

Research for Path Planning in Indoor Environment Based Improved Artificial Potential Field Method

Hu Pan, Chen Guo and Zhaodong Wang

1 Introduction

In the last few decades, the service mobile robots are developing rapidly, which could be able to execute different complex works. The path planning for mobile robot in indoor environment becomes a hot and rather complicated issue in the field of mobile robot research. Its task is to find a path in the narrow space with large number of obstacles, the path is optimal and collision-free [1]. The research on path planning is divided into the two directions, one is the global path planning, another is about the local path planning [2]. At present, the commonly methods are artificial potential field method [3], fuzzy logic [4], genetic algorithm [5] and so on. They can be applied into the unknown environment. The methods for the global path planning are cell decomposition [6], A* algorithm [7], D* algorithm, and others. The artificial potential field (APF) is a mature and efficient method in path planning, and it is simple mathematical calculation and high efficiency, so it is widely used. However, the local minimum point and GNRON (goals unreachable with obstacles nearby) problems are often occurred in APF [8], and the path jitter is easily to occur in narrow environment. Although there are a variety of methods can be used to solve it, such as random escape [9], heuristic search [10], walk along the wall and Tangent bug method [11] and so on, but these methods have some disadvantages and not suit for the path planning in indoor environment [15–17].

This paper puts forward a new method to apply into the path planning in indoor environment. We analysis the problems in APF and modify the function of repulsive potential field [12]. The intermediate target points are proposed to guide the robot walking out of the local minimum point [13], A* algorithm is used to get these intermediate target points and the global optimized path [14]. Several simulation experiments are carried out to demonstrate our improved APF method.

H. Pan · C. Guo (✉) · Z. Wang
School of Information Science and Technology, DMU, Dalian 116026, China
e-mail: dmuguoc@126.com

2 Modify the APF

2.1 *The Disadvantages of APF and the Improvement Measures*

When the APF method is applied to the mobile robot path planning, the following phenomena often appear:

1. If the target point is near the obstacle and located in the effective scope of the influence of obstacle. The repulsion force may be greater than the attraction force in this situation, the GNRON problem would happen.
2. If the resultant force turns to zero, and the robot has not yet reached the target point at the same time, the local minimum problem would occur and make the robot wanders or stops in a certain area.
3. If the robot walks in the narrow environment that the distance between two walls is especially short, the robot could not walk along the straight line in the narrow space and need extra times to reach the goal.

The problems in APF are analyzed in the front, and some methods are put forward to solve them. By adding the coefficient term to the function of the repulsive potential field, this could make the attractive force and the repulsive force reduce at the same time. They both reduce to zero until the robot reaches the target point, so the GNRON problem could be solved. Also the intermediate target point is introduced to guide the robot moving out of the local minima area. Last, adjust the value of the attractive force and the repulsive force, and the distribution of the potential field is modified to make sure the mobile robot can walk straightly along the wall.

2.2 *The Improved Potential Field Model*

According to the improvement measures are described above, the distance factor is added into the function of the repulsive potential field, the redefine function of it is shown as:

$$U_{rep} = \begin{cases} \frac{1}{2}\eta\left(\frac{1}{\rho} - \frac{1}{\rho_0}\right)^2 (X - X_g)^n, & \rho \leq \rho_0 \\ 0, & \rho > \rho_0 \end{cases} \quad (1)$$

where: U_{rep} is the repulsive potential field, ρ is the Euclidean distance from the robot to the obstacles, ρ_0 is the largest impact distance of simple obstacle, η is a positive scaling factor, n is a index coefficient, and it is 2 in this paper.

The modified potential field can solve the GNRON problem, but the local minimum problem did not solve. There may be a variety of kinds of local minima in

mobile robot's path planning and it often occurs. This paper mainly discusses the following three kinds of local minimum problem, the three map models that used to describe them are shown in Fig. 1.

In Fig. 1, there is a common feature that the target point and the robot are almost in a straight line and the obstacles are located between them in three maps. And there are some special points that the attractive force and the repulsive force is equal or almost equal and collinear but on the opposite direction in these circumstances. This would make the robot to be trapped in local minima and could not reach the target point.

Two methods are adopted to solve the local minima. First, we change the direction of repulsion force to solve the local minimum problem in Fig. 1a, the force analysis after changing the direction of the repulsive force is shown in Fig. 2a. In Fig. 2a, F_{rep} is divided into F_{rep1} and F_{rep2} , the direction of F_{rep1} is changed and the new direction is set as the tangent which along the scope of maximum impact with obstacles, so the angle between the repulsive force and the attractive force is no longer greater than 90° , the resultant force turns to toward the robot and the magnitude is greater than zero, the local minimum problem can be solved. The formulas in Fig. 2a are follows as:

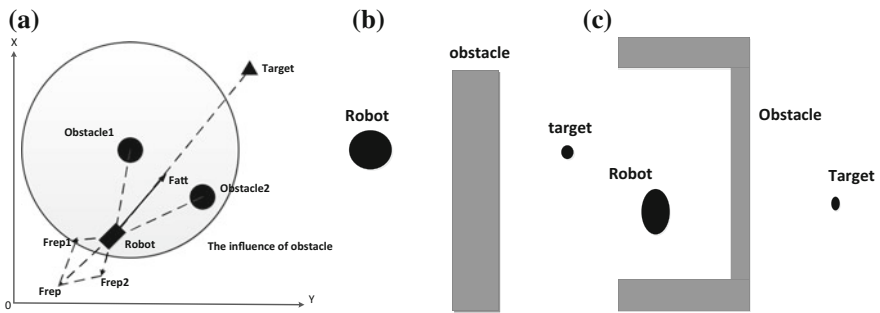


Fig. 1 Three kinds of local minimum problem

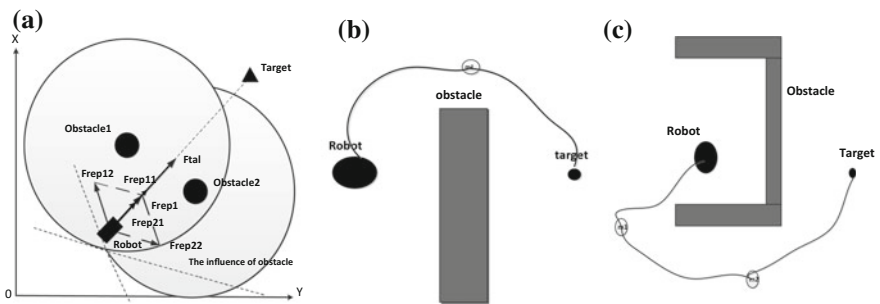


Fig. 2 Two ways of solving the three kinds of local minimum problem

$$F_{rep1} = \eta \left(\frac{1}{\rho} - \frac{1}{\rho_0} \right) \frac{1}{\rho^2} (X - X_g)^n \frac{\partial \rho}{\partial X} \quad (2)$$

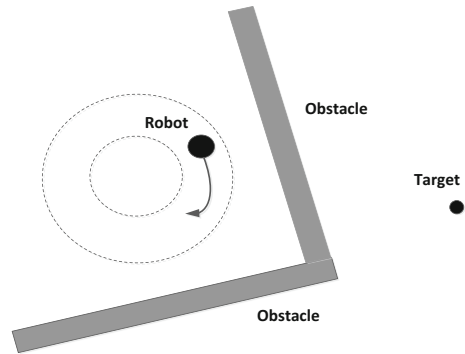
$$F_{rep2} = -\frac{n}{2} \eta \left(\frac{1}{\rho} - \frac{1}{\rho_0} \right)^2 (X - X_g)^{n-1} \frac{\partial (X - X_g)}{\partial X} \quad (3)$$

The obstacles (Fig. 1b, c) are tightly connected together and form a wall or U-shaped wall. If we just change the direction of the repulsive force as above, the robot could not bypass the wall or the U-shaped wall, so we proposed the intermediate target point based method. The additional intermediate target points are needed to be set in the special circumstances that contain U-type obstacles to guide the robot out of the local minima area. The results would be like in Fig. 2b, c by using the intermediate target point based method. This method need to detect whether the robot is located in local minimum or not firstly. The next paragraph would introduce how to detect local minima.

2.3 Detect the Local Minima

In the process of robot path planning, due to the existence of various errors, the attractive force and repulsive forces are rarely possible to the same. The so-called local stability phenomena are actually the robot moves in the range of the circle of a point and wanders in it. The kind of the so-called local stability phenomena is shown in Fig. 3.

Fig. 3 The phenomenon that the robot traps in local minima



In Fig. 3, when the robot traps into the local minimum point, the robot would wandering in the range of circle and could not get out, this is the local steady status.

In this paper, we have taken some strategies to detect whether the robot is in the local steady status or not. The specific strategies are as follows:

1. According to the robot's moving step v and the scope of the repulsive force ρ_0 , set the radius of the circle as R in the local stable area.
2. According to the speed of the robot to set the detection time as t (t is almost several times than the moving step), the program would detect whether the robot in the local steady statue every t time (t should be meet the required time that the robot could walk out of the radius of the circle).
3. Calculating the distance between every two points that the time interval is t time. If the distance is less than R , the program would turn to the corresponding algorithm to solve the local minima. Otherwise the robot is in normal state, taking the next cycle and making cycle detection.

The local minima can be detected by the three steps. The common method of setting the intermediate target point is mainly design from the view of geometric point, it needs for different analysis in different circumstances, so it is difficult to achieve the robot real-time path planning. In this paper, the A* algorithm is proposed to carry out the global path planning and the inflection points on the path can be set as the intermediate target points. The next section will describe A* algorithm in detail.

3 The Intermediate Target Point Based on A* Algorithm

A* algorithm is a typical heuristic search algorithm in artificial intelligence, compared with genetic algorithm, particle swarm algorithm, it has simple math function and small calculation, so that it is widely applied in real-time systems, artificial intelligence and so on. The core of the A* algorithm is composed of a valuation function. The valuation function is as follows:

$$f(n) = g(n) + h(n) \quad (4)$$

In formula (4), $f(n)$ is the valuation function of node n . $g(n)$ is the actual cost from the initial node to the n -node; $h(n)$ is the estimated cost of the best path from the n -node to the destination node. The function of $h(n)$ is closely related to what the actual environment is, the commonly used methods are Manhattan distance, diagonal distance and Euclidean distance, etc. Since the Euclidean distance can better represent the true distance between two points in the map, we adopt it as the calculation formula.

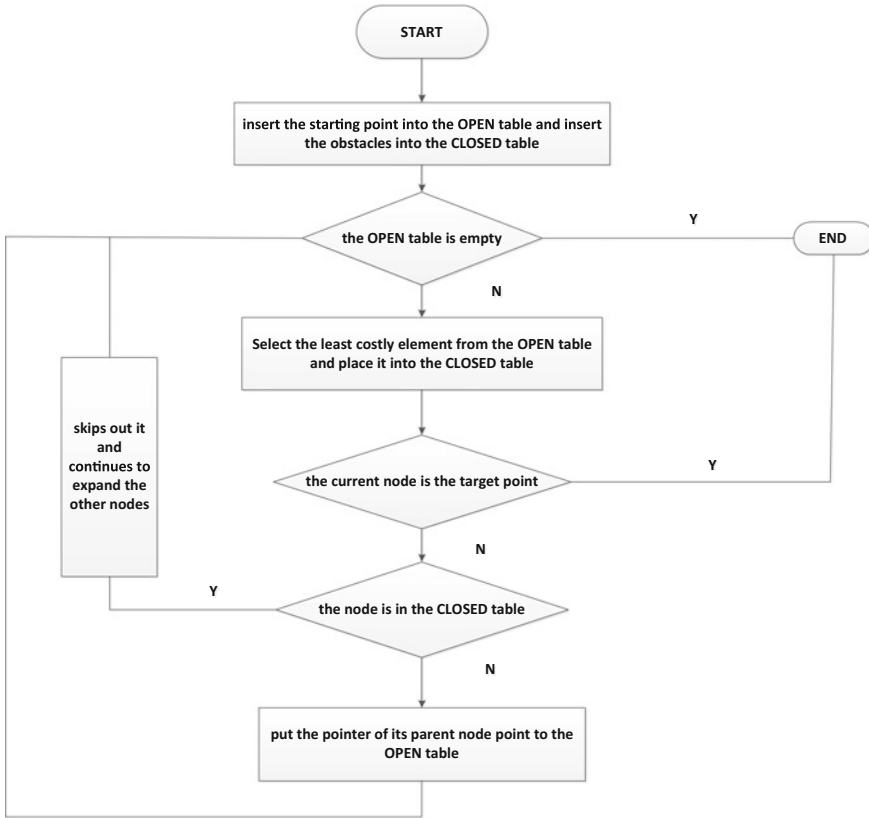


Fig. 4 The flow chart of A* algorithm

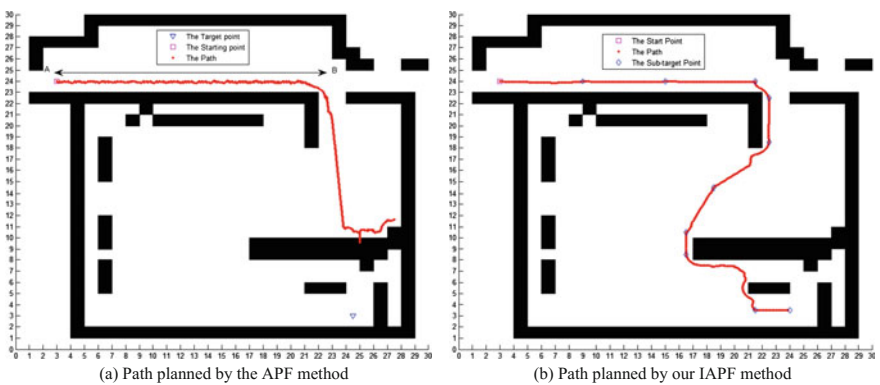


Fig. 5 The planned path in indoor environment

Table 1 The list of the intermediate target points in the global path planning

The global path planning in A* algorithm	
Starting point (3, 24)	Sub-target point 3 (16.5, 10.5)
Terminal point (24, 3.5)	Sub-target point 4 (16.5, 8.5)
Sub-target point 1 (9, 24)	Sub-target point 5 (18.5, 14.5)
Sub-target point 2 (15, 24)	Sub-target point 6 (22.5, 18.5)

After understanding the basic idea of A* algorithm, the flow chart of A* algorithm is design, it is shown in Fig. 4.

According to the actual indoor environment, we build the grid map (like Fig. 5) for the global path planning, and the path nodes are obtained by using the procedure of A* algorithm. The intermediate target points are shown in Table 1.

4 Simulation Studies

In order to prove the efficient and flexibility of our optimized IAPF algorithm, some experiments have been done in MATLAB. In these experiments, the robot is set in a certain speed, and it moves forward in the direction of the resultant virtual force, experiments are under these circumstances. The experimental environment is with 30×30 room indoor environment that includes corridor and a large laboratory. The map is built and is shown in Fig. 5, which the rectangles with different sizes are set as the obstacles in the laboratory. The coordinate of starting point and the terminal point is the same in Table 1. The number of iterations is 130. The other related parameters are shown in Table 2.

The path by the traditional APF method is shown in Fig. 5a, the robot collides with some obstacles and cannot reach the target, it would wandering at some area in some steps, and it could not bypass U-type obstacles. Additionally, the path is tortuous when the robot walks in the corridor.

In order to solve these two kinds of local minima problems, the IAPF method based on the intermediate target point is proposed. We use the A* algorithm to get the intermediate target points, the sub-target points are shown in Table 1. We also change the coefficients of attraction and repulsive force, they are turn to 25 and 8 respectively, and two sub-targets are added into the corridor. All of these measures are used to solve the local minima and ensure that the robot can walk straightly in the corridor. These improvement measures are written into the procedure of the IAPF method, and the planned path is shown in Fig. 5b. Compared with the path in

Table 2 The list of parameters in the path planning

k	η	v	ρ_0	t	R
20	5	0.2	2.7	10	2

Fig. 5a, the robot can walk along the corridor straightly and bypass the U-shaped obstacles to reach the target point. The path is smooth and collision-free.

5 Conclusions

This paper adopts a new IAPF method for the path planning in indoor environment where many kinds of obstacles are located in it. The function of the repulsive potential field is redefined and the intermediate target point based method is adopted. All of them are used to solve the common problem in APF method. Additionally, in order to make a smooth and safe path in the indoor environment, the A* algorithm is proposed and used to get the required intermediate targets. The path in simulation experiments with the IAPF method is smooth and safety.

Acknowledgements This work is supported by National Natural Science Foundation of China (Nos. 51579024, 6137114) and the Fundamental Research Funds for the Central Universities (DMU no. 3132016311).

References

1. Jiang B, Bishop AN, Anderson BDO et al (2015) Optimal path planning and sensor placement for mobile target detection. *Automatica* 60:127–139 (in Chinese)
2. Jia YH, Mei FX (2002) Simple path planning for mobile robots in the present of obstacles. *J Beijing Inst Technol* 11(2):208–211 (in Chinese)
3. Khatib O (1986) Real-time obstacle avoidance for manipulators and mobile robot. *Int J Robot Res* 5(1):90–98
4. Ye B-Q, Zhao M-F, Wang Y (2011) Research of path planning method for mobile robot based on artificial potential field. In: international conference on multimedia technology (ICMT). IEEE 2011
5. Koren Y, Borenstein J (1991) Potential field methods and their inherent limitations for mobile robot navigation. In: IEEE international conference on robotics and automation. IEEE, pp 1398–1404
6. Li G, Yamashita A, Asama H et al (2012) An efficient improved artificial potential field based regression search method for robot path planning. In: International conference on mechatronics and automation (ICMA). IEEE, pp 1227–1232
7. Ge SS, Cui YJ (2000) New potential functions for mobile robot path planning. *IEEE Trans Robot Autom* 16(5):615–620
8. Yu ZZ, Yan JH, Zhao J, Chen ZF, Zhu YH (2011) Mobile robot path planning based on improved artificial potential field method. *J Harbin Inst Technol* 43(1):50–55 (in Chinese)
9. Liu CY, Chen YQ, Liu CG (2009) Anti-collision path planning for mobile robot based on modified potential field method. *J Southeast Univ* 39(supp. 1):116–120 (in Chinese)
10. Li Q, Chen B, Wang LJ, Zhang WC (2011) An improved artificial potential field method for path planning of mobile robots. In Proceedings of the 2011 international conference on management science and intelligent control
11. Zhang JY, Zhao ZP, Liu D (2006) A path planning method for mobile robot based on artificial potential field. *J Harbin Inst Technol* 38(8):1306–1309 (in Chinese)

12. Lee MC, Park MG (2003) Artificial potential field based path planning for mobile robots using a virtual obstacle concept. In: Proceedings of the 2003 IEEE/ASME international conference on advanced intelligent mechatronics, vol 2, pp 735–740
13. Zhang MK, Li LS (2007) A method for solving local minimization problem of artificial potential field. *Comput Technol Develop* 17(5):137–139 (in Chinese)
14. Nilsson NJ (2000) Problem-solving methods in artificial intelligence. In: *Artificial intelligence: a new synthesis*. Morgan Kaufmann Publishers, Burlington
15. Li GH (2014) Distributed task allocation and path planning in dynamic environment for multi-robot guidance system, pp 103–138
16. Yu Z, Yan J, Zhao J, Chen Z, Zhu Y (2011) Mobile robot path planning based on improved artificial potential field. *J Harbin Inst Technol* 43(1):50–55 (in Chinese)
17. Guo D, Sun F, Kong T, Liu H (2016) Deep vision networks for real-time robotic grasp detection. *Int J Adv Robot Syst* 14(1)