Path Planning of Mobile Robots Based on Improved Genetic Algorithm

Zhang Ke-qi
School of Mechatronics Engineering and Automation, Shanghai University, China

Abstract: With the development of intelligent manufacturing, whether from the consideration of capacity, efficiency, or convenience, the requirements for mobile robots are increasing, reasonable regional path planning is one of the most critical needs, and a genetic algorithm is the best way to solve this problem, but in some complex working environments, traditional genetic algorithms will cause some problems, such as the path is not smooth, the steering angle is too large, the number of turns is large, etc. In this paper, an improved genetic algorithm is utilized to optimize the path-planning problem of mobile robots to circumvent the common issues arising from other approaches. The Improved Genetic Algorithm (IGA) has emerged as a significant advancement in the field of optimization techniques. By incorporating adaptive features, this refined approach yields enhanced performance and accuracy when compared to traditional genetic algorithms. Building on the foundational principles of evolutionary computation, the IGA employs innovative strategies, such as adaptive crossover and mutation operators, to navigate complex solution spaces effectively. It can also reduce computation time and increase efficiency by considering various considerations, such as environmental constraints and avoiding obstacle.

Keywords: Intelligent Manufacturing, Path Planning, Genetic Algorithms, Mobile Robots

Introduction

With the development of automation technology, most of the manufacturing industry's traditional logistics and transportation methods are inefficient, flexible, and labour-consuming, so it cannot meet efficient production needs. In order to solve this problem, almost all platforms have introduced advanced mobile robots for assistance, but how to carry out path planning has become a new problem; in the case of fixed starting and ending points, the path planning factors that need to be considered are time, node distance, energy consumption, etc., based on this, various algorithms are used to analyze path planning problems, such algorithms have strong search ability, high efficiency, easy to implement and simplify complex

Correspondents Author:
Zhang Ke-qi, Shanghai University, China
Email : zhangkeqi1@gmail.com

environments (Cenerini et al., 2023). A genetic algorithm is one of them, which imitates the development of biological evolution mechanisms in nature, establishes corresponding evolutionary models, and selects individuals in the population, cross-mutation and a series of genetic operations so that individuals in the population evolve in a favourable direction, as shown in Figure 1. Because of its high efficiency, robustness, and parallel search, it is widely used to solve the problem of path planning of mobile robots (Li et al., 2023).

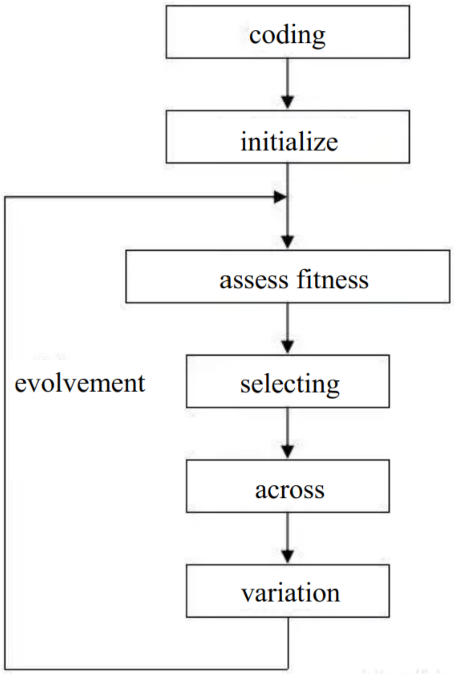

Figure 1 The population evolve in a favourable direction

Genetic algorithms have become increasingly popular in recent years due to their ability to solve complex problems in a variety of fields, from engineering to biology. These algorithms mimic the process of natural selection, where a population of candidate solutions evolves over time to find the best possible solution to a given problem. However, the traditional genetic algorithm has its limitations, and researchers have been working to improve and optimize it for better performance (Keung et al., 2023).

The key differences between traditional and improved genetic algorithms are selection methods, crossover techniques, and mutation strategies. Selection methods in improved GA are typically more sophisticated than in Traditional GA. They include tournament selection, where individuals compete against each other in a tournament to determine the best candidate for breeding. Another method is rank-based selection, where individuals are ranked based on their fitness, and the probability of selection is proportional to their rank. Crossover techniques in improved GA can also be more complex than in Traditional GA. They include methods such as uniform crossover, where each bit of the offspring is randomly selected from the corresponding bits of the parents. Another method uses multi-parental crossover, where more than two parents are involved in the breeding process. This can increase the diversity of the population and prevent premature convergence (Xie et al., 2023). Mutation strategies in improved GA can also be more sophisticated than in Traditional GA. They include methods such as adaptive mutation, where the mutation rate changes over time based on the population's fitness. Another method is a self-adaptive mutation, where the individual's fitness determines the mutation rate. When planning the mobile robot's path in the grid environment, because its working environment is divided into several small grids, the movement path of the mobile robot will also be divided into multiple segments. At this time, the path planned based on the genetic algorithm will generally have the problem of the path not being smooth, mainly because the path is composed of multiple line segments (Sun et al., 2023). From this point of view, several line segments will form several angles; the size of this angle will directly affect the length and smoothness of the path. In addition, an optimal path, that is, the shortest path must be found from the robot's current position to the target position, which is affected by the number of turns. Therefore, how to use genetic algorithms to optimize these problems is of far-reaching significance, and this paper designs new mutation operators for planning the path of mobile robots(Liu et al., 2023; Nwankwo et al., 2023).

Research Method

Improve Genetic Algorithms

First, the initial population is randomly generated, the number of individuals in the population and the gene length of the individual are taken, and the fitness value of each individual is calculated using the fitness function. To select individuals in the initialized population, you can use a roulette wheel-like method to first calculate the fitness values of all individuals in the population, then calculate the ratio of the fitness values of individuals in the population to the total population fitness values, and finally use the roulette wheel to record the number of times each individual is selected. Single-point crossing, which determines the pairing strategy when chromosome crossing, such as pairing the first half of the population with the second half of the population, and crossover when the crossover conditions are met. Mutate an individual in the population, generate a random number, compare it with the mutation probability, if it is less than the mutation probability, use the random number method to determine the individual's mutation position, and mutate the gene at this position. Calculate the fitness values for each individual in the population and rank the individuals by the size of the fitness values. Determine whether the maximum number of iterations is reached, and output the optimal individual if the conditions are met.

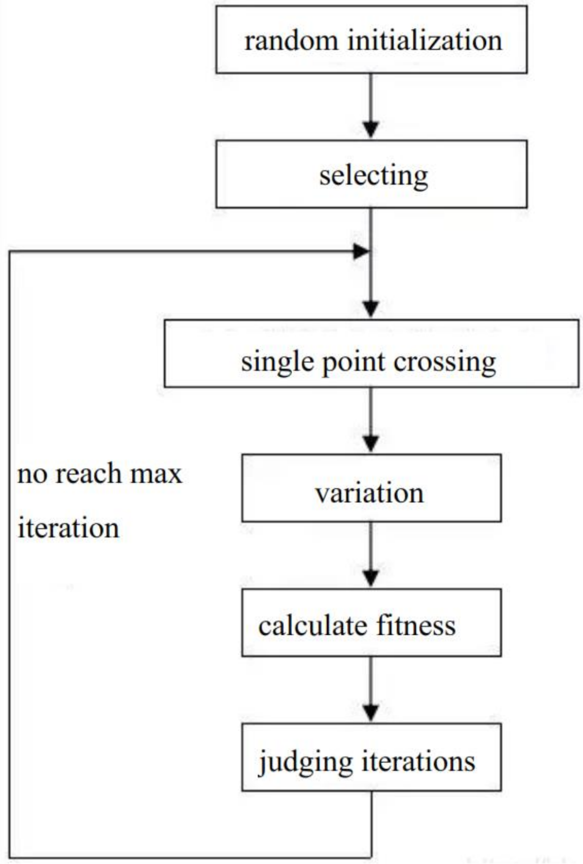

Figure 1 Improved Genetic Algorithm Steps

Operator Design

Stratified Approach

In order to select more excellent individuals so that excellent genes can be continued, it is necessary to adopt an efficient selection strategy (Petrović et al., 2022). The traditional selection strategy is to adapt individuals, generally referring to the top 10% of individuals with fitness directly copied to the next generation because this method only retains a small number of individuals with high fitness (Yu et al., 2023; Zhao & Cheah, 2023). Then as the number of iterations increases, the diversity of the population will decrease rapidly. It is now easy to get the local optimal solution, rather than the global one. So we can divide the selected criteria into four parts a, b, c, d based on the fitness function, and the fitness of the individual in the 4 levels satisfies Eq 1 :

$$Fa > Fb > Fc > Fd \tag{1}$$

Among them, the three levels of A, B, and C are directly copied to the next generation so as to ensure the diversity of population genes, so theoretically, the number of individuals in layer A is less than layer B, and layer B is less C layer, so after each iteration to update the population, first divide the population into three levels of A, B, and C according to the layering method, and then select individuals to copy to the next generation according to the roulette method (Luperto et al., 2023; Zhang et al., 2022).

Fitness Function Design

In genetic algorithms, the fitness function is the criterion for judging the ability of individuals in a population to survive, and the objective function determines it. The fitness function is non-negative and does not correspond exactly to the objective function. The larger the fitness function's value in dealing with the problem, the better the effect. In the environment of mobile robots, the fitness function needs to include indicators such as path length and energy consumption. Because our goal is to make the path the shortest and at the same time make the least number of turns, we introduce the penalty coefficient a here, where the fitness function value and the objective function value present a negative correlation, the more turns, the larger the objective function value, the smaller the fitness value, the smaller the probability of being retained when selecting individuals, where the objective function is as follows Eq 2:

$$\mathrm{F = n - li = 1d(pi, pi + 1) + am(v - vt)}\pi\mathrm{2rvt} \tag{2}$$

F is the objective function, $\mathrm{d(pi, pi + 1)}$ represents the distance between the gene point pi and $\mathrm{pi + 1}$ to form a line segment, and a is the penalty coefficient, generally greater than 1 . M is the number of turns when all nodes on the path chromosome are connected, the path is straight when $m = 0$, v is the moving speed of the mobile robot, and vt is the speed of the mobile robot when turning r is the turning radius.

F is the objective function, $\mathrm{d(pi, pi + 1)}$ represents the distance between the gene point pi and $\mathrm{pi + 1}$ to form a line segment, and a is the penalty coefficient, generally greater than 1 . M is the number of turns when all nodes on the path chromosome are connected, the path is straight when $m = 0$, v is the moving speed of the mobile robot, and vt is the speed of the mobile robot when turning r is the turning radius (Shi et al., 2023).

Improved Mutation Operators

Under normal circumstances, the mutation probability will take a very small value. Suppose the mutation probability value is too large to destroy a large number of excellent individuals. In that case, if the value is too small, it will make the population converge prematurely in the optimization process. In the mutation process of both good individuals and inferior

https://doi.org/10.58291/ijec.v2i1.84

individuals, we can improve the operator through the following so that the path is always mutated in the direction of the individual(Raikwar et al., 2022; Zhang et al., 2023).

Let the path individual $d = (N_1,\ N_2,\ N_3, \ldots., N_{i-1},\ N_i, N_{i+1}, \ldots., N_m)$, when the gene location of the mutation point and the location of the previous gene point and the location of the next gene point occurs in the following three situations, delete and mutate this point in the path.

1. If the raster sequence number of the gene location of the mutation point and the grid sequence number of the previous gene point location and the grid sequence number of the next gene point location meet one of the following two formulas:

$$\begin{gathered} N_{i+1} - N_{i-1} = 10 \\ N_{i+1} - N_{i-1} = 1 \end{gathered} \tag{3}$$

Thus, $N_{i-1},\ N_i, N_{i+1}$ form an angle of 45 degrees, at which point the gene N_i to form a new path.

2. If the raster sequence number of the gene location of the mutation point and the raster sequence number of the previous gene point location and the grid number of the next gene point location meet the following formula:

$$N_{i+1} - N_{i-1} = 11 \tag{4}$$

Thus, $N_{i-1},\ N_i, N_{i+1}$ form a 90 degrees angle, at which time the gene N_i is deleted, forming new paths.

3. If the raster sequence number of the gene location of the mutation point and the grid sequence number of the previous gene point location and the grid sequence number of the next gene point location meet one of the following two formulas:

$$\begin{gathered} N_{i+1} - N_{i+1} = 12 \\ N_{i+1} - N_{i-1} = 21 \end{gathered} \tag{5}$$

Thus, $N_{i-1},\ N_i, N_{i+1}$ form a 135 degrees angle, and if the upper and lower grids of the gene position of the mutation point are free, the gene N will be used to delete it. The new path formed with this variant operator is $d = (N_1,\ N_2,\ N_3, \ldots, N_{i-1},\ N_{i+1}, \ldots., N_m)$, forming a new path with the advantages of shorter length, fewer turns and smaller corners. Therefore, the greater the mutation probability, the better, so we take the mutation probability as 1.

https://doi.org/10.58291/ijec.v2i1.84

Result and Discussion

Take a model based on traditional genetic algorithms and select the same population size and number of iterations, for example, we take the crossover probability as 0.8 and the mutation probability as 1, and the results are shown in Table 1 after MATLAB simulation. The results show a comparison of the path length, number of turns and number of three angles for the same number of iterations in the three environments

Table 1 Comparison of the Results of the Two Algorithms

Environ ement	Genetic Algorithms	The number ofiteratio ns	Path lengt h	Numb er of turns	45-degreeang les	90-degreea ngles	135-degree angles
1	ordinary	45	14.01	14.01	14.01	14.01	14.01
	improvement	45	7	7	7	7	7
2	ordinary	40	0	0	0	0	0
	improvement	40	2	2	2	2	2
3	ordinary	35	2	2	2	2	2
	improvement	35	13.38	13.38	13.38	13.38	13.38

From the results, it can be seen that when in an environment with three different iterations, if the path generated by the improved genetic algorithm described above is adopted, both the number of angles in the path, the length of the path and the number of turns is significantly optimized.

Conclusions

The above method of improving the mutation operator has achieved a series of optimization effects by selecting, crossing, mutating and other operations on the population individuals. Still, there is also much room for improvement; for example, the path turn is not entirely smooth, it is still easy to converge early, and it is easy to fall into the optimal local solution.

In the field of intelligent manufacturing, advanced mobile robots play an important role, their work efficiency directly affects the capacity of the entire factory, and the most significant factor affecting the efficiency of mobile robots is the advantages and disadvantages of genetic algorithms, so there are many optimized versions of genetic algorithms on the market at present, they optimize genetic algorithms by changing population structure, changing fitness functions, changing cross-mutation operators, and changing simulation methods, the fact is that there is no unified genetic algorithm at present, each version has its limitations, this is also the direction of future development.

https://doi.org/10.58291/ijec.v2i1.84

References

Cenerini, J., Mehrez, M. W., Han, J.-w., Jeon, S., & Melek, W. (2023). Model Predictive Path Following Control without terminal constraints for holonomic mobile robots. *Control Engineering Practice*, *132*, 105406. https://doi.org/https://doi.org/10.1016/j.conengprac.2022.105406

Keung, K. L., Lee, C. K. M., Xia, L., Liu, C., Liu, B., & Ji, P. (2023). A cyber-physical robotic mobile fulfillment system in smart manufacturing: The simulation aspect. *Robotics and Computer-Integrated Manufacturing*, *83*, 102578. https://doi.org/https://doi.org/10.1016/j.rcim.2023.102578

Li, Z., Xu, B., Wu, D., Zhao, K., Chen, S., Lu, M., & Cong, J. (2023). A YOLO-GGCNN based grasping framework for mobile robots in unknown environments. *Expert Systems with Applications*, *225*, 119993. https://doi.org/https://doi.org/10.1016/j.eswa.2023.119993

Liu, L., Wang, X., Yang, X., Liu, H., Li, J., & Wang, P. (2023). Path planning techniques for mobile robots: Review and prospect. *Expert Systems with Applications*, *227*, 120254. https://doi.org/https://doi.org/10.1016/j.eswa.2023.120254

Luperto, M., Monroy, J., Moreno, F.-A., Lunardini, F., Renoux, J., Krpic, A., Galindo, C., Ferrante, S., Basilico, N., Gonzalez-Jimenez, J., & Borghese, N. A. (2023). Seeking at-home long-term autonomy of assistive mobile robots through the integration with an IoT-based monitoring system. *Robotics and Autonomous Systems*, *161*, 104346. https://doi.org/https://doi.org/10.1016/j.robot.2022.104346

Nwankwo, L., Fritze, C., Bartsch, K., & Rueckert, E. (2023). ROMR: A ROS-based open-source mobile robot. *HardwareX*, *14*, e00426. https://doi.org/https://doi.org/10.1016/j.ohx.2023.e00426

Petrović, M., Jokić, A., Miljković, Z., & Kulesza, Z. (2022). Multi-objective scheduling of a single mobile robot based on the grey wolf optimization algorithm. *Applied Soft Computing*, *131*, 109784. https://doi.org/https://doi.org/10.1016/j.asoc.2022.109784

Raikwar, S., Fehrmann, J., & Herlitzius, T. (2022). Navigation and control development for a four-wheel-steered mobile orchard robot using model-based design. *Computers and Electronics in Agriculture*, *202*, 107410. https://doi.org/https://doi.org/10.1016/j.compag.2022.107410

Shi, K., Wu, Z., Jiang, B., & Karimi, H. R. (2023). Dynamic path planning of mobile robot based on improved simulated annealing algorithm. *Journal of the Franklin Institute*, *360*(6), 4378-4398. https://doi.org/https://doi.org/10.1016/j.jfranklin.2023.01.033

Sun, Y., Wang, X., Lin, Q., Shan, J., Jia, S., & Ye, W. (2023). A high-accuracy positioning method for mobile robotic grasping with monocular vision and long-distance deviation. *Measurement*, *215*, 112829. https://doi.org/https://doi.org/10.1016/j.measurement.2023.112829

Xie, F., Chong, Z., Liu, X.-J., Zhao, H., & Wang, J. (2023). Precise and smooth contact force control for a hybrid mobile robot used in polishing. *Robotics and Computer-Integrated Manufacturing*, *83*, 102573. https://doi.org/https://doi.org/10.1016/j.rcim.2023.102573

Yu, Z., Yuan, J., Li, Y., Yuan, C., & Deng, S. (2023). A path planning algorithm for mobile robot based on water flow potential field method and beetle antennae search algorithm. *Computers and Electrical Engineering*, *109*, 108730. https://doi.org/https://doi.org/10.1016/j.compeleceng.2023.108730

Zhang, D., Luo, R., Yin, Y.-b., & Zou, S.-l. (2023). Multi-objective path planning for mobile robot in nuclear accident environment based on improved ant colony optimization with modified A∗. *Nuclear Engineering and Technology*, *55*(5), 1838-1854. https://doi.org/https://doi.org/10.1016/j.net.2023.02.005

Zhang, Z., Xiao, J., Liu, H., & Huang, T. (2022). Base placement optimization of a mobile hybrid machining robot by stiffness analysis considering reachability and nonsingularity constraints. *Chinese Journal of Aeronautics*. https://doi.org/https://doi.org/10.1016/j.cja.2022.12.014

Zhao, X., & Cheah, C. C. (2023). BIM-based indoor mobile robot initialization for construction automation using object detection. *Automation in Construction*, *146*, 104647. https://doi.org/https://doi.org/10.1016/j.autcon.2022.104647

https://doi.org/10.58291/ijec.v2i1.84