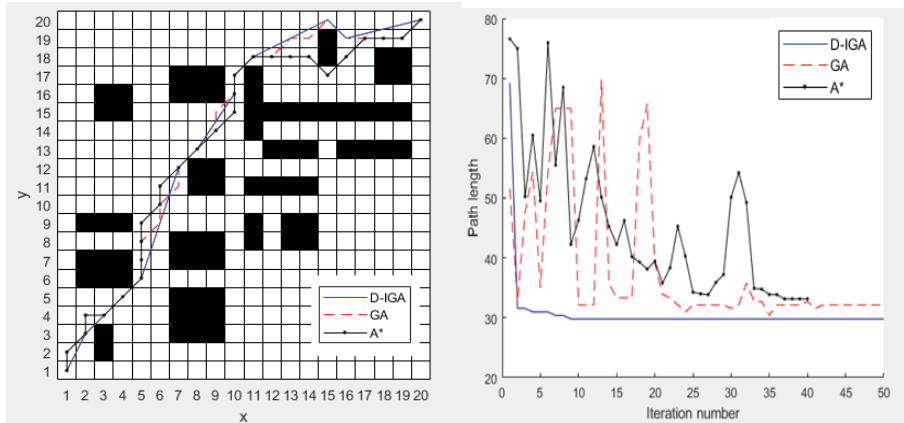


Fig. 3. Fig.3. The test results of three algorithms run on scene 2

Scene 3

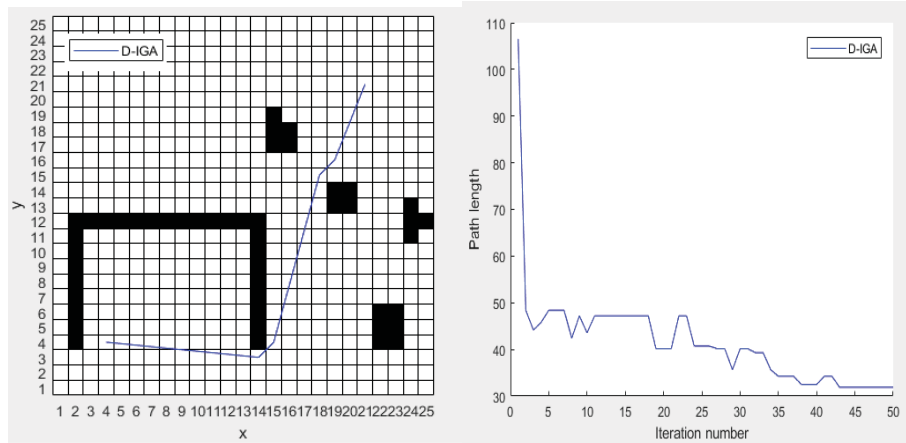


Fig. 4. Fig.4. The test results of three algorithms run on scene 3

TABLE I. TEST RESULTS FOR THREE ALGORITHMS UNDER DIFFERENT MAPS

| Map | Algorithm | Optimal solution of the algorithm | The average of the shortest distance | Average iteration time | Number of bends |
|---------|-----------|-----------------------------------|--------------------------------------|------------------------|-----------------|
| Scene 1 | A* | 33.5782 | 33.9742 | 37 | 16 |
| | GA | 32.7519 | 32.8741 | 50 | 15 |
| | D-IGA | 31.5451 | 32.0472 | 20 | 7 |
| Scene 2 | A* | 33.8746 | 33.9751 | 40 | 18 |
| | GA | 32.6758 | 33.4749 | 42 | 15 |
| | D-IGA | 30.4513 | 30.8514 | 10 | 8 |
| Scene 3 | A* | - | - | - | - |
| | GA | - | - | - | - |
| | D-IGA | 31.2348 | 32.0415 | 44 | 4 |

In the test, The parameter of D-IGA {PM,T,CP} is set as {500, 50, 0.8}. Where PM is the number of initial population; T is the iteration number, Cp is the crossover probability.

The simulation shows that under the three different complexity of the scene, the path has more turning point and redundant points which calculated by A* or GA, it is not convenient to control the mobile robot to finish the work. But the path is more smooth and shorter which calculated by D-IGA, even in the scene with “U-obstacles”, it means the improve genetic algorithm can give a better plan, achieving the global optimal. In the “U-obstacles”, A* or GA unable to reach the target grid.

V. CONCLUSION

In this paper, on the basis of genetic algorithm to make improvements, the A* algorithm as well as the heuristic function, add delete operator redundant path points out grid map, solves the conventional path planning algorithm in the grid map the path length of the shortest and the problem of turning point is more coherent and smooth to get the optimal path.

The experimental results show that: using the improved algorithm in this paper for path planning, the sum of the actual driving trajectory length, turning times and the absolute value of turning angles of the

mobile robot are all smaller than the conventional algorithm, which significantly improves the driving efficiency and stability of the mobile robot.

ACKNOWLEDGMENT

This work was supported in part by the Transformation of Scientific and Technological Achievements of Shenyang (Grant no. Z17-5-032), and the name of the work is The Transformation of Technological Achievements for Automatic Agricultural Machinery.

Proceedings of International Conference on Intelligent Systems Design & Applications. IEEE,2006

- [16] T Wei, C Long, "Path Planning for Mobile Robot Based on Improved Genetic Algorithm", Journal of Beijing university of aeronautics and astronautics, pp.1-14,2020 (in Chinese)

REFERENCES

- [1] Mac T THOA, C COPOT and D.T TRAN, "Heuristic approaches in robot path planning: A survey", Robotics and Autonomous Systems, vol.86, pp.13-14, 2019
- [2] X ZHAO, Z WANG and C.K HUANG, "Mobile robot path planning based on improved A* algorithm", Robot, vol.40(06), pp.137-144, 2018 (in Chinese).
- [3] H.Z ZHOU, X.M HU and L CEHN, "Dynamic path planning obstacle avoidance algorithm for autonomous driving", Computer application, vol.37(3), pp.883-888, 2017 (in Chinese).
- [4] B PADEN, M CAP and S.Z YONG, "A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles", in Proceedings of IEEE Transactions on Intelligent Vehicles, 2016, pp.33-55.
- [5] C ALIA, T GILLES and T REINE, "Local trajectory planning and tracking of autonomous vehicles, using clothoid tentacles method", IEEE International Symposium on Intelligent Vehicles (IV), 2015
- [6] K CHU, M LEE and M SUNWOO, "Local Path Planning for Off-Road Autonomous Driving With Avoidance of Static Obstacles", in Proceedings of IEEE Transactions on Intelligent Transportation Systems, 2012, pp.1599-1616.
- [7] J WANG, X ZHANG and B CHEN, "Heuristic optimization path-finding algorithm based on Dijkstra algorithm", Journal of University of Science & Technology Beijing, vol.3(3), pp.346-350. 2007
- [8] Q WANG, W.H HUANG and B LIU, "An improved A* algorithm for Path-Planning of Two-Wheeled Self-balancing Vehicle", in Proceedings of IEEE Conference on Industrial Electronics and Applications (ICIEA), 2018, pp.841-846.
- [9] Y XIN, H.W LIANG and M.BDU, "An improved A algorithm for searching infinite neighborhood", Robot, vol.36(5), pp.627-633. 2014 (in Chinese).
- [10] W XU, J WEI and J.M DOLAN, "A Real-Time Motion Planner with Trajectory Optimization for Autonomous Vehicles", in Proceedings of IEEE International Conference on Robotics and Automation. 2012
- [11] T GU, J SNIDER and J.M DOLAN "Focused Trajectory Planning for autonomous on-road driving", in Proceedings of IEEE Intelligent Vehicles Symposium, 2013
- [12] M WU, E CHEN and Q SHI, "Path planning of mobile robot based on improved genetic algorithm". in Proceedings of Chinese Automation Congress. IEEE ,2018
- [13] X.L Dai, S Long, Z.W Zhang and D.W Gong, "Mobile Robot Path Planning Based on Ant Colony Algorithm With A* Heuristic Method", Frontiers in neurorobotics, vol.13, 2019
- [14] C.S Shen "Commonly used selection operator in genetic algorithm is implemented in MATLAB", vol.(03), pp.199-202. 2003 (in Chinese)
- [15] Q Li, W ZHANG and Y YIN, "An Improved Genetic Algorithm of Optimum Path Planning for Mobile Robots", in