ISSN: 2414-1895 DOI: 10.6919/ICJE.202106_7(6).0050

Research on Path Planning of Mobile Robot based on Particle Algorithm

Qiang Huang

Logistics Engineering College, Shanghai Maritime University, Shanghai 201306, China.

Abstract

Particle Swarm Optimization (PSO) is an evolutionary computing technology, similar to genetic algorithm, and also an optimization tool based on iteration. The system is initialized into a group of random solutions, and the particle searches in the solution space following the optimal particle through iteration. At present, it has been widely used in function optimization, neural network training, fuzzy system control and other applications of genetic algorithms. In this paper the static problem of mobile robot path planning are studied, the grid method is used to establish the global path planning for mobile robot, on the basis of the simulation environment, respectively random standard particle swarm optimization (pso) algorithm and particle swarm optimization (pso) algorithm was used for path planning of mobile robot programming simulation, and has made the comparison and analysis to the simulation results of the two. Through the comparison of the results, it is proved that the proposed method is better than other global path planning methods in terms of convergence speed and dynamic convergence characteristics.

Keywords

Mobile Robot; Particle Swarm Optimization; Path Planning; Grid Method.

1. Introduction

Nowadays, robots are playing a more and more important role in our life. Our life style and living standard have a great relationship with robots. Among them, the automated mobile robot has also made a lot of important research results. The path planning problem is a key technology of mobile robot, and various studies on the path planning have become a hot spot. Particle swarm optimization (pso) algorithm is a reference cluster foraging behavior of birds, birds are looking for food, because food source location is unknown, so is dispersed work independently at the beginning, when one of the members to find food, will convey the information to company, then the company will according to the specific information in their respective food source to search the best path, in the end by the birds to find the best path for food. Due to its good convergence speed, no need to set too many parameters, and simple framework, particle swarm optimization algorithm is favored by many scholars[1]. At present, PSO algorithm has been widely used in many fields such as machine learning and neural network optimization.

2. Mathematical model of particle swarm optimization

Let's assume that we're searching for particles in an n-dimensional space, and let's call the number of particle clusters m. Then the set of particle swarm is: $Swarm = \{x_1^k, x_2^k, x_3^k \cdots x_m^k\}$. The number of iterations is denoted by k, and then imagine a search in an n-dimensional space. After iteration k, the point where a particle is located is represented by an n-dimensional vector $x_i^k = \left(x_{i1}^k, x_{i2}^k, x_{i3}^k \cdots x_{in}^k\right)$, $i = 1,2,3,4,\cdots,m$. This n-dimensional vector is an implicit optimal solution. The n-dimensional

ISSN: 2414-1895 DOI: 10.6919/ICJE.202106_7(6).0050

vector can also represent the velocity of i, namely $v_i^k = (v_{i1}^k, v_{i2}^k, v_{i3}^k, \cdots, v_{in}^k)$, $i = 1,2,3,4,\cdots,m$. If the number of iterations stays at particle i, then this resting point is the optimal point, which is called the individual extreme value and is denoized by $p_{best}^k = (p_{i1}^k, p_{i2}^k, p_{i3}^k, \cdots, p_{in}^k)$, $i = 1,2,3,4,\cdots,m$. If this solution is the best of all solutions, we call it the global extremum, denoted by $g_{best}^k = (p_{g1}^k, p_{g2}^k, p_{g3}^k, \cdots, p_{gn}^k)$. Then the particle from k+1 iteration to generation i in n-dimensional space can be expressed as follows:

$$v_{in}^{k+1} = \omega * v_{in}^k + c_1 r_1 (p_{in}^k - x_{in}^k) + c_2 r_2 (p_{gn}^k - x_{gn}^k) \tag{1}$$

$$x_{in}^{k+1} = x_{in}^k + v_{in}^{k+1} \tag{2}$$

$$\omega = \omega_{max} - iter * \frac{\omega_{max} - \omega_{min}}{iter_{max}} \tag{3}$$

The superposition form of each generation of particles is shown in Figure 1:

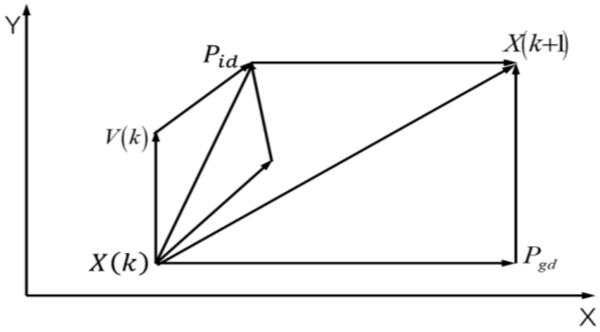


Figure 1. Iterative form of each generation of particle swarm

It should be noted that when iterating[2], the new iteration point should have the following constraints:

$$|v_{id}^{k+1}| \le V_{max} \tag{4}$$

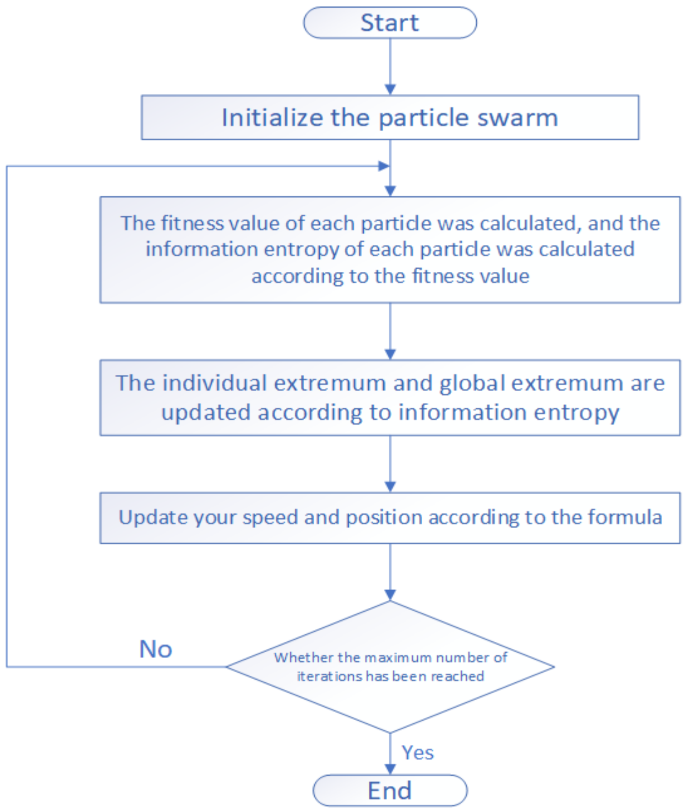


Figure 2. Flowchart of particle swarm optimization algorithm

ISSN: 2414-1895
DOI: 10.6919/ICJE.202106_7(6).0050

3. The basic process of particle swarm optimization

The basic operation steps of particle swarm optimization algorithm are as follows:

Step 1: For population initialization, a random value of initial rate and initial point is given, and then parameters are set as follows: population size m, momentum weight ω, learning factor c_1, c_2, and the maximum number of iterations $iter_{max}$;

Step 2: Determining the exact fitness value of a single particle according to the fitness function.

Step 3: Set individual extremum p_i. First, the fitness value of the current particle is calculated, and then compared with the previous one, and the optimal value is selected to replace the previous optimal value[3];

Step 4: Set the global extremum p_g. The fitness values of all particles were obtained, and then the optimal value was selected as the global extreme value by comparing the values of all particles;

Step 5: Determine the velocity and location of each particle;

Step 6: Determine if the final constraints are met. If it does, it stops iterating to the optimal value; otherwise, it goes to step 2 and starts again.

4. Parameter analysis of particle swarm optimization algorithm

The main parameters of particle swarm optimization are population size m, learning factor c_1, c_2, inertia weight ω, maximum speed V_{max} and number of iterations G_k. The choice of group size depends on the specific problem, but it is generally set at 20~40, and here it is set at 20. Some scholars pointed out that the sum of c_1 and c_2 the learning factors had better be close to 4. Here to pick up $c_1 = c_2 = 2$. Set to ω decrease from 0.9 to 0.4 as the number of iterations increases. Since the spatial model established by raster method has a very low resolution, it cannot be set V_{max} too large in selection[4]. Here is $V_{max} = 10$. The number of iterations G_k determines the termination condition of the iterative optimization calculation. Considering that the optimization of the raster method model is not complicated, it is set to G_k 400 here.

5. Environmental modeling

In this paper, the raster map method is used to build the robot moving environment. The size of the raster particle size determines the accuracy of the raster map[5]. Generally, the smaller the value, the more accurate it is.

5.1 Determination of raster granularity

The density of obstacles is the key to the selection of grid granularity, and the size of the robot is also a factor. Have a map of the overall framework, analyses the obstacles of all area is the first step, and then by the edge of a starting point, the convex polygon obstacle to triangles as the unit division, circular or irregular shape with rectangular partition for the uni[6]t, according to the obstacles in the area of the whole area of ratios to determine the grid size, specific steps are as follows:

Step 1: Randomly identify an obstacle;

Step 2: Distinguish its shape;

Step 3: The shape is a polygon. Look for an edge point and divide it into different number of independent triangles;

Step 4: For other shapes, find out all the vertices x_{max} ,y_{max} ,x_{min} .y_{min} , draw a rectangle $(x_{min,}\ y_{min})$, $(x_{max,}\ y_{max})$ with diagonal points, then take any vertex of the rectangle as the initial point, and divide it into two independent triangles;

Step 5: The area of all triangles is determined according to the calculation formula of triangle area $s = \frac{1}{2} * a * b * sin\,\alpha$;

ISSN: 2414-1895 DOI: 10.6919/ICJE.202106_7(6).0050

Step 6: Find out if there are obstacles that have been ignored. If there are obstacles, go to Step1 and start again;

Step 7: According to the formula $s_{ab} = \sum_{l \in \Omega} s_l$, the total area of the obstacle is obtained;

Step 8: Raster granularity can be obtained according to the following formula:

$$l = \begin{cases} l_{grid} = \frac{s_{ab}}{s_{total}} l_{max}, & if\ l_{grid} > l_{min} \\ l_{min}, & else \end{cases} \tag{5}$$

Step 9: Run terminates.

The flow chart for calculating raster granularity is as follows:

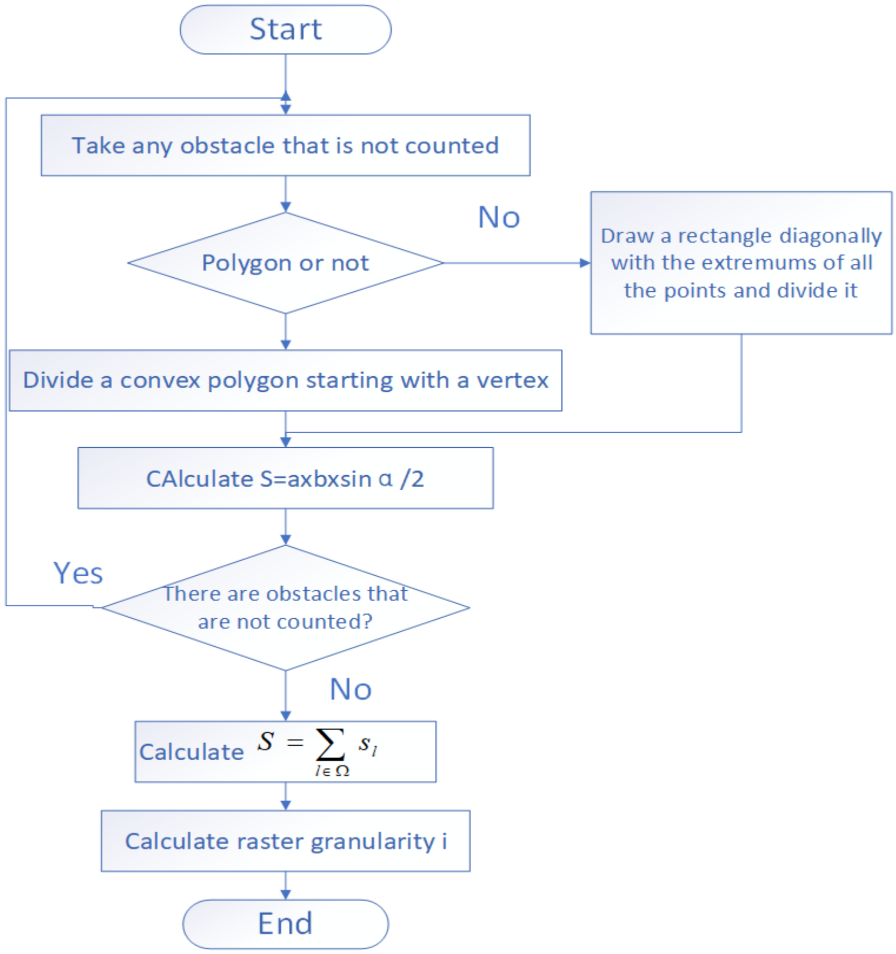


Figure 3. Calculate the raster granularity flow chart

5.2 Discretization of space and treatment of obstacle boundary

When building the model, the point where the robot is currently located is connected to the target point. This line is taken as the abscissa, and the abscissa axis rotates 90^0 as the ordinate. The grid granularity is divided, and then the space is discretized[7]. Then there is the disposal of obstacles, as follows:

(1) Integrate, that is, if the grid is not full of grid, the whole grid is counted, as shown in Figure 4.

(2) Ntegrative obstacles, that is, if there is a concave obstacle, it is regarded as a complete obstacle. As shown in Figure 5.

(3) If the robot can pass through the obstacles, the number of obstacles will remain the same. If the robot cannot pass through the obstacles, an additional obstacle will be added, as shown in Figure 6.

 Volume 7 Issue 6, 2021
ISSN: 2414-1895 DOI: 10.6919/ICJE.202106_7(6).0050

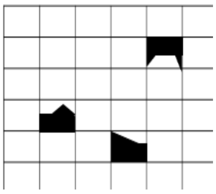


Figure 4. If there is not a grid, it counts as a grid

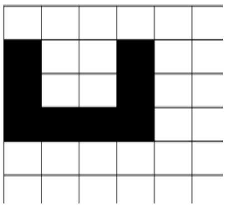


Figure 5. When the obstacle is hollow and concave, the hollow is also regarded as an obstacle

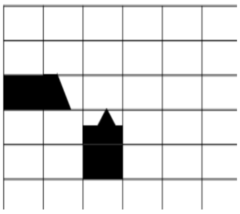


Figure 6. An adjacent obstacle less than the width of the robot is considered an obstacle

5.3 Environmental model based on particle swarm optimization algorithm

In this paper, a 20x20 raster map is established, and 26 obstacles are set as the initial points, and the coordinate is, E is the target point, and the coordinate is, the robot is regarded as a particle, and the obstacles are disposed according to the method mentioned in the previous section. The established environmental model is shown in Figure 7.

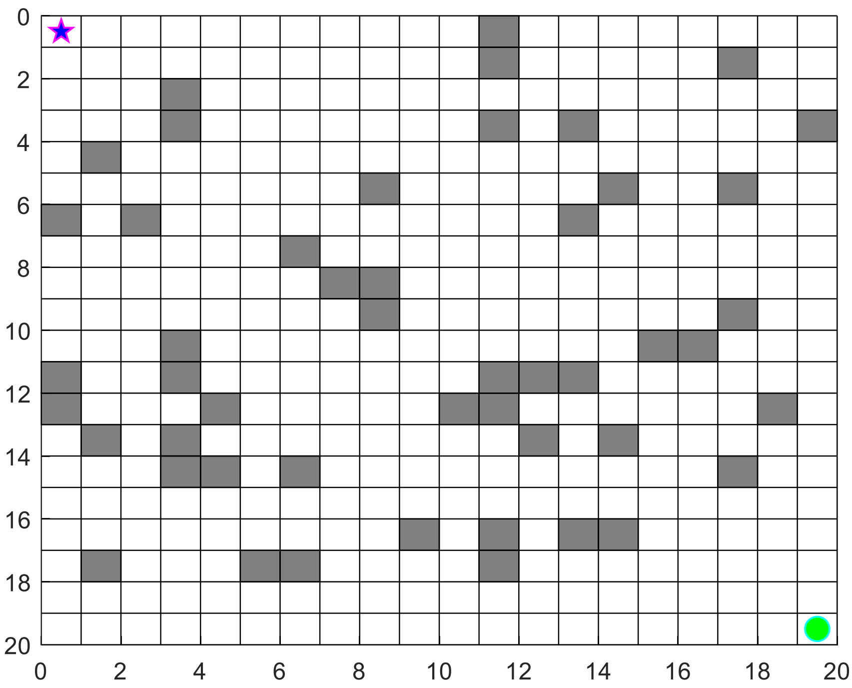


Figure 7. Environmental model based on particle swarm optimization algorithm

ISSN: 2414-1895
DOI: 10.6919/ICJE.202106_7(6).0050

6. The fitness function and particle fitness value were established

6.1 Determination of raster granularity

Any particle in the particle swarm can plan an effective trajectory. The fitness function is the ruler to measure the merits of the particle, and the fitness value determines the movement trend of the inner part of the population, so the establishment of the fitness function is the key[8]. Usually, the objective function can be used, because it can be evolved into fitness function, and the form of evolution is not fixed. The general evolution method is as follows:

$$Fit\big(f(x)\big) = f(x) \tag{6}$$

$$Fit\big(f(x)\big) = \frac{1}{f(x)} \tag{7}$$

Of course, according to different situations, the establishment of fitness function will be different. Path distance, stability and smoothness are used as metrics. In the map, the robot's path has two forms, straight or diagonal motion. The formula of path distance is as follows:

$$f_1 = \sum_{i=2}^{n} \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \tag{8}$$

The path security function is:

$$f_2 = \sum_{i=1}^{m} C \tag{9}$$

The path smoothness function is:

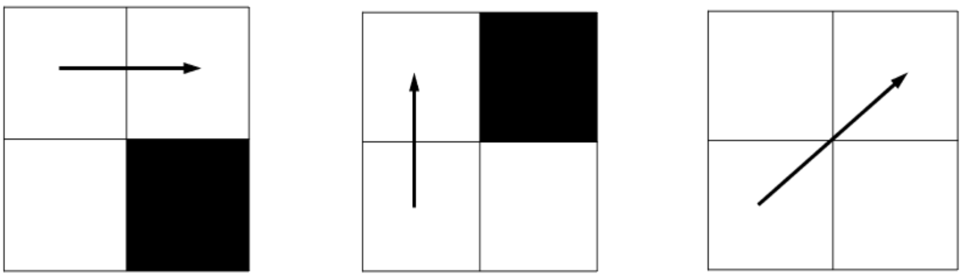
$$f_3 = \frac{n_1}{4} + \frac{n_2}{2} \tag{10}$$

In summary, the fitness function of mobile robot can be established as:

$$Fitness(pass) = \alpha \cdot f_1 + \beta \cdot f_2 + \gamma \cdot f_3 \tag{11}$$

6.2 Calculation of particle fitness value

No less than one path can be established between adjacent effective particles. In the design of path planning in this paper, the distance is mainly the distance, and the direction of motion has two kinds: straight line and diagonal line. The smaller the distance of motion, the better. When walking along a straight line, the distance is increased by 1 for each further line; when walking along a diagonal line, the distance is also increased by 1. Through comparative analysis, the distance moving along a diagonal line is smaller than that of a straight line[9]. Therefore, this study selects the way of moving along a diagonal line, as shown in Figure 8.



(a) Across the walk (b) Vertical to walk (c) Walk along the diagonal

Figure 8. Robot movement mode

Firstly, the adaptive value of the row i and row $i+1$ of particle is assumed to be $fit(i, i+1)$, namely, $Fitness$ is the adaptive value of particle, and the formula is as follows:

$$Fitness = \sum_{i=0}^{dim-1} fit(i, i+1) \tag{12}$$

$$fit(i, i+1) = \begin{cases} abs\big(x(i+1) - x(i)\big) & if\ x(i+1) \neq x(i) \\ 1 & if\ x(i+1) = x(i) \end{cases} \tag{13}$$

ISSN: 2414-1895
DOI: 10.6919/ICJE.202106_7(6).0050

7. Simulation and Analysis

In this paper, the grid method is used to build the environment map, and the simulation of mobile robot path planning based on particle swarm optimization algorithm is realized. The simulation environment is: Intel(R) Core(TM) I5-4210U CPU @ 1.70GHz (4 CPUs), ~2.4GHz, 8GB RAM, MATLAB R2018B. The robot moves from the pentacle in the upper left corner of the map to the dot position in the lower right corner. The size of the search space is set as 20x20. The coordinate of the initial point is $D(0,0)$, and the coordinate of the target point is $E(20,20)$. The detailed parameters are set as follows: the maximum number of iterations is 100, the population size is 100, the population size is 20, the cognitive coefficient is $c_1 = 0.5$, the social learning coefficient is $c_2 = 0.7$, the inertia coefficient is $\omega = 0.96$, Path planning simulation is as follows:

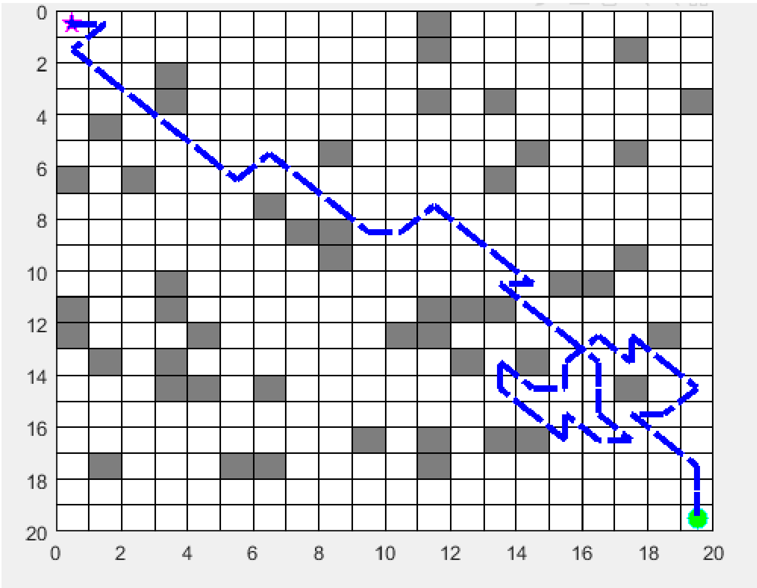


Figure 9. Particle Swarm Optimization Random Roadmap

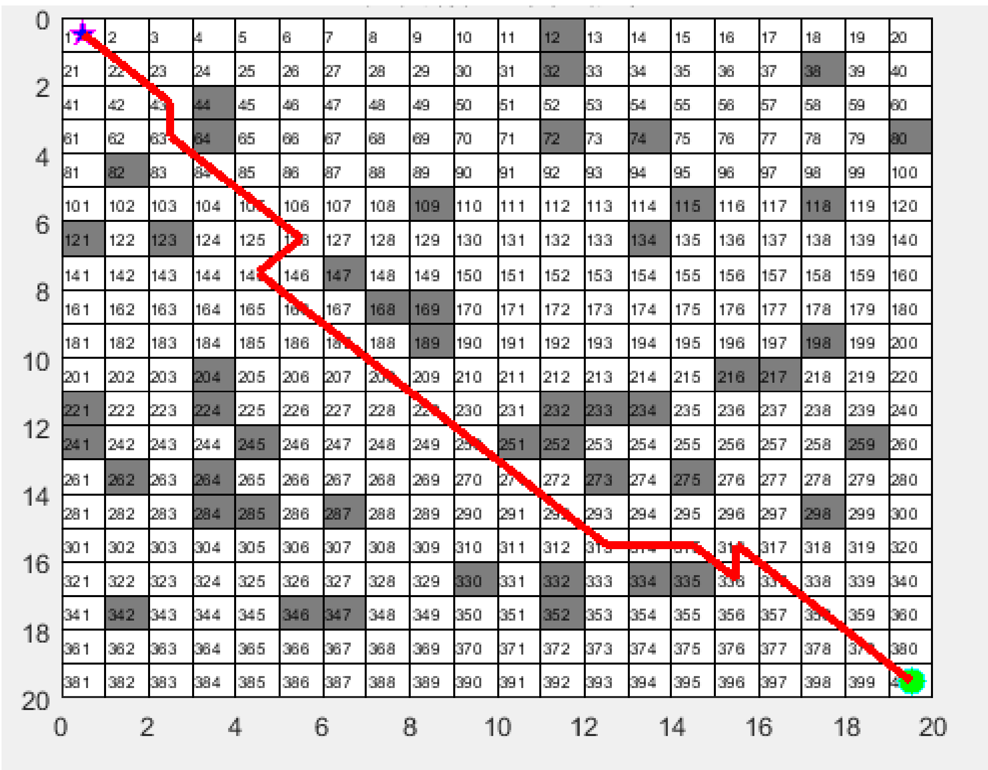


Figure 10. Particle Swarm Optimization Roadmap

ISSN: 2414-1895
DOI: 10.6919/ICJE.202106_7(6).0050

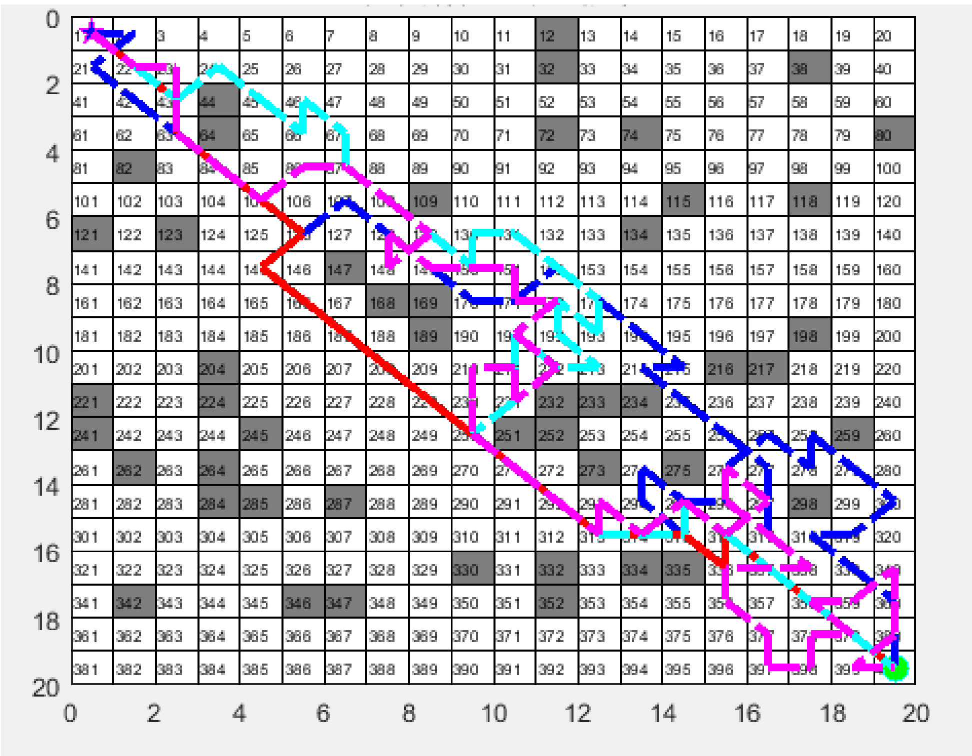


Figure 11. Particle Swarm Optimization Algorithm Comparison Roadmap

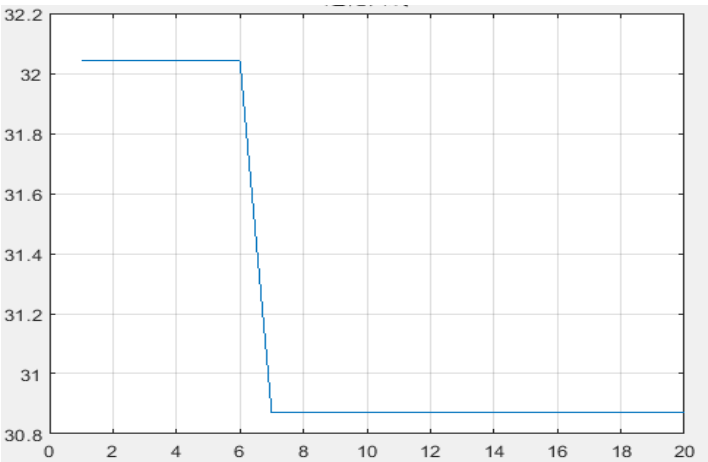


Figure 12. Evolutionary graph

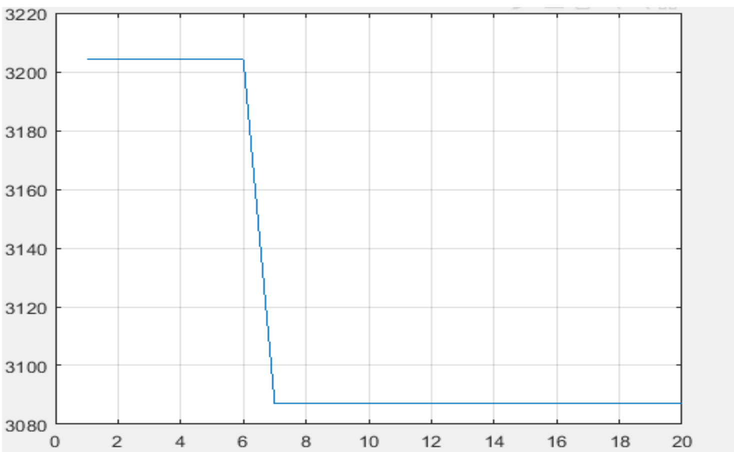


Figure 13. The changing trend of the best individual fitness value

ISSN: 2414-1895 DOI: 10.6919/ICJE.202106_7(6).0050

According to the analysis, the optimal algorithm path is (1, 22, 23, 43, 63, 84, 83, 104, 125, 146, 167, 188, 208, 229, 250, 271, 291, 312, 313, 314, 315, 336, 357, 358, 379, 400). The mobile robot needs to make four 45-degree turns. Figure 6-4 shows the searching process of the shortest path of time spent by the random particle swarm optimization algorithm. As can be seen from the figure, the particle and swarm algorithm found the optimal path after about 15 iterations, and the fitness function value representing the optimal path was 31.2132. As can be seen from Figure 9, the road map obtained by adopting the fitness function without introducing path smoothness is obviously inferior to the optimal path fitness value obtained by introducing path smoothness.

8. Conclusion

This paper mainly studies the path planning of mobile robot based on PSO algorithm. First of all, this paper introduces the mathematical model of PSO algorithm, and the algorithm of computing the basic flow, and its parameters were analyzed, including the momentum weights ω, learning factors c_1 and c_2, and then using the grid method is used for modeling environment, including the choice of grid particles and spatial discretization and deal with the obstacle boundaries as well as to the grid map are identified, the simulation model of out of the need of environment, analysis of effective particles in the grid map, set up according to the fitness function and adaptive value function to calculate the particle and finally, the route of random simulation based on PSO algorithm, respectively, and the optimal route simulation, and has carried on the contrast analysis, Finally, it is found that the optimal route obtained by smoothness is introduced.

References

[1] Eberhart R C, Kennedy J. A new optimizer using particle swarm theory [C]. Proceedings of the sixth international symposium on micro machine and human science, 1995, 1: 39-43.

[2] Sooda K, Nair T R G. A Comparative Analysis for Determining the Optimal Path using PSO and GA[J]. International Journal of Computer Applications, 2014, 32(4):8-12.

[3] Liu Xu. Particle Swarms Optimization and Its Application Development [J]. Science, Technology and Economy Guide, 2018, 26(04):138-139.

[4] Zhang Jun, Zhan Zhihui et al. Computational Intelligence [M]. Beijing: Tsinghua University Press, 2011:53-73.

[5] SUN Jun. Research on Quantum Behavior Particle Swarms Optimization Algorithm [D]. Jiangsu: Jiangnan University, 2009.

[6] Zhang Dingxue, Guan Zhihong, Liu Xinzhi. An adaptive particle swarm optimization machine with dynamic change of inertia weight, control and decision making, 2008,23(11):27-32.

[7] Ma Huawei, Zhu Yimin, Hu Xiaoxuan. A Particle Swarm Optimization Algorithm for Unmanned Aerial Vehicle Cooperative Task Planning [J]. Systems Engineering and Electronics, 2016,38(7):1583-1588.

[8] Wang Yujia. Research on Global Search Strategy of Multi-objective Particle Swarms Optimization Algorithm [D]. Shanghai Jiao Tong University, 2008.

[9] Ramos, P. M. and Paiva, C. R...All-optical pulse switching in twin-core fiber couplers with intermodal dispersion [J]. IEEE J. Quantum Electron. 1999: 983-989.