Path planning of mobile robot based on particle swarm optimization algorithm

Huimin Yang¹, Lei Jiang¹, Lijuan Wu ¹

¹Department of electronic and Information Engineering, Liaoning University of science and technology, Anshan, Liaoning,114018, China

Abstract. Aiming at the premature convergence problem of particle swarm optimization (PSO) due to the loss of population diversity in the late stage of search, this paper improves the traditional PSO and proposes the elite mean deviation induction strategy, which improves the convergence speed and accuracy of PSO. The experimental results show that the convergence speed and accuracy of the improved PSO are improved. The effectiveness of the improved algorithm is proved.

1. Introduction
Mobile robot path planning is one of the most important technologies in robotics, which marks the level of robot intelligence to a certain extent. Relevant scholars at home and abroad have tried to use various optimization methods to solve the problem of mobile robot path planning, and have made a lot of contributions to the development of robot path planning technology. Path planning of mobile robot is to optimize a collision free optimal path from the starting position to the target position in the working environment according to a certain optimization objective.[2]

Different classification index can divide the path planning of mobile robot into different categories. According to the needs of researchers, the corresponding content is often selected to find the appropriate algorithm for path solving, which is mainly divided into global path planning and local path planning. Global path planning is the state of all the information in the environment, especially the shape and position of obstacles. [8] Traditional algorithms mainly include visual graph method, artificial potential field method, element decomposition method and so on. These traditional methods have the disadvantages of large amount of calculation, low efficiency and easy to fall into local optimum.

Particle swarm optimization is an intelligent algorithm, which is very simple and efficient, strong global search ability and fast convergence speed. Many scholars began to study the application of particle swarm optimization in mobile robot path planning. However, PSO algorithm has the disadvantages of slow convergence speed and easy to fall into local extremum, and its effect in mobile robot path planning is not very ideal. Therefore, it is of great significance to study how to make PSO keep the diversity of the population in the later stage of the search and effectively escape from the local optimum, so as to ensure that the robot can reach the target point safely and efficiently. [5]

2. Heuristic cost particle swarm optimization algorithm
Particle swarm optimization (PSO) is a stochastic global optimization technique, which finds the global optimal solution through cooperation and information sharing among individuals. Suppose that the
problem is solved in d-dimensional search space, the population is composed of M particles [9], and the population can be expressed as 
\[ S = x_1^{(k)}, x_2^{(k)}, ..., x_m^{(k)} \] 
Where \( k \) is the number of iterations. In the \( k \)-th iteration, the position of the i-th particle is expressed as d-vector 
\[ x_i^{(k)} = (x_1^{(k)}, x_2^{(k)}, ..., x_d^{(k)}) \] 
among \( i = 1, 2, ..., m \), represents the position of each particle I in the search space, and also represents a solution. The velocity of each particle can be expressed as 
\[ v_i^{(k)} = (v_1^{(k)}, v_2^{(k)}, ..., v_d^{(k)}) \] 
represents the velocity of each particle in each dimension of the population. The fitness value is expressed as 
\[ fitness = f(X_i) \] 
In each iteration, according to the particle's own vector, velocity vector, and each particle's history information and population [4] information to constantly update their own state. In the PSO algorithm, once the iteration starts, the position and velocity of all particles will be updated recursively:

\[ v_i^{(k+1)} = \omega v_i^{(k)} + c_1 \phi_1 (p_i^{(k)} - x_i^{(k)}) + c_2 \phi_2 (p_{gd}^{(k)} - x_i^{(k)}) \]  \( \text{(1)} \)

\[ x_i^{(k+1)} = x_i^{(k)} + v_i^{(k+1)} \] \( \text{(2)} \)

In the type, \( \omega \) is the inertia weight factor; \( c_1, c_2 \) are the acceleration coefficient; \( \phi_1, \phi_2 \) are two random numbers of uniform distribution; \( v_i^{(k)} \) is the velocity of particle I in d-Dimension; \( x_i^{(k)} \) is the position of particle I in D Dimension; \( p_i^{(k)} \) is the best position that the particle experiences in the D dimension; \( p_{gd}^{(k)} \) is the best position in the current population.

The standard PSO algorithm will lose the diversity of the population in the later stage of the search, resulting in the algorithm can not converge to the local extremum, It leads to premature convergence and slow convergence. Aiming at this defect of PSO algorithm, this paper proposes an elite mean deviation induction strategy. The elite individual particles are initialized to increase the diversity of the population, improve the speed update model of particle swarm optimization, and improve the convergence speed of the algorithm.

The elite mean deviation is used to characterize the diversity of particle distribution in the population by observing the dispersion degree of fitness function distribution. Let the kth generation population of particle swarm s consist of particles \( x_1(t), x_2(t), ..., x_s(t) \) its fitness is \( fitness_1(t), fitness_2(t), ..., fitness_s(t) \). The average fitness of individual population

\[ fitness = \frac{1}{s} \sum_{i=1}^{s} fitness_i(t) \] 

is the fitness of the optimal individual is denoted as \( fitness_{max}(t) \) [7] individuals whose fitness is less than the average fitness are called elite individuals, the average of all elite individuals is recorded as \( \bar{f}_{all}(t) \), the mean deviation of elite is \( \Delta \tau(t) = \bar{f}_{all}(t) - fitness_{max}(t) \) When the population diversity is lost, the elite retaining strategy is used to induce the population. For the elite individuals whose fitness is better than the average fitness, in addition to retaining the global optimal individuals, all other sub optimal elite individuals are initialized according to the probability P. As the particles are initialized, and the position of the particles after initialization is random, the population diversity is increased.

Secondly, the particles in the region with poor fitness are induced to leave this region and continue to search in other regions that may contain global optimal solution. Here, the particles with fitness greater than average fitness are regarded as poor particles, it needs to be induced. The induction rules are as follows:
\[ \mu_{id}(t+1) = \begin{cases} \eta \cdot k_1, & \text{fitness}(x_i(t)) \geq \text{fitness}(x_i(t-1)) \\ \eta \cdot k_2, & \text{fitness}(x_i(t)) < \text{fitness}(x_i(t-1)) \end{cases} \] (3)

In the type, \( \mu \) is the inducing factor; \( \eta \) is a random number, \( \eta \in (0,1) \), \( k_1 \) is a positive number less than 1, \( k_2 \) is a positive number greater than 1. The update formula of particle velocity is as follows:

\[ v_{id}^{(k+1)} = \alpha v_{id}^{(k)} + c_1 k_1 (P_{id}^{(k)} - x_{id}^{(k)}) + c_2 k_2 (P_{gd}^{(k)} - x_{id}^{(k)}) + c_3 k_3 (P_{wd}^{(k)} - x_{id}^{(k)}) \] (4)

\( P_{wd}^{(k)} \) is the location of particles with poor fitness.

3. Simulation results and analysis

In order to verify the effectiveness of improved algorithm, eight commonly used standard test functions are selected to test its performance, and the improved algorithm and PSO algorithm are tested on these eight benchmark functions. The experimental times of each algorithm on the same test function are 100. This paper mainly compares the Best, Mean, Worst and Iter of each algorithm.

| Test function | Dimension | Range    | Extremum |
|---------------|-----------|----------|----------|
| \( f_1(X) = \sum_{j=1}^{d} x_j^2 \) | 30        | [-100,100] | 0        |
| \( f_2(X) = 10d + \sum_{j=1}^{d} [x_j^2 - 10\cos(2\pi x_j)] \) | 30        | [-5.12,5.12] | 0        |
| \( f_3(X) = \sum_{j=1}^{d} [100(x_{j+1} - x_j)^2 + (x_j - 1)^2] \) | 30        | [-30,30] | 0        |
| \( f_4(X) = \sum_{i=1}^{d} (\sum_{j=1}^{i} x_j)^2 \) | 30        | [-100,100] | 0        |
| \( f_5(X) = 0.5 - \frac{\sin(\sqrt{x_1^2 + x_2^2}) - 0.5}{(1 + 0.01(x_1^2 + x_2^2))^2} \) | 30        | [-2.048,2.048] | 0        |
| \( f_6(x) = \sum_{i=1}^{d} x_i \sin(\sqrt{x_i}) \) | 30        | [-500,500] | -12659.4 |
| \( f_7 = \frac{1}{4000} \sum_{i=1}^{d} x_i^2 - \prod_{i=1}^{d} \cos(\sqrt{x_i}) + 1 \) | 30        | [-600,600] | 0        |
| \( f_8 = \sum_{i=1}^{d} (x_i - (i - 0.5d))^2 \) | 30        | [-100,100] | 0        |

Table 2. test results of improved algorithm and PSO algorithm on test function.

| Test function | \( f_1 \) | \( f_2 \) | \( f_3 \) | \( f_4 \) | \( f_5 \) | \( f_6 \) | \( f_7 \) | \( f_8 \) |
|---------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| BPSO          | 0.793e-52 | -0.80432  | 0         | 96.7218   | 0         | 0         | 0.24779   |
| Mean          | 6.005e-51 | -0.00431  | 2.3739    | 20.7321   | 89.3741   | 0         | 0         | 1.107e+02 |
| Worst         | 5.743e-51 | -0.00327  | 19.4758   | 69.8432   | 90.7539   | 0         | 0         | 3.0789    |
| Iter          | 498       | 298       | 539       | 150       | 100       | 140       | 151       | 101       |
| BPSO          | 0         | -0.85456  | 0         | 96.8345   | 0         | 0         | 0.23815   |
| Mean          | 0.037481  | 21.2178   | 151.3749  | 0         | 0         | 0         | 0.7932e+02 |
| Worst         | 0         | -0.39543  | 75.3728   | 237.4123  | 0         | 0         | 1.37833   |
| Iter          | 291       | 317       | 405       | 78        | 231       | 24        | 21        | 59        |
It can be seen from table 2 that improved algorithm has achieved high convergence accuracy in the test function, followed by the convergence accuracy of improved algorithm, and the number of times to obtain the global optimal solution is also the largest. From the test results of iteration times, improved algorithm has faster convergence speed than PSO algorithm; The results of improved and PSO are the same. In general, the improved PSO algorithm in this paper is better than PSO algorithm in both convergence progress and convergence speed.

![Convergence comparison](image1)

![Comparison of icpso fitness values](image2)

![Comparison of PSO fitness values](image3)

Figure 1. The fitness curve of the improved algorithm and PSO algorithm on the test function.

Figure 1 shows the convergence curves of the two algorithms on the test function. The convergence curves of the two algorithms are obtained through 200 experiments on the test function. It can be proved that the improved algorithm has faster convergence speed.

4. Conclusion

According to the loss of population diversity in the late iteration of particle swarm optimization, the algorithm converges prematurely, easily falls into local extremum, and reduces the convergence speed. The elite mean deviation is used to increase the diversity of particle swarm and avoid falling into local extremum. The particles with poor fitness are induced to accelerate the convergence speed of
particle swarm optimization, and the particle velocity update model is improved. The improved algorithm is tested in some test functions, and the simulation results show that the algorithm has high search speed and precision.

References
[1] Eman A. Mustafa, Faehaa A. Al-Mashhadane*, Ghada A. Taqa. (2020). Turkish J. Journal of International Dental and Medical Research, 13(3), 957-962.
[2] Zheng, W., Cheng, S., & Li, Z. (2019). Soft-sensing modeling and intelligent optimal control strategy for distillation yield rate of atmospheric distillation oil refining process. Chinese Journal of Chemical Engineering, 027(005), 1113-1124.
[3] Zhong Peisi, Li tantan, Liu Mei, Chen xiulong, & Zhang Xinglan. (2019). Robot path planning method based on improved genetic algorithm and Morphin algorithm, 47(24), 33-38.
[4] Cao Yi, Guo Mengshi, & Liu Haizhou. (2018). Obstacle avoidance spatial path planning of serial manipulator based on improved RRT algorithm, v.46; No.468(18), 75-81.
[5] Lafhaj, Z., Goueygou, M., Djerbi, A., & Kaczmarek, M. (2006). Correlation between porosity, permeability and ultrasonic parameters of mortar with variable water/cement ratio and water content. Cement & Concrete Research, 36(4), 625-633.
[6] Xu Xiaohui, & Zhang an. (2006). Image segmentation with optimal entropy threshold based on particle swarm optimization. Computer engineering and applications, 42(10), 8-11.
[7] W Zhang. "EULAR evidence based recommendations for gout. Part I: Diagnosis. Report of a task force of the standing committee for international clinical studies including therapeutics (ESCISIT)." (2006).
[8] Baek, W., & Bommareddy, S. (1995). Optimal m-ary data fusion with distributed sensors. IEEE Transactions on Aerospace & Electronic Systems, 31(3), 1150-1152.
[9] Harralson, H.H., Teulings, H.L., & Miller, L.S. (2018). Poster Temporal Spatial Differences Online Offline Signatures.
[10] Leverault, C.M., McCormick, M.F., Lajara, R.J., Lam, A.W., Ta, P., & Stolz, H. W., et al. (1997). Compact housing for a computer workstation.