Robotica: (2019) volume 37, pp. 1883–1903. © Cambridge University Press 2019
doi:10.1017/S026357471900033X

Multi-Robot Obstacle Avoidance Based on the Improved Artificial Potential Field and PID Adaptive Tracking Control Algorithm

Zhenhua Pan, Dongfang Li, Kun Yang and Hongbin Deng*

Department of Electromechanical Engineering, Beijing Institute of Technology, Beijing, China. E-mails: pzh-mingzhe@outlook.com, 188377985@qq.com, yk3210281001@163.com

(Accepted March 3, 2019. First published online: April 16, 2019)

SUMMARY

As for the obstacle avoidance and formation control for the multi-robot systems, this paper presents an obstacle-avoidance method based on the improved artificial potential field (IAPF) and PID adaptive tracking control algorithm. In order to analyze the dynamics and kinematics of the robot, the mathematical model of the robot is built. Then we construct the motion situational awareness map (MSAM), which can map the environment information around the robot on the MSAM. Based on the MSAM, the IAPF functions are established. We employ the rotating potential field to solve the local minima and oscillations. As for collisions between robots, we build the repulsive potential function and priority model among the robots. Afterwards, the PID adaptive tracking algorithm is utilized to multi-robot formation control. To demonstrate the validity of the proposed method, a series of simulation results confirm that the approaches proposed in this paper can successfully address the obstacle- and collision-avoidance problem while reaching formation.

KEYWORDS: Multi-robot; IAPF; Obstacle avoidance; MSAM; Formation control.

1. Introduction

Presently, multi-robot systems add more applications and challenges to the robotics field such as performing collaborative tasks in manufacturing, surveillance, space exploration and forest fire detection.[1] The obstacle avoidance and formation control are both challenging tasks, and have significant roles in the multi-robot systems. The obstacle and collision avoidance can ensure that the robots reach the targets with an optimal path without any collisions in the given complex environment. The coordination of multi-robot systems has received considerable attention in recent years.[2,3] How to make the multi-robot systems maintain the formation after avoiding the obstacles and collisions is the central issue and difficulty of multi-robot research, which is the main task of this paper.

The obstacle avoidance can ensure the robots to have abilities of automatically computing a collision-free and shortest path to the target in the working environment, which has achieved remarkable results, such as the A* algorithm,[4] grid method,[5] visibility graph method,[6] ant colony algorithm,[7] particle swarm optimization algorithm[8] and the rapidly expanding random tree search algorithm.[9,10] Artificial potential field (APF) is one of the classical approaches that are used for obstacle avoidance, which is first proposed by Khatib.[11] This method is particularly appealing because of its concise mathematical description and simplicity. However, it has some natural limitations,[12] which

* Corresponding author. E-mail: denghongbin@bit.edu.cn

includes the goals non-reachable with obstacle nearby (GNRON) and the common situations of local minima. To overcome the local minima, Li[13] presented an effective improved artificial potential field (IAPF)-based regression search method, which has achieved a good performance in path planning without local minima and oscillations. Combined with a two-layered fuzzy logic inference engine, Tsourveloudis et al.[14] proposed an electrostatic potential field (EPF) method for real-time mobile robot path planning in a dynamic environment. Rajvanshi[15] used the APF method for path planning in the complex environment, and artificial goal approach to solve the local minima. Ahmed[16] proposed the particle swarm optimization method to modify the APF method, and solved the local minima successfully. Although the potential field methods have achieved great success in obstacle avoidance of single-robot systems, the key problem is that potential field functions alone may not resolve the obstacle avoidance and conflicts among agents. Furthermore, Loizou[17] introduced a novel approach to solve the navigation problem by mapping an obstacle-cluttered environment to a trivial domain called the point world, where the navigation task is reduced to connecting the images of the initial and destination configurations by a straight line. Unfortunately, this approach requires the priori knowledge of the obstacles, which is not applicable to multi-robot obstacle avoidance.

The formation control of the multi-robot systems has been intensively studied in recent years. By applying the finite-time convergence protocols, Xiao[18] developed a new finite-time formation control framework for multi-agent systems with a large population of agents, and addressed the time-invariant formation, time-varying formation and trajectory tracking problems. For distributed flocking systems, Meng[19] designed a collection of graph-based nonlinear feedback control and Olfati-Saber[20] presented a theoretical framework for design and analysis of distributed flocking algorithms. Both of them can deal with formation maintenance and collision avoidance. With the decentralized model predictive control (MPC) and consensus-based control, Kuriki[21] proposed a cooperative formation control method with collision-avoidance capability for a multi-agent system. However, the formation control methods mentioned above did not consider obstacle avoidance. Therefore, the obstacle and collision avoidance for the multi-robot formation has achieved extensive attention from many researchers. Nagy[22] implemented a controller for the multi-agent systems by using genetic algorithm (GA) to build a potential field, but the oscillation and local minima are not under consideration. In addition, Dai[23] described a switching formation strategy for the multi-robot with velocity constraints to avoid the obstacles. In this strategy, robots can achieve a safe path by using the geometric obstacle-avoidance control method (GOACM). Unfortunately, this strategy is just applicable for the simple obstacle environment. When more robots are working in the complex environment, the robots have the risk of collision with the obstacles and local minima.

In this paper, as for the problems of risk of collisions between robots, GNRON, local minima and formation control, an obstacle-avoidance method based on the IAPF and PID adaptive tracking control algorithm is proposed for the multi-robot systems. Compared with the existing literatures on the obstacle and collision avoidance for the multi-agent systems, the main contributions of our work are summarized as follows.

(1) We designed the motion situational awareness map (MSAM) for the robots, by making grids on the MSAM. The detected information of the obstacles and other robots around the robot can be mapped on the polar coordinate grids. Furthermore, based on the MSAM, the obstacles can be modeled as a circular shape, which are easy to be handled by the APF method, and the MSAM will update in real time to ensure that the robot can make optimal decision at any time.

(2) Based on the MSAM, an adjustable repulsive function from the obstacles is designed to generalize the local minima a standard situation. Aiming at the GNRON and local minima, we put forward the rotating potential field, which is perpendicular to the attractive potential field, to ensure that the robot's resultant force is never equal to zero, and the direction of this rotating force is determined by the position of the obstacles and target. For avoiding the collisions between robots, we established a priority model for every robot and designed a repulsive function when the robots are approaching each other.

(3) The PID adaptive tracking control algorithm is designed for multi-robot formation control, which can ensure that the multi-robot system recovers its original formation after avoiding the obstacles. Moreover, the Lyapunov function is designed to verify the stability of the system.

The paper is organized as follows. In Section 2, we build the mathematical modeling for the robot, and analyze its dynamic and kinematic characteristics. Afterwards, the MSAM is designed. Section 3 concerns two conceptual solutions: the IAPF multi-robot obstacle-avoidance control method is

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

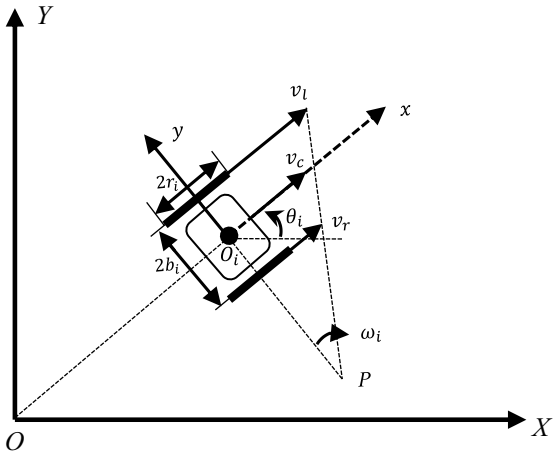

Fig. 1. The two-wheel nonholonomic mobile robot model.

designed, and the leader–follower formation model and PID adaptive tracking control algorithm are proposed for formation control. Section 4 shows the results of simulation to demonstrate the performance of the proposed approaches. In Section 5, the paper ends with the conclusions and open issues for future research.

2. Robot Mathematical Model

2.1. Kinematics model of robot with two actuated wheels

As shown in Fig. 1, the robot R_i ($i = 1, 2, \ldots, n$) achieves the motion and orientation by two drive wheels. The radius of the wheels is r_i. The distance between the wheels is denoted by $2b_i$ and O_i located in the middle of the right and left wheels. {O,X,Y} is an inertial Cartesian and {O_i,X,Y} is the local coordinate system of the robot. v_l is the velocity of left wheel, v_r is the velocity of right wheel and v_c is the velocity of the robot center. If ($v_l \neq v_r$), we can get the angular velocity w_i.

According to the robot model, the forward velocity is determined by the speed of the wheels

$$v_c = \frac{v_l + v_r}{2} \tag{1}$$

There is a constraint:

$$\omega = \frac{v_l - v_r}{2b_i} \tag{2}$$

Under the ideal situation, and in terms of the principle of rigid motion, the trajectory of the robot is a circle, and the radius can be represented as

$$R = \frac{v_c}{\omega} = \frac{b_i(v_l + v_r)}{(v_l - v_r)} \tag{3}$$

The kinematics equation of the robot can be expressed as

$$q = \begin{bmatrix} x & y & \theta \end{bmatrix}^T u \tag{4}$$

$$u = \begin{bmatrix} v_c & \omega \end{bmatrix}^T \tag{5}$$

$$\begin{bmatrix} \dot{x} & \dot{y} & \dot{\theta} \end{bmatrix}^T = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{b} & -\frac{1}{b} \end{bmatrix} \begin{bmatrix} v_l \\ v_r \end{bmatrix} \tag{6}$$

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

Equation (4) is the pose state matrix, Eq. (5) is the motion state matrix and Eq. (6) can be represented as

$$\dot{q} = S(q)V = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_c \\ \omega \end{bmatrix} \tag{7}$$

where $S(q)$ is 3×2 smooth linear independence matrix and V is the motion matrix of the robot.

2.2. *Dynamic model of robot*

The dynamic equations of the generalized coordinates are established as follows:

$$M(q)\dot{V} + V_m(q, \dot{q})V + F(\dot{q}) + G(q) = B(q)\tau \ - A^T(q)\lambda \tag{8}$$

$$M(q) = \begin{bmatrix} m & 0 & mg\sin\theta \\ 0 & m & -mg\cos\theta \\ mg\sin\theta & -mg\cos\theta & I \end{bmatrix} \tag{9}$$

$$V_m(q, \dot{q}) = \begin{bmatrix} 0 & 0 & mg\dot{\theta}\cos\theta \\ 0 & 0 & mg\dot{\theta}\sin\theta \\ 0 & 0 & 0 \end{bmatrix} \tag{10}$$

$$B(q) = \frac{1}{r}\begin{bmatrix} \cos\theta & \cos\theta \\ \sin\theta & \sin\theta \\ b & -b \end{bmatrix} \tag{11}$$

where $F(\dot{q})$ is surface friction matrix, $G(q)$ is gravity matrix, $A^T(q)$ is constraint matrix, $M(q)$ is positive inertia matrix, $V_m(q, \dot{q})$ is centrifugal and the Coriolis matrix, $B(q)$ is input transformation matrix, τ is driving force of friction, b is the coefficient of rotational viscosity and λ is the binding vector.

Differentiating Eq. (8) yields

$$\ddot{q} = \dot{S}(q)V + S(q)\dot{V} \tag{12}$$

Substituting Eqs. (7) and (12) into (8), we obtain

$$M(q)\left[\dot{S}(q)V + S(q)\dot{V}\right] + V_m(q, \dot{q})S(q)V + F(\dot{q}) + G(q) = B(q)\tau - A^T(q)\lambda \tag{13}$$

Multiply both sides of Eq. (13) by $S^T(q)$, and consider $F(\dot{q}) + G(q)$ as a disturbance item η

$$\begin{aligned} &\left[S^T(q)M(q)S(q)\right]\dot{V} + \left[S^T(q)M(q)\dot{S}(q) + S^T(q)V_m(q, \dot{q})S(q)\right]V + S^T(q)\eta \\ &\quad = S^T(q)B(q)\tau - S^T(q)A^T(q)\lambda \end{aligned} \tag{14}$$

Ignore the unknown binding λ

$$\bar{M}(q)\dot{V} + \bar{V}_m(q, \dot{q})V + \bar{\eta} = \bar{B}(q)\tau \tag{15}$$

where

$$\bar{M}(q) = S^T(q)M(q)S(q)$$

$$\bar{V}_m(q, \dot{q}) = S^T(q)M(q)\dot{S}(q) + S^T(q)V_m(q, \dot{q})S(q)$$

$$\bar{B}(q) = S^T(q)B(q)$$

$$\bar{\eta} = S^T(q)\eta$$

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

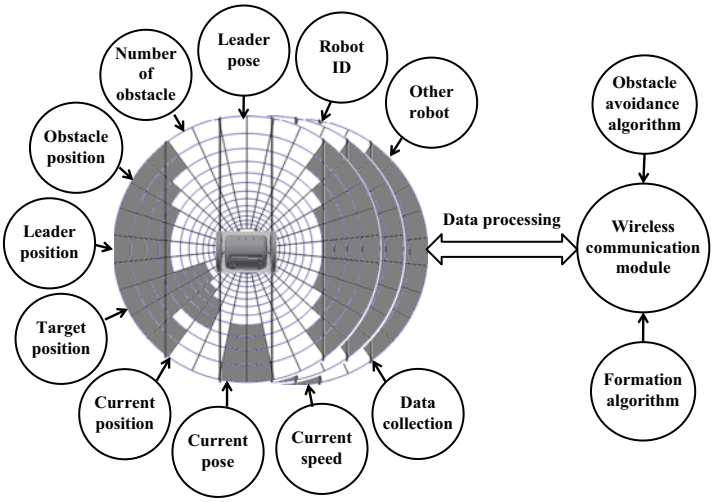

Fig. 2. The MSAM block diagram of the robot.

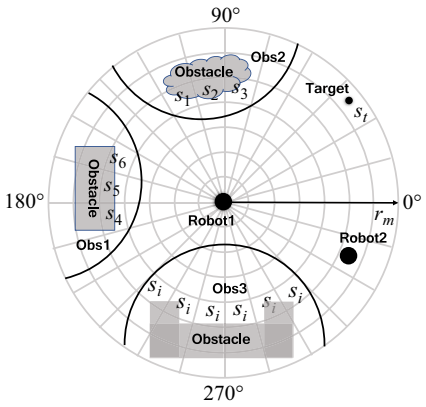

Fig. 3. The motion situational awareness map.

The dynamic mechanical equation of the robot can be simplified as

$$\begin{bmatrix} m & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \dot{V} \\ \dot{\omega} \end{bmatrix} - \begin{bmatrix} 0 & mg\omega \\ -mg\omega & 0 \end{bmatrix} \begin{bmatrix} v_c \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{r} & \frac{1}{r} \\ \frac{b}{r} & -\frac{b}{r} \end{bmatrix} \begin{bmatrix} \tau_l \\ \tau_r \end{bmatrix} + \begin{bmatrix} F_d \\ \tau_d \end{bmatrix} \tag{16}$$

2.3. Motion situational awareness map for robot

The MSAM is an environmental awareness system which is built based on the image, radar, GPS, the other sensors' data and the robot's state parameters. The MSAM block diagram of the robot is shown in Fig. 2. The MSAM is designed in polar coordinates, and the robot is regarded as the origin of polar coordinates, as shown in Fig. 3. We can adjust the radius of MSAM r_m as we need. By making grids on the MSAM, the detected environment data around the robot are mapped on the polar coordinate grids.

From Fig. 3, the obstacles, other robots and target around robot1 can be mapped on the corresponding grids of the MSAM. The polar coordinates function of the MSAM can be expressed as

$$\begin{cases} x = r\cos\theta \\ y = r\sin\theta \end{cases} \tag{17}$$

We divide the circumferential angle θ $[0\ ,\ 2\pi)$ of the MSAM into n equal parts, each part $\theta_i = \frac{2\pi}{n}$, $(i = 1, 2, \ldots, n)$, and the radius r_m of MSAM can be divided into m equal

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

parts $r_j = \frac{r_m}{m}$, $(j = 1, 2, \ldots, m)$. Therefore, the MSAM can be divided into $m \times n$ grids $s_{ij}(r, \theta)$, $\left(\frac{2\pi i}{n} \le \theta < \frac{2\pi(i+1)}{n}, \frac{r_m j}{m} \le r < \frac{r_m(j+1)}{m}\right)$, which can be represented as

$$s_{ij}(x, y) = \begin{cases} x = r\cos\theta \\ y = r\sin\theta \end{cases}, \left(i\Delta\theta \le \theta < (j+1)\Delta\theta, j\Delta r \le r < (j+1)\Delta r\right) \tag{18}$$

where $\Delta\theta = \frac{2\pi}{n}$, $\Delta r = \frac{r_m}{m}$.

We assume $s_{ij}(j\Delta r\cos i\Delta\theta, j\Delta r\sin i\Delta\theta)$ as the coordinates of grid s_{ij}. It obviously shows that when the obstacles, other robots and target are detected and mapped on the MSAM, the corresponding grids will be occupied. We choose the grids that are close to the robot as the obstacle grid. Therefore, the S_1, S_2 and S_3 correspond the Obs2; S_4, S_5 and S_6 conform the Obs1; S_t represents the target; and the other robots also have the corresponding grids. We take the Obs1 as an example to illustrate the process of MSAM dealing with the obstacles. The grids (S_1, S_2, S_3) are occupied by Obs1, the arc AB is obtained as the closest edge of Obs1 from the Robot1 O, and the coordinates of grid s_2 are the center of arc AB. The radian of arc AB is $3\Delta\theta$; thus, the distance from O to straight line AB is

$$dis_{ob1} = r_o\cos\tfrac{3}{2}\Delta\theta \tag{19}$$

where r_o is the radius of arc AB:

$$r_o = \eta\Delta r = \eta\tfrac{r_m}{m} \tag{20}$$

where η is a positive constant.

There is only one circle $O_1(x_1, y_1)$ satisfies $(O_1O \perp AB)$ & $(r_{o1} = r_o)$. We assume O_1 as a virtual obstacle Obs1 which is represented as

$$(x - x_1)^2 + (y - y_1)^2 = {r_{o1}}^2 \tag{21}$$

Similar to Obs1, we can get the O_2 for Obs2. All the obstacles around the robot can be treated as the circular obstacles, which is easier for APF to deal with the local minima. The MSAM is updated in real time to ensure that the robot can make optimal decision at any time, which can provide rich environmental information for multi-robot path planning and formation control.

3. The Proposed Method

To resolve the obstacle avoidance for multi-robot systems, two issues are under consideration. First, instead of behaving independently, the multi-robot should coordinate to avoid both collision and obstacle. Secondly, robots should efficiently reform the leader–follower formation after avoiding obstacles.[23] Under consideration of these two issues, a switching formation strategy is described. According to the MSAM, if no obstacles are detected, the multi-robot system forms and maintains the predefined formation. When the obstacles occur to robots, both the leader and follower robots will avoid the obstacles based on the IAPF method, and the followers by using PID adaptive tracking control algorithm to reform the predefined formation.

3.1. Improved artificial potential field

The APF method assumes the robots moving in an abstract artificial force field, which is consisted of the repulsive potential field and the attractive potential field in the workspace.[24,25] The force of the robot in the APF is shown in Fig. 4. The target produces an attractive force F_{att}, which monotone diminishing as the distance between the robot and the obstacle decreasing. While the obstacles generate a repulsive force F_{rep}, which is inversely proportional to the distance from the robot to the obstacles, and it is pointing away from the obstacles. γ is the angle between the obstacle and target, F_a is the resultant force of F_{att} and F_{rep}, and the robot will move in the direction of the resultant force until reaching the target.

The most commonly used potential function field is

$$U(q) = U_{\text{att}}(q) + \sum_{i=1}^{n} U_{\text{rep}}(q) \tag{22}$$

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

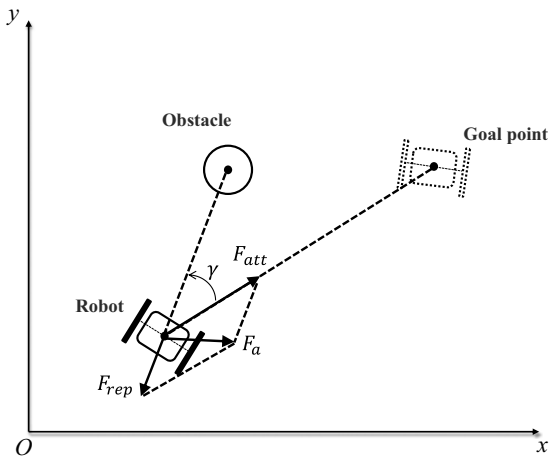

Fig. 4. Strength forced on the robot in artificial potential field.

where $U_{\text{att}}(q)$ and $U_{\text{rep}}(q)$ are the potential functions of the goal point and the obstacles, q is the coordinate of robot $(x, y)^T$, and n is the number of the obstacles within the scope of influence.

The attractive potential function is given by

$$U_{\text{att}}(q) = \frac{1}{2}k_g(q - q_g)^2 \tag{23}$$

where k_g is a positive scaling factor and q_g is the goal position.

The definition of attractive force is calculated as the negative gradient of the attractive potential field

$$F_{\text{att}}(q) = -grad\big(U_{\text{att}}(q)\big) = -k_g\big(q - q_g\big) \tag{24}$$

where F_{att} is a vector toward zero as the robot approaches the target, and the components can be written as

$$\begin{cases} F_{\text{att}}\,(x) = -k_g\big(x - x_g\big) \\ F_{\text{att}}\,(y) = -k_g\big(y - y_g\big) \end{cases} \tag{25}$$

where $F_{\text{att}}(x)$ and $F_{\text{att}}(y)$ is the attractive forces directed to x and y. The repulsive potential function can be represented as

$$U_{\text{rep}}(q) = \begin{cases} \dfrac{1}{2}k_r\left(\dfrac{1}{f\,(q - q_o)} - \dfrac{1}{\rho}\right)^2, f(q - q_o) \le \rho \\ 0, f(q - q_o) > \rho \end{cases} \tag{26}$$

where k_r is a positive scaling factor, q_o is the obstacle position, $f(q - q_o)$ is the distance between the robot and the obstacles, and ρ is the largest impact distance of the MSAM.

The definition of the repulsive force is the negative gradient of the repulsion potential field

$$\begin{aligned} F_{\text{rep}}(q) &= -grad\big(U_{\text{rep}}(q)\big) \\ &= \begin{cases} k_r\left(\dfrac{1}{f(q - q_o)} - \dfrac{1}{\rho}\right)\dfrac{1}{f^2(q - q_o)} \\ grad\big(f(q - q_o)\big), f(q - q_o) \le \rho \\ 0, f(q - q_o) > \rho \end{cases} \end{aligned} \tag{27}$$

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

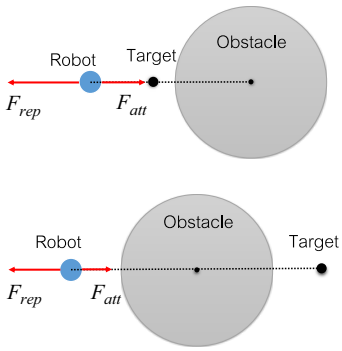

Fig. 5. The common situations of local minima and GNRON.

Similar to the attractive force, F_{rep} can be written as

$$F_{\text{rep}}(x) = \begin{cases} k_r\left(\dfrac{1}{f\,(q-q_o)} - \dfrac{1}{\rho}\right)\dfrac{1}{f^2(q-q_o)}\dfrac{x-x_o}{f\,(q-q_o)}, f(q-q_o) \le \rho \\ 0, f(q-q_o) > \rho \end{cases}$$
$$F_{\text{rep}}(x) = \begin{cases} k_r\left(\dfrac{1}{f(q-q_o)} - \dfrac{1}{\rho}\right)\dfrac{1}{f^2(q-q_o)}\dfrac{y-y_o}{f(q-q_o)}, f(q-q_o) \le \rho \\ 0, f(q-q_o) > \rho \end{cases} \tag{28}$$

Although the traditional APF has a better performance in obstacle avoidance, it has fatal problems of local minima and GNRON. Based on the MSAM, U_{rep} and F_{rep} can be written as

$$U_{\text{rep}} = \begin{cases} \dfrac{1}{2}k_r\left(\dfrac{1}{f\,(m,n)} - \dfrac{1}{\rho}\right)^2, f(m,n) \le \rho \\ 0, f(m,n) > \rho \end{cases} \tag{29}$$

$$\begin{aligned} F_{\text{rep}} &= -grad\big(U_{\text{rep}}(q)\big) \\ &= \begin{cases} k_r\left(\dfrac{1}{f\,(m,n)} - \dfrac{1}{\rho}\right)\dfrac{1}{f^2(m,n)} \\ grad\big(f\,(m,n)\big), f(m,n) \le \rho \\ 0, f(m,n) > \rho \end{cases} \end{aligned} \tag{30}$$

where $f(m,n)$ represents the distance between the obstacles and robot. F_{rep} is related to the MSAM. We can change m and n to adjust F_{rep}, which improves the practicality of the potential function and is suitable for any kinds of obstacles.

The common situations of the GNRON and local minima are as shown in Fig. 5. We define the local minima and GNRON problem as

$$\sum_{i=1}^{n} F_{\text{rep}} \ge -F_{\text{att}} \tag{31}$$

As can be seen from Fig. 6, if the target T_2 and robot1 are both within the impact range r_o of the Obs2, and the $f\,(q-q_o) \le f\,\big(q-q_g\big)$, we treat this situation as a GNRON problem. To solve this problem, we make $U_{\text{att}} \ge U_{\text{rep}}$. U_{rep} and F_{rep} are designed as

$$U_{\text{att}} = \begin{cases} \dfrac{1}{2}k_g(q-q_g)^2 \\ \dfrac{1}{2}k_g(q-q_g)^2 + \dfrac{1}{2}k_r\left(\dfrac{1}{f\,(m,n)} - \frac{1}{\rho}\right)^2, f\,\big(q-q_g\big) < f\,(q-q_o) < \rho \end{cases} \tag{32}$$

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

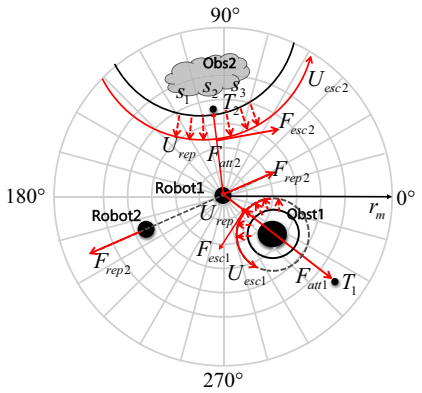

Fig. 6. The IAPF based on MSA.

$$F_{\text{att}} = \begin{cases} k_g(q - q_g) \\ k_g(q - q_g) + k_r \left(\dfrac{1}{f\,(m,n)} - \dfrac{1}{\rho} \right) \dfrac{1}{f(m,n)^2} \\ grad\,(f\,(m,n))\,, f\left(q - q_g\right) < f\,(q - q_o) < \rho \end{cases} \tag{33}$$

In Fig. 6., when the attractive force from T_1 and the repulsive force from Obs1 are equal and have the opposite direction, the resultant force is zero. It will cause the robot to be trapped in the local minima or oscillations. To solve this problem, we employ the rotating potential field U_{esc} when the obstacles are within the impact range of the MSAM, which can produce force F_{esc} to guide the robot to escape from the local minima. As shown in Fig. 6, U_{esc} is perpendicular to the attractive potential field U_{rep}, and the center of U_{esc} is the obstacles. If $\gamma \neq 0$, U_{esc} has the same direction of $F_{\text{att}} sin\gamma$; otherwise, the direction of U_{esc} is counterclockwise around the obstacles.

The rotating escape potential field U_{esc} can be represented as

$$U_{\text{esc}}\,(q) = \begin{cases} \dfrac{1}{2} k_e (q - q_o)^2, f\,(m,n) \leq \rho \\ 0, f\,(m,n) > \rho \end{cases} \tag{34}$$

where k_e is a positive scaling factor. The purpose of introducing the rotating potential field is to escape the local optimum ensuring that the robot does not deviate from the desired route. q_o is the obstacle position, $f\,(q - q_o)$ is the distance between the robot and the obstacles, and ρ is the largest impact distance of the obstacles.

The definition of escape force F_{esc} is calculated as the negative gradient of the rotating escape potential field

$$F_{\text{esc}}\,(q) = \begin{cases} k_e (q - q_o), f\,(m,n) \leq \rho \\ 0, f\,(m,n) > \rho \end{cases} \tag{35}$$

The direction of F_{esc} is perpendicular to the attractive force F_{att}.

The collision avoidance between the robots is another tough issue in multi-robot systems. When the robot R_i approaches to R_j, the repulsive potential will be generated to avoid collisions, which is designed as

$$U_{ij}\,(q) = \begin{cases} -k_{ij} \left(\ln(q_{ij} - r) - \frac{q_{ij}}{r_{ij} - r} \right), \; q_{ij} \leq \rho \wedge q_{ij} \neq r \\ 0, \; q_{ij} > \rho \end{cases} \tag{36}$$

where k_{ij} is a positive coefficient, q_{ij} is the distance of R_i and R_j, and r represents the safe distance between the robots.

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

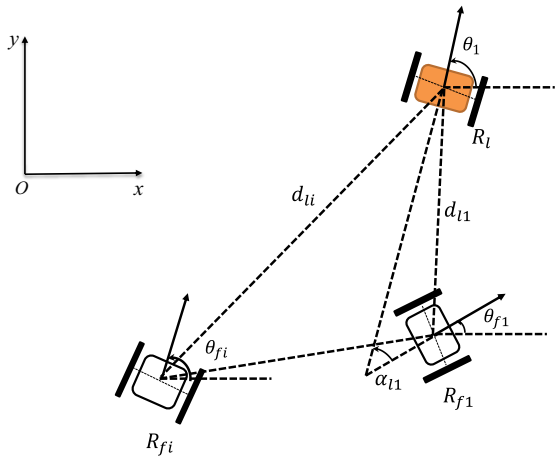

Fig. 7. The model of leader–follower formation control.

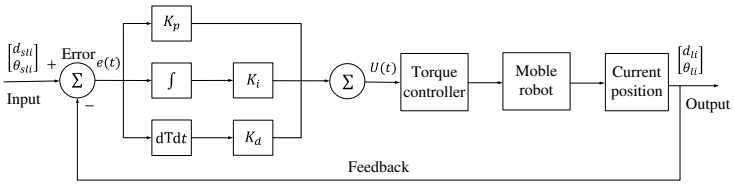

Fig. 8. PID adaptive tracking control algorithm.

From Eq. (36), the repulsive force can be expressed as

$$F_{ij}(q) = -\nabla U_{ij}(q) = \begin{cases} k_{ij}\left(\frac{1}{q_{ij}-r} - \frac{1}{\rho-r}\right), & q_{ij} \le \rho \wedge q_{ij} \ne r \\ 0, & q_{ij} > \rho \end{cases} \tag{37}$$

From Eq. (37), when $(q_{ij} - r) \to 0$, $F_{ij}(q) \to \infty$. Therefore, the robots will not collide with each other.

The resultant force of the robot is

$$F_a(q) = \sum_{i=1}^{n} F_{\text{rep}}(q) + \sum_{i=1}^{n} F_{\text{esc}}(q) + F_{\text{att}}(q) + \sum_{j=1}^{n_0} F_{ij}(q) \tag{38}$$

where n_0 is the number of the robots within the scope of influence.

Remark 1. The existing IAPF method,[13,15,16,26] designed the potential function based on a priori environment model. Unfortunately, a realistic changing environment significantly shortens the life of any priori model. In an unknown complex environment, it is almost impossible to establish the potential functions without local minima. This paper designed the adjustable and flexible potential functions (29) based on the MSAM. In order to reduce the local minima, we introduce a rotating potential field (34). Moreover, we designed the repulsive potential function (36) to avoid collisions between robots.

3.2. Multi-robot formation control

3.2.1. Leader–follower formation model. For the multi-robot systems, there are three kinds of approaches for formation control: the leader–follower method,[27] behavior-based method[28] and the virtual structure method.[29,30] Formation control mainly includes assignment of feasible formations and switching between formations. In this paper, the leader–follower formation approach[31] is adopted to the system. The robot R_l is assigned to be the leader robot, which can determine the motion of each follower robots $R_{fi}(i = 1, 2, \ldots, n)$. The formation control model is shown in Fig. 7. d_{l1} is defined

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

as the desired distance between R_{fi} and R_l. α_{li} is the desired bearing angle from the orientation of the follower robot to the axis connecting R_{fi} and R_l. We assume the way-point posture of the follower robot R_{fi}

$$P_{fi} = (\, x_{fi}\ y_{fi}\ \theta_{fi}\,)^T$$

and

$$\begin{bmatrix} x_{fi} \\ y_{fi} \end{bmatrix} = \begin{bmatrix} x_l(t) - d_{li}\cos\left(\alpha_{li} + \theta_l(t)\right) \\ y_l(t) - d_{li}\sin\left(\alpha_{li} + \theta_l(t)\right) \end{bmatrix} \tag{39}$$

$$\theta_{fi} = \arctan\Big(\big(y_{fi}(t) - y_{fi}(t-1)\big)/\big(x_{fi}(t) - x_{fi}(t-1)\big)\Big) \tag{40}$$

Then the kinematic equations of the system are expressed as

$$\begin{cases} \dot{x}_{fi} = v_{fi}\cos\theta_{fi} \\ \dot{y}_{fi} = v_{fi}\sin\theta_{fi} \\ \dot{\theta}_{fi} = \omega_{fi} \end{cases} \tag{41}$$

In order to keep the formation, the dynamic equation of the followers and leaders is described as

$$\begin{cases} \dot{d}_{li} = v_{fi}\cos\beta_i - v_l\cos\alpha_{li} + \omega_{fi}\sin\beta_i \\ \dot{\alpha}_{li} = \dfrac{1}{d_{li}}\big(v_l sin\alpha_{li} - v_{fi}\cos\beta_i - d_{li}\omega_l + \omega_{fi}\cos\beta_i\big) \\ \dot{\theta}_{fi} = \theta_{fi} \end{cases} \tag{42}$$

Equation (42) can be rewritten as

$$\begin{bmatrix} \dot{d}_{li} \\ \dot{\alpha}_{li} \end{bmatrix} = \boldsymbol{F} + \boldsymbol{G}\boldsymbol{u_o} \tag{43}$$

where $\boldsymbol{F} = \begin{bmatrix} -v_{li}\cos\alpha_{li} \\ \frac{v_{li}\sin\alpha_{li} - d_{li}\omega_l}{d_{li}} \end{bmatrix}$, $G = \begin{bmatrix} \cos\beta_i & \sin\beta_i \\ -\frac{\sin\beta_i}{d_{li}} & \frac{\cos\beta_i}{d_{li}} \end{bmatrix}$, $u_o = \begin{bmatrix} v_{fi}\ \omega_{fi} \end{bmatrix}^T$, $\beta_i = \theta_l + \alpha_{li} - \theta_{fi}$. Since $det(\boldsymbol{G}) \neq 0$, the inverse of $\boldsymbol{G}$ exists. Denoting $u = Gu_o$ and substituting it into (43) give

$$\boldsymbol{u} = -\boldsymbol{F} - \boldsymbol{K}\boldsymbol{e} \tag{44}$$

where $\boldsymbol{K}$ indicates positive constant gains. Thus, the tracking error $\boldsymbol{e}$ converges to zero exponentially, meaning that the followers could maintain some given separation distance and bearing angle from the leader.

3.2.2. PID adaptive tracking control. To ensure that each robot can track its way-points and form the formation correctly, an adaptive tracking control algorithm was proposed based on the PID control algorithm. The distinguishing feature of the PID controller is the ability to use the three control terms of proportional, integral and derivative influence on the controller output to apply accurate and optimal control.[30] Figure 8 shows the principles of how these terms are generated and applied. The PID controller continuously calculates an error value $e\,(t)$ as the difference between a desired safe distance and tracking angle of the followers $\begin{bmatrix} d_{sli}\ \theta_{sli} \end{bmatrix}^T$ and a measured process variable $\begin{bmatrix} d_{li}\ \theta_{li} \end{bmatrix}^T$, and applies a correction based on proportional, integral, and derivative terms. The controller attempts to minimize the error over time by adjustment of a control variable $U(t)$, which is used to control the torque controller of robots, through the MSAM to calculate the current position of robots, then the position is sent to the PID controller.

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

According to the leader–follower formation model, the distance d_{li} and angle θ_{li} between the followers R_{fi} and leader are calculated as

$$d_{li} = \sqrt{\left(x_{fi} - x_{li}\right)^2 + \left(y_{fi} - y_{li}\right)^2} \tag{45}$$

$$\theta_{li} = \arctan \frac{y_{fi} - y_{li}}{x_{fi} - x_{li}} \tag{46}$$

The formation tracking error is

$$\boldsymbol{e} = \left[\, e_d,\, e_\theta \,\right]^T = \left[\, d_{sli} - d_{li},\, \theta_{sli} - \theta_{li} \,\right]^T \tag{47}$$

The tracking error $\boldsymbol{e}$ converges to zero exponentially, which means that the followers could maintain the given separation distance and bearing angle from the leader in the leader– formation. The overall PID control function can be expressed as

$$U_{\text{pid}}(t) = K_p \boldsymbol{e}(t) + K_i \int_0^t \boldsymbol{e}(\tau) d\tau + K_d \frac{d\boldsymbol{e}(t)}{dt} \tag{48}$$

where K_p, K_i and K_d are non-negative and denote coefficients for the proportional, integral, and derivative terms, $\boldsymbol{e}(t)$ represents the control error, and $U_{\text{pid}}(t)$ represents the output control variable.

Our PID controller is described in the following form:

$$\boldsymbol{u}_{\text{pid}} = \boldsymbol{P}^{\boldsymbol{T}} \boldsymbol{k}_{\text{pid}} \tag{49}$$

where $\boldsymbol{P}^{\boldsymbol{T}} = \begin{bmatrix} e_d & \int_0^t e_d(\tau) d\tau & \dot{e}_d & 0 & 0 & 0 \\ 0 & 0 & 0 & e_\theta & \int_0^t e_\theta(\tau) d\tau & \dot{e}_\theta \end{bmatrix}^T$, $\boldsymbol{k}_{\text{pid}} = \left[\, k_{dp}\ k_{di}\ k_{dd}\ k_{\theta p}\ k_{\theta i}\ k_{\theta d} \,\right]$.

From (44) and (66), we can obtain

$$e_u = u_{\text{pid}} - u \tag{50}$$

We assume that there exists an optimal bounded time-varying parameter vector $\boldsymbol{k}^*_{\text{pid}}$, which holds

$$\boldsymbol{u}(\boldsymbol{e}, t) = \boldsymbol{P}^{\boldsymbol{T}} \boldsymbol{k}^*_{\text{pid}} + \boldsymbol{\delta}(\boldsymbol{e}, t) \tag{51}$$

where $\boldsymbol{\delta}(\boldsymbol{e}, t)$ is bounded and satisfies

$$\boldsymbol{\delta}^T \boldsymbol{\delta} \le \tau_0 \boldsymbol{e}^T \boldsymbol{e} + \tau_1 \tag{52}$$

where τ_0 and τ_1 are positive constants.

To obtain the $\boldsymbol{k}^*_{\text{pid}}$ for our controller, from (65) and (66)

$$\boldsymbol{e}_u = \boldsymbol{P}^{\boldsymbol{T}} \boldsymbol{k}_{\text{pid}} - \boldsymbol{P}^{\boldsymbol{T}} k^*_{\text{pid}} - \boldsymbol{\delta} \tag{53}$$

where $\tilde{\boldsymbol{k}}_{\text{pid}} = \boldsymbol{k}^*_{\text{pid}} - \boldsymbol{k}_{\text{pid}}$ denotes the error vector of adaptive PID control parameter. We have

$$\boldsymbol{k}_{\text{pid}} = \boldsymbol{P}\left(\boldsymbol{e}_u + \boldsymbol{P}^T \boldsymbol{k}^*_{\text{pid}} + \boldsymbol{\delta}\right) \tag{54}$$

where $\boldsymbol{P}^T \boldsymbol{k}^*_{\text{pid}} + \boldsymbol{\delta}$ can be treated as a positive constant. The gradient of with $\boldsymbol{k}_{\text{pid}}$ can be calculated by

$$\dot{\boldsymbol{k}}_{\text{pid}} = \boldsymbol{P}\dot{\boldsymbol{e}}_u \tag{55}$$

Deriving the tracking error e in (44) yields

$$\begin{aligned} \dot{\boldsymbol{e}} &= \boldsymbol{F} + \boldsymbol{u}_{\text{pid}} \\ &= \boldsymbol{F} + \boldsymbol{u}_{\text{pid}} - \boldsymbol{u} + \boldsymbol{u} \\ &= -\boldsymbol{K}\boldsymbol{e} + \boldsymbol{e}_u \end{aligned} \tag{56}$$

To analyze the stability of the dynamic error system, the following theorem is established.

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

Theorem 1. *Consider the dynamic PID control represented by (65) and assume that the assumption (53) is satisfied. Then, the tracking errors (47) can converge to 0 by appropriately choosing the parameters. The leader–follower formation control system is asymptotically stable.*

Proof. The Lyapunov function is given as follows:

$$V = \tfrac{1}{2}\boldsymbol{e}^T\boldsymbol{e} + \tfrac{1}{2}\tilde{\boldsymbol{k}}_{\text{pid}}^T\tilde{\boldsymbol{k}}_{\text{pid}} \tag{57}$$

The time derivative of the Lyapunov function (57) along with (53), (55) and (56) satisfies

$$\begin{aligned}\dot{V} &= e^T\dot{\boldsymbol{e}} + \tilde{\boldsymbol{k}}_{\text{pid}}^T\left(\dot{\boldsymbol{k}}_{\text{pid}}^* - \dot{\boldsymbol{k}}_{\text{pid}}\right)\\ &= \boldsymbol{e}^T(-\boldsymbol{K}\boldsymbol{e} + \boldsymbol{e}_u) - \tilde{\boldsymbol{k}}_{\text{pid}}^T\boldsymbol{P}\boldsymbol{e}_u + \tilde{\boldsymbol{k}}_{\text{pid}}^T\dot{\boldsymbol{k}}_{\text{pid}}^*\\ &= -\boldsymbol{e}^T K e + \boldsymbol{e}^T\boldsymbol{e}_u - \left(\boldsymbol{e}_u^T + \boldsymbol{\delta}^T\right)\boldsymbol{e}_u + \tilde{\boldsymbol{k}}_{\text{pid}}^T\dot{\boldsymbol{k}}_{\text{pid}}^*\\ &= -\boldsymbol{e}^T K e + \boldsymbol{e}^T\boldsymbol{e}_u - \boldsymbol{e}_u^T\boldsymbol{e}_u - \boldsymbol{\delta}^T\boldsymbol{e}_u + \tilde{\boldsymbol{k}}_{\text{pid}}^T\dot{\boldsymbol{k}}_{\text{pid}}^*\end{aligned} \tag{58}$$

For estimating the upper bound of $\dot{V}$, we obtain

$$\begin{aligned}\boldsymbol{e}^T\boldsymbol{e}_u &= -\left(\tfrac{1}{4}\boldsymbol{e}_u^T\boldsymbol{e}_u - \tfrac{1}{2}\boldsymbol{e}_u^T e - \tfrac{1}{2}\boldsymbol{e}^T\boldsymbol{e}_u + \boldsymbol{e}^T\boldsymbol{e}\right) + \tfrac{1}{4}\boldsymbol{e}_u^T\boldsymbol{e}_u + \boldsymbol{e}^T\boldsymbol{e}\\ &= -\left(\tfrac{1}{2}\boldsymbol{e}_u - \boldsymbol{e}\right)^T\left(\tfrac{1}{2}\boldsymbol{e}_u - \boldsymbol{e}\right) + \tfrac{1}{4}\boldsymbol{e}_u^T\boldsymbol{e}_u + \boldsymbol{e}^T\boldsymbol{e}\\ &\le \tfrac{1}{4}\boldsymbol{e}_u^T\boldsymbol{e}_u + \boldsymbol{e}^T\boldsymbol{e}\end{aligned} \tag{59}$$

Similar to (59), $-\boldsymbol{\delta}^T\boldsymbol{e}_u \le \frac{1}{4}\boldsymbol{e}_u^T\boldsymbol{e}_u + \boldsymbol{\delta}^T\boldsymbol{\delta}$, $\tilde{\boldsymbol{k}}_{\text{pid}}^T\dot{\boldsymbol{k}}_{\text{pid}}^* \le \frac{1}{4}\tilde{\boldsymbol{k}}_{\text{pid}}^T\tilde{\boldsymbol{k}}_{\text{pid}} + \left(\dot{\boldsymbol{k}}_{\text{pid}}^*\right)^T\dot{\boldsymbol{k}}_{\text{pid}}^*$. We have

$$\begin{aligned}\dot{V} &\le -\boldsymbol{e}^T\boldsymbol{K}\boldsymbol{e} + \tfrac{1}{4}\boldsymbol{e}_u^T\boldsymbol{e}_u + \boldsymbol{e}^T\boldsymbol{e} - \boldsymbol{e}_u^T\boldsymbol{e}_u + \tfrac{1}{4}\boldsymbol{e}_u^T\boldsymbol{e}_u + \boldsymbol{\delta}^T\boldsymbol{\delta} + \tfrac{1}{4}\tilde{\boldsymbol{k}}_{\text{pid}}^T\tilde{\boldsymbol{k}}_{\text{pid}} + \left(\dot{\boldsymbol{k}}_{\text{pid}}^*\right)^T\dot{\boldsymbol{k}}_{\text{pid}}^*\\ &= -\tfrac{1}{2}\boldsymbol{e}_u^T\boldsymbol{e}_u - \boldsymbol{e}^T(K - I)\boldsymbol{e} + \boldsymbol{\delta}^T\boldsymbol{\delta} + \tfrac{1}{4}\tilde{\boldsymbol{k}}_{\text{pid}}^T\tilde{\boldsymbol{k}}_{\text{pid}} + \left(\dot{\boldsymbol{k}}_{\text{pid}}^*\right)^T\dot{\boldsymbol{k}}_{\text{pid}}^*\\ &\le -\tfrac{1}{2}\boldsymbol{e}_u^T\boldsymbol{e}_u - \boldsymbol{e}^T(K - I)\boldsymbol{e} + \tau_0\boldsymbol{e}^T\boldsymbol{e} + \tau_1 + \tfrac{1}{4}\left\|\tilde{\boldsymbol{k}}_{\text{pid}}\right\|^2 + \left\|\dot{\boldsymbol{k}}_{\text{pid}}^*\right\|^2\\ &= -\tfrac{1}{2}\boldsymbol{e}_u^T\boldsymbol{e}_u - \boldsymbol{e}^T[K - (1+\tau_0)I]\boldsymbol{e} + \tau_1 + \tfrac{1}{4}\left\|\tilde{\boldsymbol{k}}_{\text{pid}}\right\|^2 + \left\|\dot{\boldsymbol{k}}_{\text{pid}}^*\right\|^2\end{aligned} \tag{60}$$

where I is the identity matrix. Since $\dot{\boldsymbol{k}}_{\text{pid}}^*$ and $\boldsymbol{\delta}$ are upper bounded, there exists a positive constant b satisfying

$$\tau_1 + \left\|\dot{k}_{\text{pid}}^*\right\|^2 = b \tag{61}$$

Substituting (61) into (60), we have

$$\begin{aligned}\dot{V} &\le -\tfrac{1}{2}\boldsymbol{e}_u^T\boldsymbol{e}_u - \boldsymbol{e}^T\left[K - (1+\tau_0)I\right]\boldsymbol{e} + \tfrac{1}{4}\left\|\tilde{k}_{\text{pid}}\right\|^2 + b\\ &\le -aV + b\end{aligned} \tag{62}$$

where $a = \min(min2(k_{1,2} - 1 - \tau_0), 0.5)$.

Hence

$$V(t) \le \left(V(0) - \tfrac{b}{a}\right)e^{-at} + \tfrac{b}{a} \tag{63}$$

which indicates

$$\|\boldsymbol{e}(t)\| \le \sqrt{\|e(0)\|^2 + \left\|\tilde{\boldsymbol{k}}_{\text{pid}}(0)\right\|^2 + \tfrac{2b}{a}e^{-(1/2)at}} + \sqrt{\tfrac{2b}{a}} \tag{64}$$

Therefore, the tracking errors d_{li} and θ_{li} converge to an enough small neighborhood of the origin, and $\|\boldsymbol{e}(t)\|$ converges to $\sqrt{\frac{2b}{a}}$ as $t \to 0$. This completes the proof. ■

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

Remark 2. The reason we choose the PID control algorithm for multi-robot formation control is that it has a simple structure, fast response, easier parameter adjustment, is widely used in various fields and has achieved remarkable results.[32–34] The formation PID control proposed in ref. [32] has better performance in robust formation control. Unfortunately, it did not consider the case of the obstacle and collision avoidance in the working environment. The proposed adaptive PID algorithm is combined with the IAPF and MSAM, which can be used for formation control after the obstacle and collision avoidance.

3.3. Multi-robot obstacle-avoidance control algorithm

In this paper, the multi-robot obstacle-avoidance control algorithm mainly solves the obstacle avoidance, the collision avoidance between robots and formation control.

3.3.1. Avoiding the collisions between robots. For avoiding the collisions between robots, we designed two strategies: one is the robots avoiding collisions each other when they meet and treat each other as the moving obstacles and the other is to establish a priority model[15] for each individual robot. When the robots meet each other, high-priority robots have the right to pass priority, and the other robots have to stop and wait until the high-priority robots pass through.

In the system, we set a fixed ID id_i $(i = 1, 2, \ldots, n)$ for each robot, and the leader has the highest priority p_1. For the followers, the priority is p_i $(i = 2, \ldots, n)$. If there is no formation control between the robots, the priority is calculated as

$$p_i = id_i,\ (i = 1, 2, \ldots, n) \tag{65}$$

If the system was in formation control, the priority is consisted by p_i^d and p_i^a, and p_i is calculated as

$$p_i = \begin{cases} p_i^d,\ \left(p_i^d \neq p_j^d\right) \\ p_i^a,\ \left(p_i^d = p_j^d\right) \\ id_i,\ \left(p_i^d = p_j^d, p_i^a = p_j^a\right) \end{cases} \tag{66}$$

where $p_i^d = \frac{1}{d_{i1}}$ and $p_i^a = \text{sgn}\,(\alpha_{li})\left(\frac{\pi}{2} - |\alpha_{li}|\right)$, for follower robots R_i and R_j. When $p_i^d \neq p_j^d$, the comparison of p_i and p_j depends on the comparison of p_i^d and p_j^d; when $p_i^d = p_j^d$, the comparison of p_i and p_j relies on the comparison of p_i^d and p_j^d; and for other case the comparison of p_i and p_j depends on the comparison of id_i and id_j.

3.3.2. The obstacle avoidance for multi-robot formation. Figure 9 shows the obstacle-avoidance control flowchart for the multi-robot system. As the gradient of the IAPF will interfere with the PID term in the formation control, the PID term to achieve formation may generate new local minima for the IAPF. Our solution is to design a priority strategy for avoiding obstacles. If there are no obstacles, the robots form and maintain the predefined formation based on the PID adaptive tracking control algorithm. For crossing and avoiding obstacles, based on the MSAM, the leader tries to avoid the obstacles by using the IAPF, and the followers have to switch into an obstacle-avoidance formation with the leader. After checking the distance between the two neighboring followers, if the collisions occur, based on the priority model, the followers have to change their paths and avoid the collisions each other. If there is no collision avoidance, the followers calculate the current distance and angle with the leader in real time. After the obstacle avoidance, the followers have the ability to reform the predefined formation by using the PID adaptive tracking control algorithm, until all the robots reach the targets.

4. Simulations

To evaluate the performance of the proposed strategies, a series of simulations have been implemented in MATLAB. The physical parameters of the robots are $b_i = 0.17$ and $r_i = 0.12$, and the radius of the robot MSAM is defined as $r_m = 5$. The parameters for IAPF algorithm are defined as $k_g = 5$, $k_r = 30$, $k_e = 0.01$ and $\rho = 3$. The parameters for PID adaptive tracking control algorithm are defined as $K_p = 8$, $K_i = 5$ and $K_d = 5$. The initial values of PID parameters are all selected to be zero.

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

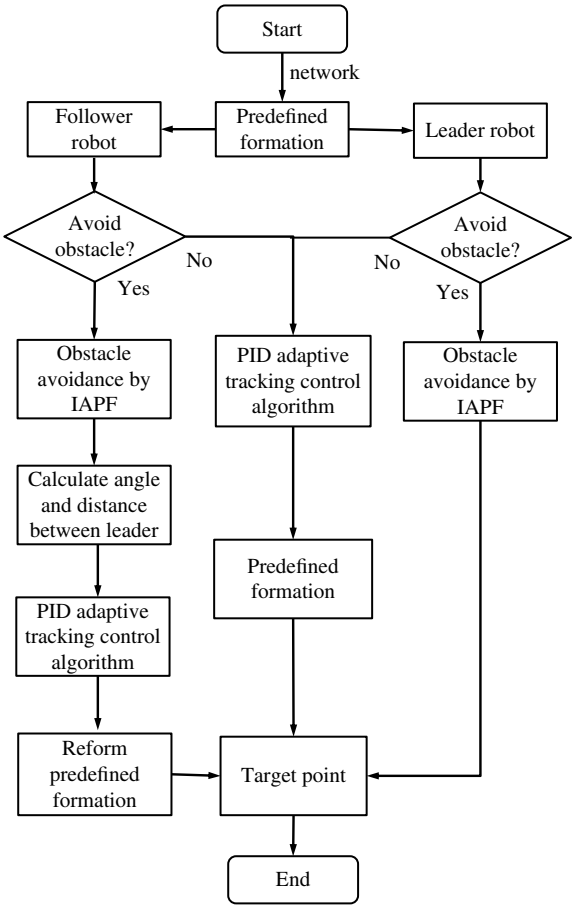

Fig. 9. The obstacle-avoidance control flow chart for multi-robot system.

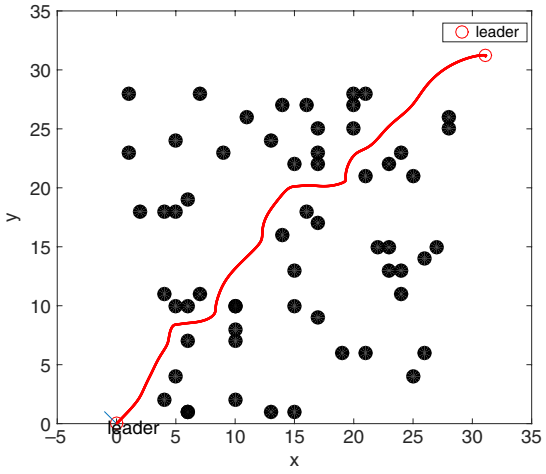

Fig. 10. Trajectories of the single robot crossing and avoiding the obstacles.

The simulation step length of the robots is 0.08. All of the parameters in the simulation are treated as non-dimensional.

4.1. Obstacle avoidance of single robot

Figure 10 shows the trajectories of the single robot crossing and avoiding the random obstacles based on the IAPF. It is obvious that the robot has a better performance in avoiding the complex

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

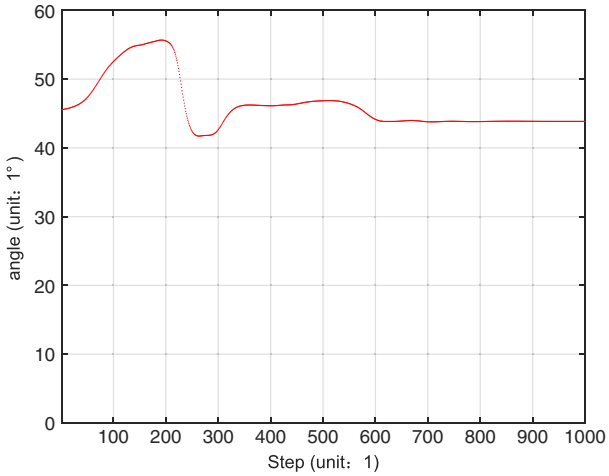

Fig. 11. The angle curve of the resultant force.

environment obstacles, and the angle curve of the resultant force is shown in Fig. 11 (we assumed the x-axis to the right as the positive direction 0°), which can match the trajectories as well.

4.2. Collision avoidance between robots

For the multi-robot collision avoidance, we explored two different approaches: one is designed based on the priority model. When robots meet each other, the higher priority robots have the right to pass priority, and the other robots have to stop and wait until the high-priority robot pass through, as shown in Fig. 12(a) and (b). We simulate with six robots moving toward the opposite direction in pairs, and the simulation results show that this method can realize the collision avoidance. Another method does not consider the priority model. All of the robots can be treated as the moving obstacles, so the robots avoid each other when they meet. From Fig. 12(c) and (d), the simulation results show that this method can avoid collision successfully as well. In addition, the digitals in Fig. 12 represent the current position of the robots at the same time.

4.3. Obstacle avoidance of multi-robot formation

Figure 13 shows the trajectories of six robots crossing two static obstacles. The followers form a predefined formation with the leader. The initial states of the leader and follower robots are given in Table I. The desired distances of the five followers with the leader are $d_{l1} = d_{l2} = 1.4$, $d_{l3} = d_{l5} = 2.8$ and $d_{l4} = 2$. The positions of two obstacles are (7, 2.5) and (7, −2.5), and the radius is 1.5, so the distance of two obstacles is 2. The formation is supposed to cross two obstacles.

As shown in Fig. 13, at the beginning, the robots are moving in the predefined formation. In order to cross and avoid the obstacles, the robots have to change the formation. When the robots detect an obstacle, the leader avoids the obstacles based on the IAPF, and the followers calculate their new desired bearing angles with the leader in real time. Then according to the PID adaptive tracking control algorithm and the IAPF, all of the robots can cross and avoid the obstacles successfully. After crossing the obstacles, six robots quickly reform the predefined formation and move toward the targets. The trajectory tracking errors of robots 1–5 are shown in Fig. 14. Although oscillations occur during the switching formation, the trajectory tracking errors can quickly converge to zero and the followers' error curves all match the trajectories as well. In the light of the leader crosses through two symmetrical obstacles, so the leader and robot 4 have the same trajectory, and the trajectories of robot 1 and robot 2 are symmetric as well as trajectories of robots 3 and 5.

Figure 15 shows the trajectories of six robots crossing and avoiding three static obstacles. The initial states of the robots are given in Table II and the desired distances of five followers with the leader are $d_{l1} = d_{l2} = 1.4$, $d_{l3} = d_{l5} = 2.8$, and $d_{l4} = 2$. The positions of three obstacles are (7, 2.5), (7, −2.5) and (14, 1.3). The radius is 1.5. The robots are required to cross and avoid three obstacles.

As shown in Fig. 15, at the beginning, the robots are moving in the predefined formation. Then the robots start to change the formation to cross the two obstacles. When the leader meets the third obstacle, it avoids the obstacle based on the IAPF algorithm. The followers calculate their new desired

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

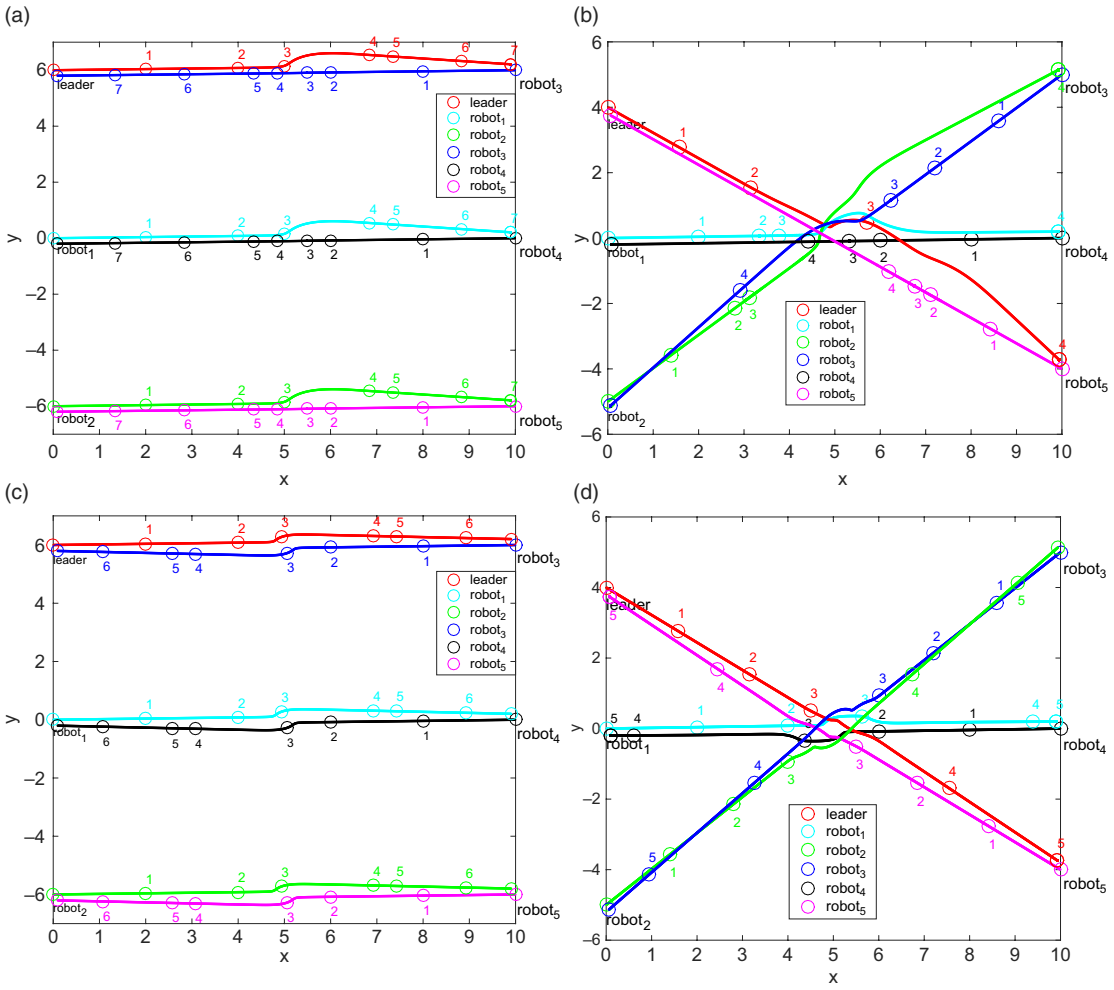

Fig. 12. Trajectories of collision avoidance between robots.

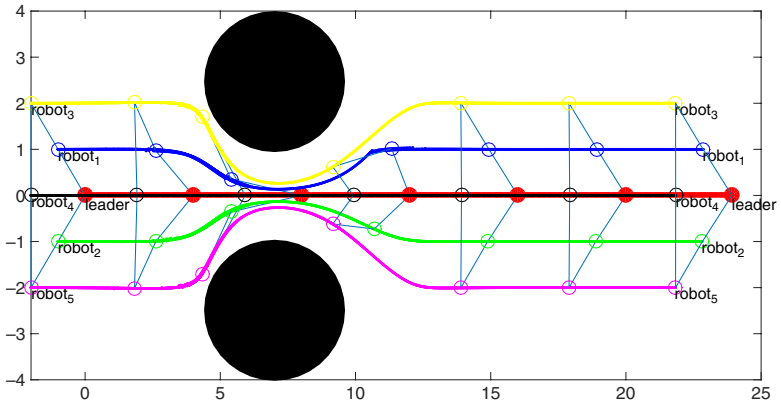

Fig. 13. The trajectories of six robots crossing and avoiding obstacles.

bearing angles with the leader according to the PID adaptive tracking control algorithm and the IAPF, and that is the reason robots 3 and 4 back to avoid the obstacles from the bottom of the barrier. After all of the robots cross and avoid the obstacles successfully, six robots quickly reform the predefined formation. The trajectory tracking errors of the followers robots 1–5 are shown in Fig. 16. Although oscillations occur during the switching formation, the trajectory tracking errors can quickly converge to zero and the error curves of the followers all match the trajectories as well.

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

Table I. The initial states of the leader and follower robots.

Robots	**Leader**	**Robot1**	**Robot2**	**Robot3**	**Robot4**	**Robot5**
(x, y)	(0,0)	(−1,1)	(−1,−1)	(−2,2)	(−2,0)	(−2,−2)

Table II. The initial states of the leader and follower robots.

Robots	**Leader**	**Robot1**	**Robot2**	**Robot3**	**Robot4**	**Robot5**
(x,y)	(0,0)	(−1,1)	(−1,−1)	(−2,2)	(−2,0)	(−2,−2)

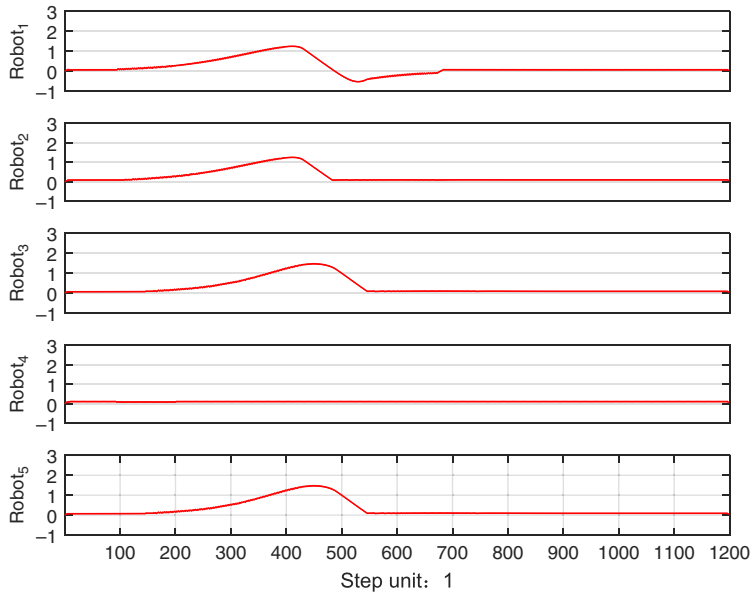

Fig. 14. The tracking errors of follower robots.

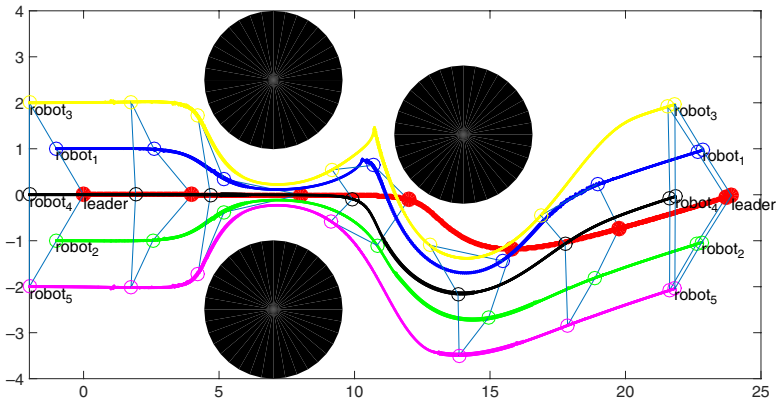

Fig. 15. The trajectories of six robots crossing and avoiding obstacles.

Figure 17 shows the trajectories of six robots crossing and avoiding the random distribution obstacles. The initial states of the leader and follower robots are given in Table III. The desired distances of the five followers with the leader are $d_{l1} = d_{l2} = 2.8$, $d_{l3} = d_{l5} = 5.7$ and $d_{l4} = 4$. The robots are required to cross and avoid the random distribution static obstacles.

As shown in Fig. 17, at the beginning, the robots are moving in the predefined formation. When they come across the obstacles, the leader avoids the obstacles based on IPAF, and the followers calculate their new desired bearing angles with the leader based on the PID adaptive tracking algorithm

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

Table III. The initial states of the leader and follower robots.

Robots	**Leader**	**Robot1**	**Robot2**	**Robot3**	**Robot4**	**Robot5**
(x,y)	(0,15)	(−2,17)	(−2,−13)	(−4,19)	(−4,15)	(−4,11)

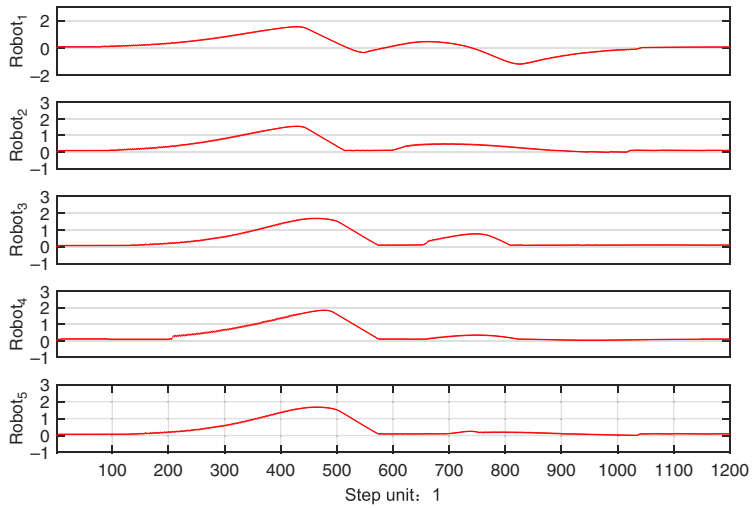

Fig. 16. The trajectory tracking errors of the follower robots.

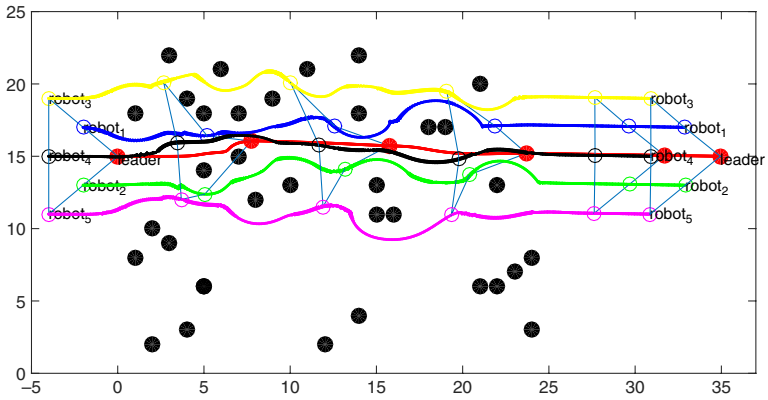

Fig. 17. The trajectories of six robots crossing and avoiding random distribution obstacles.

and the IAPF. All of the robots can cross and avoid random distribution obstacles successfully. After crossing obstacles, the robots try to reform the original formation. The trajectory tracking errors of the follower robots 1–5 are shown in Fig. 18. Although the oscillations occur during switching formations, the trajectory tracking errors can quickly converge to zero and the followers' error curves all match the trajectories as well.

5. Conclusion

In this paper, the IAPF and PID adaptive tracking control methods have been proposed for multi-robot obstacle and collision avoidance. Based on the IAPF, the local minimum has been reduced greatly, the leader has planned an optimal path without any collisions and guided the followers to the target, and the followers can switch into an obstacle-avoidance formation. The PID adaptive tracking control can make the multi-robot system to recover its original formation when the obstacle avoidance has been completed. Moreover, with the robot priority model, the proposed strategy has solved the collisions between the robots as well. Finally, we have simulated several common situations. The results demonstrated the effectiveness of the proposed approaches to solve the obstacle and collision avoidance for multi-robot system. However, it is still a very difficult task to achieve formation control

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

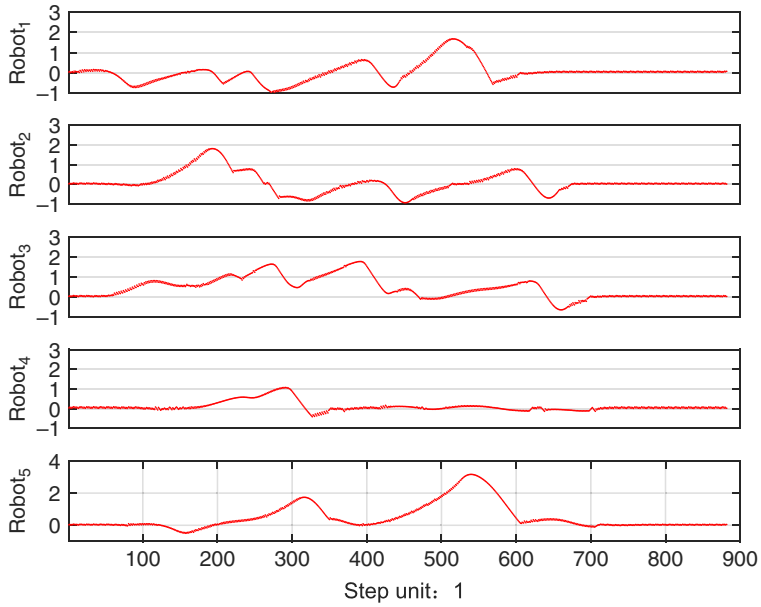

Fig. 18. The trajectory tracking errors of the follower robots.

globally, which will be our future direction. Furthermore, we also plan to optimize and extend the algorithm for flocking control.

Acknowledgement
This paper is funded by International Graduate Exchange Program of Beijing Institute of Technology.

References

1. T. Mac, C. Copot, D. T. Tran, R. De Keyser, "A hierarchical global path planning approach for mobile robots based on multi-objective particle swarm optimization," *Appl. Soft Comput.* **59**, 68–76 (2017).
2. I. Y. Song, D. Y. Kim, H.-S. Ahn and V. Shin, "Simultaneous Pedestrian and Multiple Mobile Robot Localization Using Distributed Extended Kalman Filter," *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, Bangkok, Thailand (2008) pp. 1056–1069.
3. E. Glorieux, S. Riazi and B. Lennartson, "Productivity/energy optimization of trajectories and coordination for cyclic multi-robot systems," *Robot. Comput. Integr. Manuf.* **49**, 152–161 (2018).
4. P. E. Hart, N. J. Nilsson and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Syst. Sci. Cybern.* **4**, 100–107 (1968).
5. W. Howden, "The sofa problem," *Comput. J.* **11**, 299–301 (1968).
6. E. Masehian and M. Amin Naseri, "A voronoi diagram-visibility graph potential field compound algorithm for robot path planning," *J. Field Robot.* **21**, 275–300 (2004).
7. M. Dorigo, V. Maniezzo and A. Colorni, "Ant system: Optimization by a colony of cooperating agents," *IEEE Trans. Syst. Man Cybern. Part B Cyber.* **26**, 29–41 (1996).
8. J. van Ast, R. Babuska and B. De Schutter, "Particle swarms in optimization and control," *IFAC Proc.* **41**, 5131–5136 (2008).
9. T. Ju, S. Liu, J. Yang and D. Sun, "Rapidly exploring random tree algorithm-based path planning for robot-aided optical manipulation of biological cells," *IEEE Trans. Automat. Sci. Eng.* **11**, 649–657 (2014).
10. A. Bircher, K. Alexis, U. Schwesinger, S. Omari, M. Burri and R. Siegwart, "An incremental sampling-based approach to inspection planning: The rapidly exploring random tree of trees," *Robotica.* **35**(6), 1327–1340 (2017).
11. O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Rob. Res.* **5**, 90–98 (1986).
12. Y. Koren and J. Borenstein, "Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation," *Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, Sacramento, CA, USA, vol. 2 (1991) pp. 1398–1404.
13. G. Li, Y. Tamura, A. Yamashita and H. Asama, "Effective improved artificial potential field-based regression search method for autonomous mobile robot path planning," *Int. J. Mechatr. Automat.* **3**(3), 141–170 (2013).
14. N. C. Tsourveloudis, K. P. Valavanis and T. Hebert, "Autonomous vehicle navigation utilizing electrostatic potential fields and fuzzy logic," *IEEE Trans. Robot. Automat.* **17**(4), 490–497 (2001).

https://doi.org/10.1017/S026357471900033X Published online by Cambridge University Press

15. A. Rajvanshi, S. Islam, H. Majid, I. Atawi, M. Biglerbegian and S. Mahmud, "An efficient potential-function based path planning algorithm for mobile robots in dynamic environments with moving targets," *Br. J. Appl. Sci. Technol.* **9**, 534–550 (2015).
16. A. A. Ahmed and T. Y. Abdalla, "Path planning of mobile robot by using modified optimized potential field method," *Int. J. Comput. Appl.* **113**(4), 6–10 (2015).
17. S. G. Loizou, "The navigation transformation," *IEEE Trans. Robot.* **33**(6), 1516–1523 (2017).
18. F. Xiao, L. Wang, J. Chen and Y. Gao, " Finite-time formation control for multi-agent systems," *Automatica.* **45**, 2605–2611 (2009).
19. M. Ji and M. Egerstedt, "Distributed coordination control of multi-agent systems while preserving connectedness," *IEEE Trans. Robot.* **23**(4), 693–703 (2007).
20. R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Trans. Automat. Contr.* **51**(3), 401–420 (2006).
21. Y. Kuriki and T. Namerikawa, "Formation control with collision avoidance for a multi-UAV system using decentralized MPC and consensus-based control," *SICE J. Control Measur. Syst. Integr.* **8**, 285–294 (2015).
22. I. Nagy, "Behaviour study of a multi-agent mobile robot system during potential field building," *Acta Polytech. Hung.* **6**, 111–136 (2009).
23. Y. Dai, Y. Kim, S. Wee, D. Lee and S. Lee, "A switching formation strategy for obstacle avoidance of a multi-robot system based on robot priority model," *ISA Trans.* **56**, 123–134 (2015).
24. P. Hou, H. Pan and C. Guo, "Simulation research for mobile robot path planning based on improved artificial potential field method recommended by the AsiaSim," *Int. J. Model. Simul. Sci. Comput.* **8**, 1750046 (2017).
25. X.-P. Hu, Z.-Y. Li and C. Jing, "A path planning method based on artificial potential field improved by potential ow theory," *2nd International Conference on Computer Science and Technology*, Guilin, Guangxi, China (2017) pp. 617–625.
26. S. A. Masoud and A. A. Masoud, "Constrained motion control using vector potential fields," *IEEE Trans. Syst. Man Cybern. A Syst. Hum.* **30**(3), 251–272 (2000).
27. J. Chen, D. Sun, J. Yang and H. Chen, "Leader-follower formation control of multiple nonholonomic mobile robots incorporating a receding-horizon scheme," *Int. J. Robot. Res.* **29**, 727–747 (2010).
28. G. Lee and D. Chwa, "Decentralized behavior-based formation control of multiple robots considering obstacle avoidance," *Intel. Serv. Robot.* **11**(1) 1–12 (2017).
29. Y. Abbasi, S. A. A. Moosavian and A. B. Novinzadeh, "Formation control of aerial robots using virtual structure and new fuzzy-based self-tuning synchronization," *Trans. Inst. Meas. Control* **39**(12), 1906–1919 (2017).
30. L. Dong, Y. Chen and X. Qu, "Formation control strategy for nonholonomic intelligent vehicles based on virtual structure and consensus approach," *Procedia Eng.* **137**, 415–424 (2016).
31. J. P. Desai, J. Ostrowski and V. Kumar, "Controlling Formations of Multiple Mobile Robots," *Proceedings of the IEEE Intemational Conference on Robotics and Automation*, Leuven (1998) pp. 2864–2869.
32. D. Shen, W. Sun and Z. Sun, "Adaptive PID formation control of nonholonomic robots without leader's velocity information," *ISA Trans.* **53**, 474–480 (2014).
33. V. Mariappan, M. Lee, J. Cho and J. Cha, "On board vision-based object tracking control stabilization using PID controller," *Int. J. Adv. Culture Technol.* **4**, 81–86 (2016).
34. J.-W. Jung, V. Q. Leu, T. D. Do, E.-K. Kim and H. H. Choi, "Adaptive PID speed control design for permanent magnet synchronous motor drives," *IEEE Trans. Power Electronics* **30**, 900–908 (2015).