

Formation control for autonomous robots with collision and obstacle avoidance using a rotational and repulsive force-based approach

Anh Duc Dang¹, Hung Manh La², Thang Nguyen³ 
and Joachim Horn¹

Abstract

In this article, we address a formation control problem for a group of autonomous robots to track a moving target in the presence of obstacles. In the proposed method, desired formations, which consist of virtual nodes arranged in specific shapes, are first generated. Then, autonomous robots are driven toward these virtual nodes without collisions with each other using a novel control scheme, which is based on artificial force fields. The convergence analysis is shown based on Lyapunov's stability. The novelty of the proposed approach lies in a new combination of rotational force field and repulsive force field to design a mechanism so that robots can avoid and escape complex obstacle shapes. The effectiveness of the proposed method is illustrated with numerical examples using V-shape and circular shape formations.

Keywords

Formation control, stability of a swarm, collision avoidance, vector fields, swarm intelligence

Date received: 20 December 2017; accepted: 7 April 2019

Topic: Mobile Robots and Multi-Robot Systems

Topic Editor: Lino Marques

Associate Editor: Veysel Gazi

Introduction

Research activities in multiagent systems have offered a wide range of applications in various areas such as physics, biology, cybernetics, and agriculture. In the field of multiagent systems, one of the most important problems is formation control, in which agents in the system are required to form desired shapes. Typical potential applications in formation control include search and rescue missions, forest fire detection and surveillance, source seeking, and so on. Autonomous robots of a multiagent system can be underwater vehicles,^{1,2} unmanned aerial vehicles,^{3,4} mobile sensor networks,^{5–12} rectangular agents,¹³ and non-holonomic mobile robots.^{14–18}

Centralized control protocols have been constructed based on the common assumption that the information of

all agents is available or the multiagent system possesses all-to-all communication. The drawbacks of the centralized

¹Institute of Control Engineering, University of the Federal Armed Forces Hamburg, Holstenhofweg, Hamburg, Germany

²Department of Computer Science and Engineering, Advanced Robotics and Automation Lab, University of Nevada, Reno, NV, USA

³Modeling Evolutionary Algorithms Simulation and Artificial Intelligence, Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City, Vietnam

Corresponding author:

Thang Tien Nguyen, Modeling Evolutionary Algorithms Simulation and Artificial Intelligence, Faculty of Electrical & Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City, 700000, Vietnam.

Email: nguyentienthang@tdtu.edu.vn



Table 1. Summary of related work in multiagent formation control.

| LCF | DFF | FS |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Flocking of mobile sensor networks, ⁵⁻⁷ flocking of multiagent systems, ⁸⁻¹¹ and adaptation and stability of a swarm in a dynamic environment. ^{12,29,30} | Formation control following a framework, ^{33,35,36} and formation control following dynamic region. ^{31,32} | Our article presents a new approach to multi-robot formation control following a desired shape formation (V-shape ^{37,38} or circular shape, ^{39,40} etc.) to track, and encircle a moving target under the effect of the dynamic environment. |
| LCF is mainly designed based on the random connections among the neighboring members in an α -lattice configuration. The convergence and adaptation of a swarm in a dynamic environment were verified. However, when we need a practical formation, such as V-shape, circular shape or linear formation, and so on. LCF method is still constrained. In this method, each robot's motion depends on the motion of its neighbors. | In the DFF method, all robots in the group move together within a given framework or region. They stay within a moving region and are able to adjust their formation by rotating and scaling during their movement. This method does not require specific orders or positions of all robots inside the given region. Each robot's motion depends on the motion of its neighbors and framework. | In our approach, the shape of the desired formation is easily designed. Robots are independently controlled by the attractive potential field from the virtual nodes in the desired formation. Moreover, the designed virtual nodes guarantee that the neighboring robots do not interact with each other. Hence, robots easily converge to these virtual nodes under the velocity matching. Using an added rotational vector field, robots can quickly escape convex and concave obstacles to continue to track the target. |

LCF: α -lattice connection formation; DFF: dynamic framework formation; FS: formation following desired shape.

communication control architect are inflexibility and large computational costs for each controller for each agent especially when the number of robots is large. In contrast, a distributed or decentralized control approach can provide more flexibility, easier implementation, and less computation loads as the controller of each agent only requires the information of its neighbor agents.^{6,7,18-21}

In formation control of a multiagent system, desired tasks of the system, such as tracking a moving target, are executed by the cooperation of a robot team. The formation control problem of autonomous robots is motivated by natural behaviors of creatures such as fish schooling, bird flocking, or ant swarming. Here, the members in the formation are required to collaborate with others to achieve common goals such as velocity matching and collision avoidance. The formation of a swarm of mobile robots can be generated and controlled by different methods in the literature. It is well-known that the artificial potential field method plays a pivotal role in controlling the coordination and the motion of a swarm so that its agents move toward target positions.²²⁻²⁸ It is well established that the potential field generates impulsive/attractive forces for mobile robots to avoid collision and maintain distances in coordination control problems.⁵ Many works on the formation control method using the random connections among neighboring members in a swarm as an α -lattice configuration were reported in a variety of papers.^{5-12,29,30} In this direction, the attractive/repulsive force fields are employed to link neighboring robots so that a robust formation without collisions is established. In a different approach, a dynamic framework was introduced by Hou et al.³¹ and Cheah et al.,³² in which the robots of a swarm can adapt

their formation by rotating and scaling during their movement. Similarly, the attractive/repulsive force fields were also utilized by Eren³³ to control robots to converge to given positions of a desired shape.

Although the artificial potential field has been regarded as a powerful tool for path planning of mobile robots, it still possesses some limitations due to local minimum problems. This can lead to a situation where a robot can get stuck if facing with an obstacle. In this context, the attractive force of the target and the repulsive force of the obstacles are equal and collinear but in an opposite direction. As a result, the total force on the robot is equal to zero, which traps the robot there. In addition, the traditional potential field method exhibits shortcomings in complex environments which contain convex and concave obstacle shapes, such as U-shaped obstacles or long walls and so on. When facing with these kinds of obstacles, robots can be trapped in them, and hence prevented from reaching targets.³⁴ The summary of related work is shown in Table 1. Recent research methods were proposed to address local minima avoidance^{41,42} that the reader can refer to.

In many studies, the dynamics of agents are modeled as single integrator,^{13,43,44} double integrators,^{5,7,10,45,46} and unicycles.^{47,48} In narrow space, the shapes of agents are taken into consideration for the formation control problem.^{13,44} In this article, a formation control problem for point-based agents with double-integrator dynamics is addressed. Theoretical development of the paper sets a framework, which can be employed to design a formation control system for various agents with more complicated dynamics and shapes. The particle model in this article has been extensively used in previous work. The work of Olfati-Saber⁵ presents a

theoretical framework for design and analysis of distributed flocking algorithms using the double-integrator dynamics for the dynamics of an agent. In applications, it is employed to model mobile robots that have omnidirectional motion capability, such as the Rovio mobile robots.^{7,10} It is also used in theoretical work for mobile sensor networks.^{45,46}

The main contributions of this article are in the following. The formation control algorithms are designed to drive multiple robots to converge to the desired positions to capture a moving target while avoiding collision with other agents and obstacles. However, the communication is still all-to-all since there is a need to compare the distances of agents to the target to choose the leader of the swarm. Once the leader is chosen, the control law of an agent is calculated based on the positions and velocities of its neighbor agents. In this case, the control algorithm is distributed, which helps save computational costs. The stability of the formation is maintained and collision avoidance among robots is obtained while the swarm tracks a moving target. In addition, the novelty of the article lies in the combination of the rotational and repulsive force fields, from which the obstacle avoidance control algorithm is constructed to drive robots to escape obstacles without collisions.

The structure of the article is organized as follows. In section *Formulation*, the problem formulation is presented. In section *Control algorithms*, the formation control algorithms are presented. Section *Case study* presents a case study for two typical examples of V-shape and circular shape formation control. Section *Simulation results* presents some simulation results. Finally, section *Conclusion* concludes the article with some remarks and future research topics.

Problem formulation

In this article, a swarm of N robots is considered with their mission of tracking a moving target in two-dimensional space. Let $p_i = (x_i, y_i)^T$ and $v_i = (v_{ix}, v_{iy})^T$ be the position and velocity vectors of robot i ($i = 1, 2, 3, \dots, N$), respectively. The dynamic model of robot i is

$$\begin{aligned} \dot{p}_i &= v_i \\ \dot{v}_i &= u_i \end{aligned} \quad (1)$$

for $i = 1, 2, 3, \dots, N$.

Conditions for a formation of autonomous robots are as follows:

- There are no collisions among robots.
- Robots must converge to the desired positions.
- Robots must avoid collision with obstacles.
- The formation must be maintained under the influence of the environment.

A desired formation of a multi-robot system consists of virtual nodes which will be occupied by the robots of the system. To facilitate the control design and analysis, we introduce the following definitions.

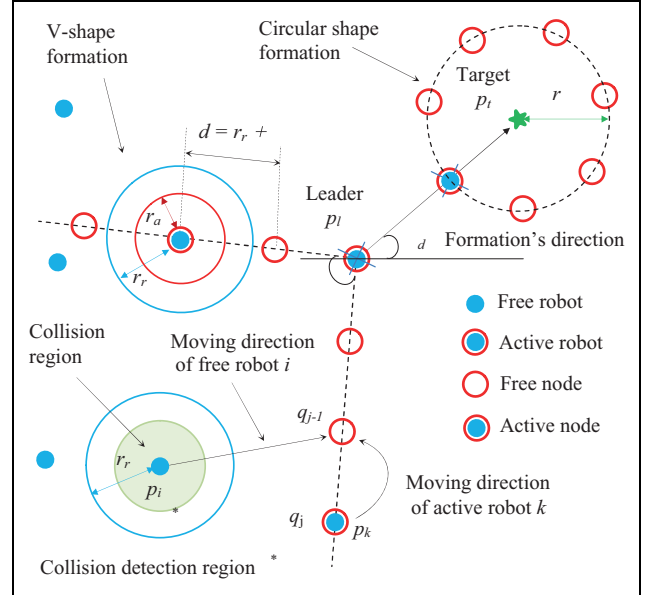


Figure 1. Description of the formation control method using the desired V-shape and circular shape structure. Agents use algorithm 1 to reach their virtual nodes to form the desired formations. The leader is selected by using algorithm 2.

Definition 1. In the desired formation, a virtual node j ($j = 1, 2, \dots, N$; $q_j = (x_j, y_j)^T$; $v_j = (v_{jx}, v_{jy})^T$) is a desired position for robot i , for which $\lim_{t \rightarrow \infty} \|p_i(t) - q_j(t)\| = 0$.

Definition 2. Let $d_{p_i q_j} < r_a$ be the distance of robot i to virtual node j ($j = 1, 2, 3, \dots, N$). If $d_{p_i q_j} < r_a$, where $r_a = d/2 - \lambda_a$, d is the distance between two neighbor virtual nodes, and λ_a is a positive constant, then robot i is an active robot. Otherwise, it is called a free robot. Here, r_a is denoted as the radius of the active circle surrounding the virtual node (see Figure 1).

Definition 3. Let l denote the index of the current leader of the swarm.

Definition 4. In the desired formation, virtual node j is called an active node if a robot i ($i = 1, 2, \dots, N$) lies in the active circle of the virtual node j (see Figure 1). Otherwise, it is called a free virtual node.

Definition 5. The desired V-shape formation consists of two line formations. These line formations are driven by a selected leader and connected by a formation angle ϕ . In each line of the formation, the virtual nodes are equidistant to its neighboring ones.

Definition 6. The desired circular shape formation is a circle in which all virtual nodes are distributed evenly on the circle and the target lies in the center of the circle.

Definition 7. Let $\mathcal{N}_i^k(t)$ be the set of the neighboring robots of the robot i at time t , such that

$$\mathcal{N}_i^k(t) = \{\forall k : d_i^k = \|p_i - p_k\| \leq r_r, k \in 1, \dots, N, k \neq i\} \quad (2)$$

where r_r is the repulsive radius surrounding each robot, and d_i^k is the Euclidean distance between robots k and i .

Definition 8. A set of the neighboring obstacles of robot i at time t is defined as

$$\mathcal{O}_i^o(t) = \{\forall k : d_{io_k} \leq r_\beta, k \in 1, \dots, M\} \quad (3)$$

where $r_\beta > 0$, M is the number of obstacles, and $d_{io_k} = \|p_i - p_{o_k}\|$ are the obstacle detection range and the Euclidean distance between robot i and obstacle o_k , respectively.

Control algorithms

This section presents control algorithms to guarantee that all robots will converge to their desired positions in the desired formation. While tracking a moving target, collision and obstacle avoidance of the swarm must be achieved.

Control architecture

To address our problem, we propose the control input u_i for each robot as follows

$$u_i = \begin{cases} u_i^t + u_i^o & \text{if robot } i \text{ is a leader} \\ u_i^f + u_i^o + u_i^c & \text{otherwise} \end{cases} \quad (4)$$

where u_i^f is used to control the formation connection, u_i^o drives robots to avoid obstacles, u_i^c helps robots avoid collision during the motion of the swarm, and u_i^t serves as a tracking controller. Hence, it can drive the leader of the swarm to move toward the target. In equation (4), when collision avoidance for robot i is active, other robots lie in its repulsive radius r_r , which activates u_i^c . Similarly, when an obstacle lies in the obstacle sensing range r_β , obstacle avoidance is active, leading to the activation of u_i^o . The stability analysis of the main results in this article will be based on Lyapunov's stability theorem.^{49,50}

Formation connection control algorithm

Initially, there exists an attractive force field for each virtual node j ($j = 1, 2, \dots, N$). The resulting attractive force fields drive free robots to move toward their desired positions. The free robots will then become active and will occupy these desired positions. The tracking control objective for each robot is to make the distance between robot i and virtual node j ($d_{p_i q_j} = \|p_i - q_j\|$) approach zero as fast as possible. In other words, $\lim_{t \rightarrow \infty} \|p_i - q_j\| = 0$ and $\lim_{t \rightarrow \infty} \|v_i - v_j\| = 0$. Based on the above argument, the formation control law for formation connection is presented in algorithm 1.

Algorithm 1. Reaching the desired positions at the virtual nodes in the desired formation.

Consider a robot i and virtual nodes j ($i, j = 1, \dots, N, i \neq j$). Determine the shortest distance from p_i to all the virtual nodes q_j and the scalar constant c_{ij}^f :

$$d_{ij_1} = \min\{d_{p_i q_j} = \|p_i - q_j\|, j = 1, \dots, N\}$$

$$c_{ij}^f = \begin{cases} 1, & \text{if } q_j \text{ is active} \\ 0, & \text{if } q_j \text{ is free.} \end{cases}$$

where j_1 is the virtual node whose distance to robot i is shortest.

if $d_{ij_1} \leq r_a$ & $c_{ij_1}^{p1} = 1$ **then**

$$u_i^f = -\kappa_i^{p1}(p_i - q_{j_1}) - \kappa_i^v(v_i - v_{j_1}) + \dot{v}_{j_1}$$

else

if $d_{ij_1} \leq r_a$ & $c_{i(j_1-1)} = 0$ **then**

$$u_i^f = -\kappa_i^{p2}(p_i - q_{j_1-1}) - \kappa_i^v(v_i - v_{j_1-1}) + \dot{v}_{j_1-1}$$

else

if $d_{ij_1} > r_a$ **then**

if $c_{ij_1} = 0$ **then**

$$u_i^f = -\kappa_i^{p3}(p_i - q_{j_1}) / \|p_i - q_{j_1}\| - \kappa_i^v(v_i - v_{j_1}) + \dot{v}_{j_1}$$

else

Determine the shortest distance from p_i to the free virtual nodes q_j of the desired formation:

$$d_{ij_2} = \min\{d_{p_i q_j} = \|p_i - q_j\|, c_{ij} = 0, j = 1, \dots, N, j \neq j_1\}$$

where j_2 is the virtual node in the set $\{j = 1, \dots, N, j \neq j_1\}$ whose distance to robot i is shortest.

$$u_i^f = -\kappa_i^{p3}(p_i - q_{j_2}) / \|p_i - q_{j_2}\| - \kappa_i^v(v_i - v_{j_2}) + \dot{v}_{j_2}.$$

end

end

end

end

In algorithm 1, κ_i^{p1} , κ_i^{p2} , κ_i^{p3} , and κ_i^v are positive constants; $(p_i - q_j)$ and $(v_i - v_j)$ are the relative position and velocity vectors between robot i and virtual node j , respectively. Two potential fields $f_{ij}^1 = -\kappa_i^{p3}(p_i - q_j) / \|p_i - q_j\|$ and $f_{ij}^2 = -\kappa_i^{p1}(p_i - q_j)$ are employed as the artificial attractive forces. Free robots are driven toward the desired

formation using the constant potential field f_{ij}^1 , while active robots are controlled to approach the virtual nodes using the linear potential field f_{ij}^2 . In addition, the component $-\kappa_i^v(v_i - v_j)$ serves as a damping term. Thus, using algorithm 1, the robots of the swarm can quickly converge to the desired positions, which are the virtual nodes of the desired formation. The stability of the formation is stated in the following.

Theorem 1. Consider the active robot i with its dynamic model (1) and control input u_i^f given in algorithm 1 at the active node j in the desired formation. If the velocity of node j is smaller than the maximum velocity of robot i , and node $j - 1$ is also active, then the system (1) will converge to the equilibrium state, at which $p_i = q_j$ and $v_i = v_j$ for all i and j .

Proof. In order to analyze the stability of the robot i at the active node j when the node $j - 1$ is also active, we rewrite the control law u_i^f as follows

$$u_i^f = f_{ij}^2 - \kappa_i^v(v_i - v_j) + \dot{v}_j \quad (5)$$

Consider the vector field $f_{ij}^2 = -\kappa_i^{p1}(p_i - p_j) = (P_i, Q_i, Z_i)^T$ where $P_i = \kappa_i^{p1}(x_i - x_j)$, $Q_i = \kappa_i^{p1}(y_i - y_j)$, and $Z_i = 0$. According to Freeman and Kokotovic,⁴⁹ we obtain

$$\text{rot}(f_{2i}^j) = \left(\frac{\partial Z_i}{\partial y_i} - \frac{\partial Q_i}{\partial z_i}, \frac{\partial P_i}{\partial z_i} - \frac{\partial Z_i}{\partial x_i}, \frac{\partial Q_i}{\partial x_i} - \frac{\partial P_i}{\partial y_i} \right)^T = 0 \quad (6)$$

Equation (6) shows that the vector field f_{ij}^2 is irrotational. Consider the scale function

$$V_{ij} = \frac{1}{2} \kappa_i^{p1} (p_i - q_j)(p_i - q_j)^T \quad (7)$$

Taking the negative gradient of the function V_{ij}^f , we obtain

$$\begin{aligned} -\nabla V_{ij} &= -\nabla \left(\frac{1}{2} \kappa_i^{p1} (p_i - q_j)(p_i - q_j)^T \right) \\ &= -\nabla \left(\frac{1}{2} \kappa_i^{p1} ((x_i - x_j)^2 + (y_i - y_j)^2) \right) \\ &= \left(-\frac{1}{2} \kappa_i^{p1} \frac{\partial}{\partial x_i} ((x_i - x_j)^2 + (y_i - y_j)^2)^T \right. \\ &= \left. \left(-\frac{1}{2} \kappa_i^{p1} \frac{\partial}{\partial y_i} ((x_i - x_j)^2 + (y_i - y_j)^2) \right)^T \right. \\ &= \left(-\kappa_i^{p1} (x_i - x_j) \right)^T \\ &= \left(-\kappa_i^{p1} (y_i - y_j) \right)^T = -\kappa_i^{p1} (p_i - p_j) \\ &= f_{ij}^2 \end{aligned} \quad (8)$$

Equations (6) and (8) show that the vector field f_{ij}^2 is a potential field, and its potential function is V_{ij}^f .

Let $\bar{x}_1 = p_i - q_j$ and $\bar{x}_2 = v_i - v_j$ be the relative position and velocity of the robot i and node j , respectively. The error dynamic model of the system is given as

$$\dot{\bar{x}}_1 = \bar{x}_2 \quad (9)$$

$$\dot{\bar{x}}_2 = u_i^f - \dot{v}_j, i, j = 1, 2, \dots, N \quad (10)$$

From equations (5) and (9),

$$\dot{\bar{x}} = A\bar{x} \quad (11)$$

where $\bar{x} = (\bar{x}_1, \bar{x}_2)^T$ and

$$A = \begin{bmatrix} O_2 & I_2 \\ -\kappa_{ip1}^j I_2 & -\kappa_i^v I_2 \end{bmatrix} \quad (12)$$

where O_2 is the zero matrix of 2 and I_2 is the identity matrix of 2. It can easily be shown that the eigenvalues of matrix A are the roots of the polynomial $s^2 + \kappa_i^v s + \kappa_i^v$, which have negative real parts since κ_i^{p1} is positive. Hence, the system (11) is asymptotically stable.⁵⁰ \square

Remark 1. It is well established that the Lyapunov theory is employed for the stability proof of formation and flock control of multiagent systems.⁵ For more information about the Lyapunov theory, the reader is referred to the work of Freeman and Kokotovic⁴⁹ and Khalil.⁵⁰

Remark 2. Figure 1 illustrates a desired formation (V-shape or circular shape formation) of N virtual nodes. The desired formation contains a desired position for each robot. Algorithm 1 allows each free robot i to firstly seek its closest free virtual node (for instance, j) so that it will be active at this virtual node. If the position of an active robot i is still not at the desired one (e.g. robot k in Figure 1 with $k = 1, 2, \dots, N, k \neq l$), then it will automatically move to virtual node ($j - 1$) until it occupies the desired position.

In order to allow robot i to approach node j as fast as possible, we use the attractive force from node j , which is proportional to its velocity. This means that the factor κ_i^{p1} in algorithm 1 depends on the velocity v_j . This factor is given as

$$\kappa_i^{p1} = \kappa_i^{pd} + \varepsilon_1 \|v_j\| \quad (13)$$

where κ_i^{pd} and ε_1 are positive.

Collision avoidance control algorithm

This section presents a method for the collision avoidance among the robots during their movement based on the artificial repulsive potential field. The potential field has been widely used in addressing flocking and formation control of multiagent systems.^{5,9,22-24} In general, a potential function $\xi(\cdot)$, which depends on the distance among agents or between agents and obstacles, is constructed. A typical feedback

control law consists of two parts: the gradient of the potential function and the velocity consensus term.^{5,9,22–24}

In order to avoid the collision between robots i and k ($i, k = 1, 2, \dots, N; i \neq k \neq l$), the local repulsive force field is created surrounding each robot within the repulsive radius r_r as

$$f_{ik} = \left(\left(\frac{1}{d_{ik}} - \frac{1}{r_r} \right) \frac{\kappa_{ik}^{c1}}{(d_{ik})^2} - \kappa_{ik}^{c2} (d_{ik} - r_r) \right) c_{ik}^c n_{ik} \quad (14)$$

where positive constants κ_{ik}^{c1} and κ_{ik}^{c2} are employed to deal with fast interaction. The unit vector from robots k to i is described as $n_{ik} = (p_i - p_k) / \|p_i - p_k\|$. The scalar c_{ik}^c is defined as N

$$c_{ik}^c = \begin{cases} 1, & \text{if } k \in \mathcal{N}_i^k(t) \\ 0, & \text{otherwise} \end{cases} \quad (15)$$

The algorithm for the collision avoidance is built based on the repulsive vector field (14) combined with the relative velocity vector $\kappa_{ik}^c (v_i - v_k)$ between robots k and i as follows

$$u_i^c = \sum_{k=1, k \neq i}^N (f_{ik} - c_{ik}^c \kappa_{ik}^c (v_i - v_k)) \quad (16)$$

The controller (16) reveals that neighboring robots are controlled to leave each other when distances among them are too close. As in the study of Olfati-Saber,⁵ we employ repulsive fields to avoid collision among agents. The analysis is quite similar to that in the study of Olfati-Saber.⁵ Therefore, our control mechanism guarantees that collision avoidance in the swarm is achieved.

Obstacle avoidance control algorithm

This section presents a control algorithm for robots to pass through M obstacles. The obstacle avoidance control algorithm for each member robot i ($i = 1, 2, \dots, N$) is designed as follows

$$u_i^o = \sum_{h=1}^M (f_i^{op} + f_i^{or} + \kappa_{ih}^o c_{ih}^o (v_i - v_{o_h})) \quad (17)$$

where relative velocity vector $(v_i - v_{o_h})$ between robot i and its neighboring obstacle o_h is utilized as a damping term with damping scaling factor κ_{ih}^o , and the scalar c_{ih}^o is defined as

$$c_{ih}^o = \begin{cases} 1, & \text{if } h \in \mathcal{O}_i^o(t) \\ 0, & \text{otherwise} \end{cases} \quad (18)$$

The repulsive force field f_i^{op} is created to drive robot i to move away from its neighboring obstacle (see Figure 2). This force field is designed as

$$f_i^{op} = \left(\left(\frac{1}{d_{io_h}} - \frac{1}{r_\beta} \right) \frac{\kappa_i^{o1}}{(d_{io_h})^2} - \kappa_i^{o2} (d_{io_h} - r_\beta) \right) c_{io_h}^o n_{io_h} \quad (19)$$

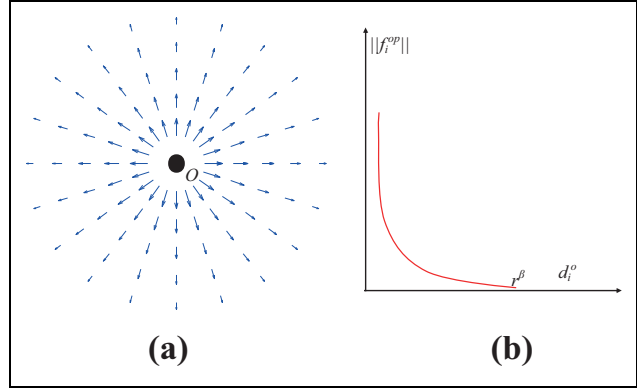


Figure 2. The description of the repulsive force field f_i^{op} surrounding the neighboring obstacle o of the robot i (a), and its amplitude $\|f_i^{op}\|$ (b).

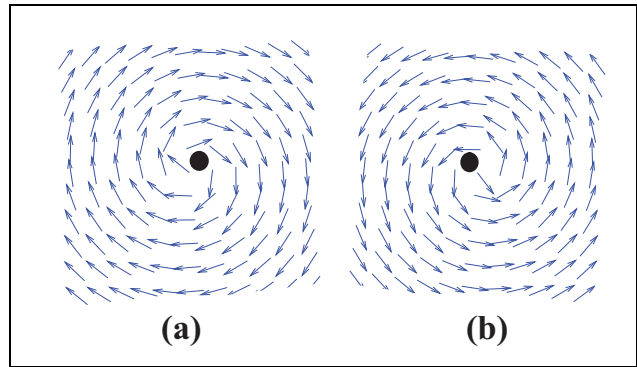


Figure 3. The clockwise rotational force field (a) and the counter clockwise rotational force field (b).

where positive factors κ_i^{o1} and κ_i^{o2} are used to control the fast obstacle avoidance, r_β is the obstacle detection range defined in definition 8, and $n_{io_h} = (p_i - p_{o_h}) / \|p_i - p_{o_h}\|$ is a unit vector.

In control law (17), the rotational force field (see Figure 3) is added to combine with the repulsive force (see Figure 2) to drive robot i to quickly escape its neighboring obstacle. While the potential force field is used to enable it to avoid collision with its neighboring obstacle, the rotational force field is used to solve the local minimum problems; for instance, the robot meets a trapping point, at which the repulsive force of the obstacles and the attractive force of the target are balanced. Under the effect of the added rotational force field, the robot always escapes this trapping point. Furthermore, when robot is trapped in complex obstacles (e.g. U-shape or long wall), the rotational vector field will help it find a new path to escape these obstacles. The direction of the rotational force can be clockwise or counter clockwise (see Figure 3). Hence, this rotational force is built as

$$f_i^{or} = w_{io_h} c_{io_h} n_i^{or} \quad (20)$$

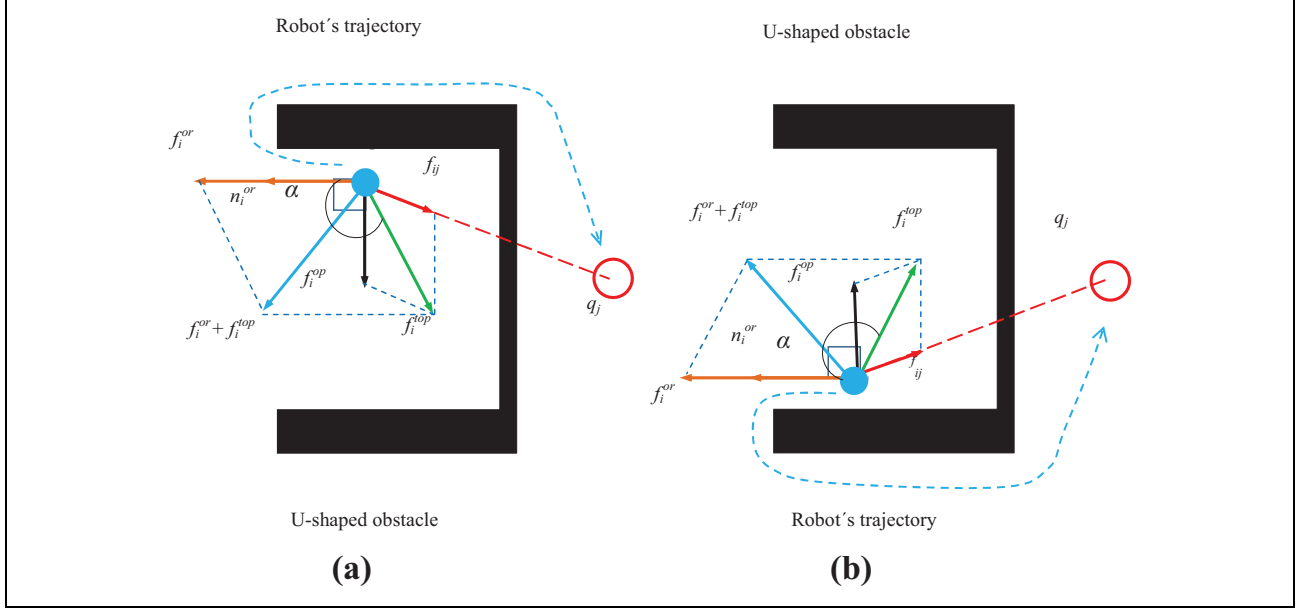


Figure 4. The description of the obstacle escape of a robot i while reaching a virtual node j : clockwise (a) and counter clockwise (b).

where the unit vector n_i^{or} is given as

$$n_i^{or} = c_i^{or} ((y_i - y_{o_h})/d_{io_h}, -(x_i - x_{o_h})/d_{io_h})^T \quad (21)$$

where the scalar c_i^{or} is used to define the direction for the rotational force: the rotational force is clockwise if $c_i^{or} = 1$ and counter clockwise if $c_i^{or} = -1$. Now, we consider the relationship between this unit vector and the vector $p_i - p_{o_h}$.

Let σ be the angle between these vectors. Then, we have

$$\cos \sigma = \frac{c_i^{or} ((x_i - x_{o_h})(y_i - y_{o_h}) - (y_i - y_{o_h})(x_i - x_{o_h}))}{(d_{io_h})^2} = 0 \quad (22)$$

Equation (22) shows that the unit vector is always perpendicular to vector $(p_i - p_{o_h})$. The positive gain factor w_{io_h} in equation (20) is used as a control element to drive robots to quickly escape obstacles. Therefore, this control element is designed such that the total force on robots always has the direction in the selected rotational direction. This control element is given as follows

$$w_{io_h} = (1 + c)(\|f_i^{top}\| + \lambda_o) \quad (23)$$

where λ_o is a positive factor and the force f_i^{top} is described as

$$f_i^{top} = f_{ij} + f_i^{op} \quad (24)$$

where f_i^{op} is the repulsive force from the neighboring obstacles of robot i , and f_{ij} is the attractive force from the virtual node j in the desired formation (see algorithm 1). The constant c in equation (23), which depends on angle α between

the sum vector f_i^{top} and the unit vector n_i^{or} (see Figure 4), is described as follows

$$c = \begin{cases} c_1, & \text{if } \alpha < \pi/2 \\ c_2, & \text{otherwise} \end{cases} \quad (25)$$

where two constants c_1 and c_2 can be chosen as $-1 < c_1$, $0 < c_2$, and $c_1 < c_2$. Equation (23) guarantees that robot i always moves in the direction of the rotational force f_i^{or} . Hence, it can easily escape the obstacles in order to continue with its swarm to track the moving target.

The total vector for the control law for agent i is $f^s = f_i^{or} + f_i^{top}$. From equations (20) and (23), $\|f_i^{or}\| = w_{io_h}$. It is clear that when $\alpha < \pi/2$, the angle between f^s and n_{io_h} is smaller than $\pi/2$. When $\alpha \geq \pi/2$, $\|f_i^{or}\| = w_{io_h} > \|f_i^{top}\|$. This implies that the angle between f^s and n_{io_h} is smaller than $\alpha/2$, which is smaller than $\pi/2$. As a result, the total vector drives the agent to the direction to escape the obstacle (see Figure 4).

The obstacle avoidance analysis of the proposed scheme follows a similar approach as in theorem 6 in the technical report by Olfati-Saber.⁵¹ Unlike the method of Olfati-Saber,⁵¹ in our method, when an agent meets an obstacle, a rotational force field is generated which changes total force field. Hence, the direction of the agent is turned around the obstacle.

We summarize the above analysis in the following theorem.

Theorem 2. Consider the active robot i with its dynamic model (1). If robot i detects an obstacle in its obstacle

detection range as in Figure 4, the control law in equation (17) will drive robot i to escape the obstacle.

Remark 3. The protocol for collision avoidance for the agents of the system in subsection Control architecture employs a repulsive force in equation (14). Here, the shapes of agents are simplified to be points or circles. In contrast, the obstacle avoidance in this section is realized using repulsive and rotational forces in equations (19) and (20) for a class of complex obstacles (which can be large and nonconvex). This enables agents to escape the obstacles effectively. The constraint posed on a nonconvex obstacle is that the radius of the osculating circle of the obstacle is larger than r_β , which is the obstacle detection range. The rotational field is only activated if the obstacle is in the obstacle detection range of the agent. However, if there are many rotational forces due to many obstacles, the control algorithm of the agent will have the rotational force from the obstacle with the highest priority.

Remark 4. The complex obstacle used in our analysis is in a U-shape, which is nonconvex (see Figure 4). A rigorous study would be conducted to see whether our obstacle avoidance approach can work with a more general class of complex obstacles in the future.

Target tracking control algorithm

Firstly, a robot, which is closest to the target, is selected as the leader in order to generate the desired formation. Then, this leader leads its formation to track a moving target. When the leader encounters a risk, such as it is broken or trapped in obstacles, it must transfer its leadership to another, and becomes a free member robot in the swarm. The leader is selected as in algorithm 2.

The target tracking controller, which is designed based on the relative position between the leader and the target, has to guarantee that the formation's motion is always driven toward the target. As introduced above, the V-shape and circular shape formations are utilized to track and encircle a moving target. Hence, the tracking task is to make the distance between the leader and the target $d_l^t = \|p_l - p_t\|$, approaching the radius of the desired circular formation r^τ as fast as possible. This means that $\lim_{t \rightarrow \infty} \|p_l - p_t\| = 0$ and $\lim_{t \rightarrow \infty} \|v_l - v_t\| = 0$. Based on the above analysis, the control law for the target tracking is proposed as follows

$$u_l^t = f_l^t - \kappa_l^v(v_l - v_t) + \dot{v}_t \quad (26)$$

where κ_l^f and \dot{v}_t are the positive constant and the acceleration of the target, respectively, and $(v_l - v_t)$ is the relative velocity vector between the leader and the target.

The potential field from the target is used to drive the leader moving toward the target, and is given by

Algorithm 2. Leader selection.

Update data: The actual position of robots p_i ($i = 1, \dots, N, i \neq l$), obstacle's information, the target's position p_t , the actual position of the leader ($p_\xi = p_l$).

if time $t = 0$ (at initial time) **then**

Compute the shortest distance from the robot p_i to the target p_t in order to determine the leader as

$$d_{i_1}^t = \min\{\|p_i - p_t\|, i = 1, \dots, N\}; p_l = p_{i_1}$$

else

if the current leader meets obstacle or is broken then the leadership is transferred to a member that is free and has the closest distance to the target **then**

$$d_{i_2}^t = \min\{\|p_i - p_t\|, i = 1, \dots, N, i \neq \xi, \text{free}\}$$

$$p_l = p_{i_2}$$

else

Maintain the leadership of the current leader:

$$p_l = p_\xi.$$

end

end

$$f_l^t = \begin{cases} \left(\left(\frac{1}{d_l^t} - \frac{1}{r^\tau} \right) \frac{\kappa_l^{f2}}{(d_l^t)^2} - \kappa_l^{f3} \frac{(d_l^t - r^\tau)}{r^\tau - r^\tau} \right) n_l^t, & \text{if } d_l^t < r^t \\ -\kappa_l^{f1} n_l^t, & \text{otherwise} \end{cases} \quad (27)$$

where κ_l^{f2} and κ_l^{f3} are positive constants, and r_t and r_τ are the target approaching radius and the desired radius of the circular formation ($r_{\min} < r^\tau < r^t$), respectively. Here, $\kappa_l^{f1} = \kappa_l^{fd1} + \varepsilon_2 \|v_t\|$ where κ_l^{fd1} and ε_2 are positive constants. The unit vector along the line connection from the target to the leader is computed as $n_l^t = (p_l - p_t) / \|p_l - p_t\|$. In equation (27), the constant attractive force $f_l^{f2} = -\kappa_l^{f1} n_l^t$ is used to track the target when $d_l^t > r^t$. On the other hand, the attractive/repulsive force field surrounding the equilibrium position, at which $\|p_l - p_t\| = r^\tau$ and $v_l - v_t = 0$, is employed to encircle the target when $d_l^t \leq r^t$. Hence, using this combined vector field, the leader can easily approach the target at the equilibrium position.

Theorem 3. Consider the leader l described by model (1) and governed by control law (26) when $d_l^t \leq r^t$. If the velocity of the target is smaller than the maximum velocity of the leader, then the system (1) will converge to the equilibrium state, at which $v_l = v_t$ and $(p_l - p_t) = \tau(p_l - p_t) / \|p_l - p_t\|$.

Proof. Firstly, we consider the vector field $f_{l1}^t = K_l n_l^t$ in equation (27), where $K_l = (\kappa_l^{f2}/d_l^t - \kappa_l^{f3}/r^\tau)(d_l^t)^2 -$

$\kappa_l^{t1}(d_l^t - r^\tau)/(r^t - r^\tau)$. Let $P_l = K_l(x_l - x_t)/\|p_l - p_t\|$, $Q_l = K_l(y_l - y_t)/\|p_l - p_t\|$, and $Q_l = 0$. Then, we have

$$\text{rot}(f_{1l}^t) = \left(\frac{\partial Z_l}{\partial y_l} - \frac{\partial Q_l}{\partial z_l}, \frac{\partial P_l}{\partial z_l} - \frac{\partial Z_l}{\partial x_l}, \frac{\partial Q_l}{\partial x_l} - \frac{\partial P_l}{\partial y_l} \right)^T = 0 \quad (28)$$

Equation (28) shows that the vector field f_{1l}^t is not rotational.

Consider the scale function

$$V_l^t = \frac{1}{2} \left(\kappa_l^{t2} \left(\frac{1}{d_l^t} - \frac{1}{r^\tau} \right)^2 + \frac{\kappa_l^{t1}(d_l^t - r^\tau)^2}{(r^t - r^\tau)} \right) \quad (29)$$

Taking the negative gradient of V_l^t , we obtain

$$\begin{aligned} -\nabla V_l^t &= -\nabla \left(\frac{\kappa_l^{t2}}{2} \left(\frac{1}{d_l^t} - \frac{1}{r^\tau} \right)^2 + \frac{\kappa_l^{t1}(d_l^t - r^\tau)^2}{2(r^t - r^\tau)} \right) \\ &= - \left(\frac{\kappa_l^{t2}}{d_l^t} - \frac{\kappa_l^{t2}}{r^\tau} \right) \nabla \left(\frac{1}{d_l^t} - \frac{1}{r^\tau} \right) \\ &\quad - \frac{\kappa_l^{t1}(d_l^t - r^\tau)}{(r^t - r^\tau)} \nabla (d_l^t - r^\tau) \\ &= \left(\frac{\kappa_l^{t2}}{d_l^t} - \frac{\kappa_l^{t2}}{r^\tau} \frac{1}{(d_l^t)^2} \nabla d_l^t - \frac{\kappa_l^{t1}(d_l^t - r^\tau)}{(r^t - r^\tau)} \right) \nabla d_l^t \\ &= \left(\left(\frac{\kappa_l^{t2}}{d_l^t} - \frac{\kappa_l^{t2}}{r^\tau} \right) \frac{1}{(d_l^t)^2} - \frac{\kappa_l^{t1}(d_l^t - r^\tau)}{(r^t - r^\tau)} \right) \nabla d_l^t \\ &= \left(\left(\frac{\kappa_l^{t2}}{d_l^t} - \frac{\kappa_l^{t2}}{r^\tau} \right) \frac{1}{(d_l^t)^2} - \frac{\kappa_l^{t1}(d_l^t - r^\tau)}{(r^t - r^\tau)} \right) \frac{(p_l - p_t)}{\|p_l - p_t\|} \\ &= f_{1l}^t \end{aligned} \quad (30)$$

Hence, equations (28) and (30) show that the vector field f_{1l}^t is also a potential field, and its potential function is V_l^t .

In order to analyze the stability of the leader at the equilibrium position, at which $\|p_l - p_t\| = r^\tau$ and $v_l - v_t = 0$, let $\tilde{x}_1 = p_l - p_t$ and $\tilde{x}_2 = v_l - v_t$. Then, the error dynamic of the system is described as

$$\dot{\tilde{x}}_1 = \tilde{x}_2 \quad (31)$$

$$\dot{\tilde{x}}_2 = \dot{v}_l - \dot{v}_t = u_l^t - \dot{v}_t \quad (32)$$

From equations (26) and (31), we have

$$\dot{\tilde{x}}_1 = \tilde{x}_2 \quad (33)$$

$$\dot{\tilde{x}}_2 = \nabla V_l^t - \kappa_l^{tv} \tilde{x}_2 \quad (34)$$

To analyze the stability of the system (33), we choose the following Lyapunov function candidate

$$V_\tau = V_l^t + \frac{1}{2} \tilde{x}_2^T \tilde{x}_2 \quad (35)$$

Taking the time derivative of equation (35) along the trajectory of the system (33), we obtain

$$\dot{V}_\tau = \nabla V_l^t \dot{\tilde{x}}_1 + \tilde{x}_2^T \dot{\tilde{x}}_2 = \kappa_l^{tv} \tilde{x}_2^T \tilde{x}_2 \leq 0 \quad (36)$$

According to LaSalle's theorem, $\tilde{x}_2 \rightarrow 0$.⁵⁰ It can easily be shown that \tilde{x}_2 is uniformly continuous. Hence, according to Barbalat's lemma, $\dot{\tilde{x}}_2 \rightarrow 0$.⁵² This implies from equation (33) that $\nabla V_l^t \rightarrow 0$ or $f_{1l}^t \rightarrow 0$. Hence, $d_l^t \rightarrow r^\tau$ due to the description of the potential force function in equation (27). So, the system (33) is stable at the equilibrium position when using the control law (26). In other words, using the controller (26), the leader will converge to the equilibrium position, at which the distance between the leader and the target is equal to the radius of the desired circular formation. \square

Remark 5. The formation of the swarm changes with respect to the relative position between the leader and the target. Initially, a robot will be chosen as a leader to lead its formation toward the target if it is closest to the target. During the motion of the swarm, if the current leader faces any risk (for instance, it is broken from the formation or hindered by the environment), then a new leader is nominated. Algorithm 2 guarantees that this new leader will reorganize the formation and continue to lead the swarm to track the target.

Remark 6. Obstacles in the environment can be sensed by several methods. Onboard sensors of each agent can detect them. In addition, hybrid approaches can employ a group of aerial drones to capture the images of obstacles and agents and transfer data to the agents.^{19,20}

Remark 7. The paper⁵³ addressed the swarm tracking problem to capture a moving target in a specific formation using artificial potentials and sliding mode control. The dynamics used in this work are fully actuated. The artificial potentials are used for the formation and tracking control goals. Sliding techniques are used to deal with uncertainty, which exhibits robustness of the proposed control method. The control framework by Yao et al.⁵³ was then applied for agents with nonholonomic dynamics by Gazi et al.⁵⁴ Our control strategy shares the same principle as in the studies of Yao et al.⁵³ and Gazi et al.⁵⁴ where artificial potentials are employed for formation and target tracking control. On the one hand, the work of Yao et al.⁵³ and Gazi et al.⁵⁴ considers robustness issues, which we do not address in this article. On the other hand, our article deals with collision and obstacle avoidance, which are not included in the article.^{53,54}

Case study

The proposed algorithms as discussed in the previous section can allow the robots to follow predefined formations such as V-shape, circular, or line shape. Due to their

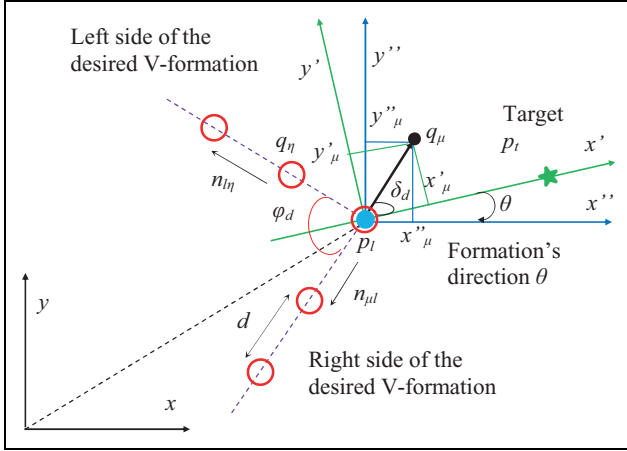


Figure 5. The description of the method to build the V-shape desired formation.

similarity, we just present two cases of V-shape and circular shape. Let $p_l = (x_l, y_l)^T$ and $p_t = (x_t, y_t)^T$ be the positions of the leader and the target, respectively. The relative position vector between the leader and the target is $(p_l - p_t) = (x_l - x_t, y_l - y_t)^T$, and the distance between them is determined as $d_l^2 = \sqrt{(x_l - x_t)^2 + (y_l - y_t)^2}$.

V-shape formation

Firstly, in order to build the V-shape desired formation, we design the right side of this V-shape based on the desired formation angle ϕ_d and the relative position between the leader and the target. As presented in Figure 5, the coordinates of the base node q_μ on the coordinate system $x'y'$ ($q'_\mu = (x'_\mu, y'_\mu)^T$) are determined as follows

$$\begin{pmatrix} x'_\mu \\ y'_\mu \end{pmatrix} = \|q_\mu - p_l\| \begin{pmatrix} \cos \delta_d \\ \sin \delta_d \end{pmatrix} \quad (37)$$

where the angle $\delta_d = \angle(q_\mu - p_l), (p_t - p_l)$ is equal to $\phi_d/2$ and $\phi_d > 0$ is the desired formation angle (see Figure 5). The base node q_μ is employed to generate virtual nodes on the right side of the V-shape. It lies on the line which deviates with an angle of δ_d from the line connecting the leader and the target. The distance between the base node q_μ and the leader is equal to the distance of consecutive virtual nodes on the right wing of the V-shape. By rotating and translating equation (37) according to coordinate systems $x''y''$ and xy ,⁵⁵ we obtain the position of the desired node q_μ on the coordinate system xy as follows

$$q_\mu = p_l + Rq'_\mu \quad (38)$$

From base node q_μ and the position of the leader, we determine a unit vector along the line connecting from q_μ to p_l as $n_{\mu l} = (p_l - q_\mu) / \|p_l - q_\mu\|$. The rotational

matrix R , which depends on the rotational angle θ , is determined as

$$R = \begin{cases} \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}, & \text{if } \theta \text{ rotates clockwise} \\ \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix}, & \text{otherwise} \end{cases} \quad (39)$$

Now, a virtual node j ($d_j^l = \|p_l - q_j\| = jd$; $q_j = (x_j, y_j)^T$; $v_j = (v_{jx}, v_{jy})^T$; $j = 1, 2, \dots, N^*$; $N^* \in \mathbb{R}$) is determined by the unit vector $n_{\mu l}$ as

$$(q_j - p_l) = jdn_{\mu l} \quad (40)$$

Here, d is the distance between any two consecutive virtual nodes. Note that $\|p_l - q_\mu\| = d$. Substituting $n_{\mu l} = (p_l - q_\mu) / \|p_l - q_\mu\|$ into equation (40), we obtain

$$q_j = (1 + j)p_l - jq_\mu \quad (41)$$

Equation (41) can be rewritten as

$$\begin{pmatrix} x_j \\ y_j \end{pmatrix} = \begin{pmatrix} (1 + j)x_l \\ (1 + j)y_l \end{pmatrix} - \begin{pmatrix} jx_\mu \\ jy_\mu \end{pmatrix} \quad (42)$$

Equation (42) shows that when j changes from $j = 1$ to $j = N^*$, we obtain the formation of the N^* virtual nodes. These nodes lie on a line connecting p_l and q_μ and are equally spaced (the right side of the desired V-shape formation; see Figure 5).

Similarly, the virtual node j on the left side of the V-shape is designed as

$$(q_j - q_l) = jdn_{l\eta} \quad (43)$$

where $n_{l\eta} = (q_\eta - p_l) / \|q_\eta - p_l\|$ is a unit vector. Hence,

$$q_j = (1 - j)p_l + jq_\eta \quad (44)$$

where q_η is the position of the base node on the line which deviates from the line through the leader and the target with an angle of $(\pi - \delta_d)$ (see Figure 5). The base node q_η of the left side is similar to the base node q_μ of the right side of the V-shape. The distance between the base node q_η and the leader is equal to the distance of consecutive virtual nodes on the left wing of the V-shape. Similar to equation (38), this base node is determined as

$$q_\eta = p_l + Rq'_\eta \quad (45)$$

Using equation (45), we obtain the formation of the virtual node j . They lie on the line through p_l and q_η and are equally spaced (the left side of the V-shape; see Figure 5). Finally, the law to generate the desired V-shape formation of the virtual nodes j ($j = 1, 2, \dots, N$) is proposed as

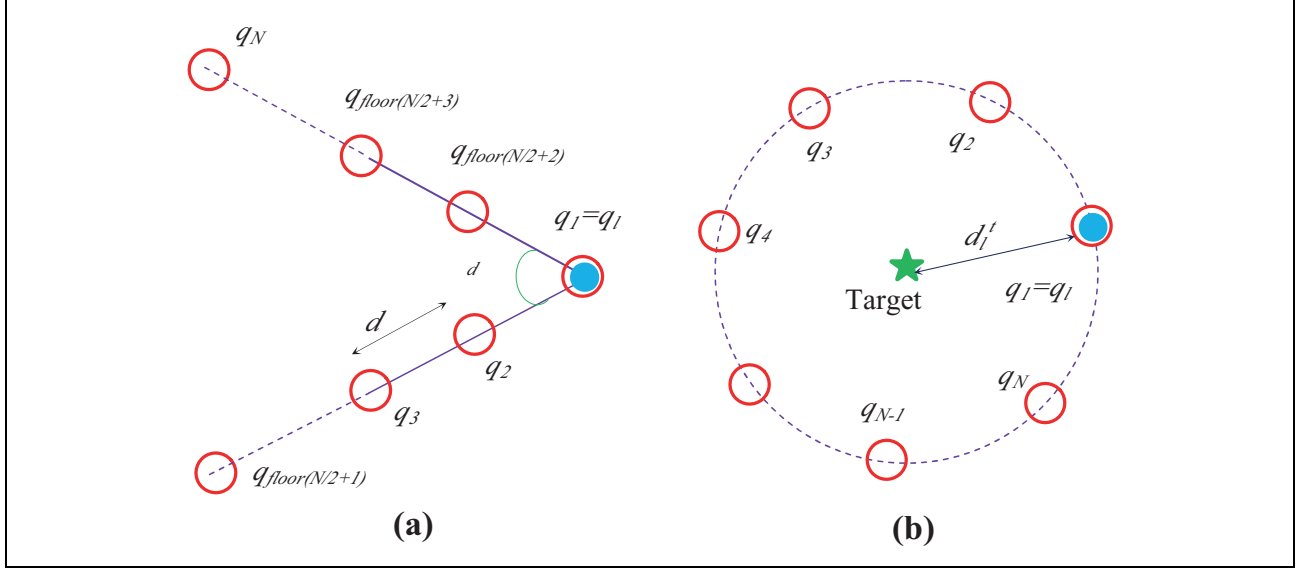


Figure 6. The description of the distributed virtual node j in the desired V-shape formation (a) and in the desired circular shape formation (b).

$$q_j = \begin{cases} (1 + \xi_1)p_l - \xi_1 q_\mu, & \text{if } j \leq N/2 + 1 \\ (1 - \xi_2)p_l + \xi_2 q_\mu, & \text{otherwise} \end{cases} \quad (46)$$

where $\xi_1 = j - 1$ and $\xi_2 = j - 1 - \text{floor}(N/2)$ are positive factors. Using equation (46), the virtual nodes j ($j = 1, 2, \dots, N$) will be evenly distributed to both sides of the leader. Hence, we obtain a V-shape formation due to these desired virtual nodes and a constant formation angle $\phi_d = 2\delta_d$, as shown in Figure 6(a). In some cases, such as under the influence of an environmental factor (noises, wind, obstacle avoidance, etc.), the formation angle needs to be changed to adapt to the effect of this environment. Thus, the formation angle is chosen as

$$\phi(t) = \phi_d + \varepsilon_3 \phi(t) \quad (47)$$

where ε_3 is a positive constant, and $\phi(t)$ is used as a sensing function that decides the formation angle $\phi(t)$. However, this formation angle $\phi(t)$ must guarantee that there are no collisions among the members in the formation. In other words, it depends on the repulsive radius of each robot. Hence, the smallest formation angle is computed as $\phi_{\text{dmin}} = \arccos(1 - r_r^2/2d^2)$.

Circular formation

As presented above, the circular formation is used to encircle the moving target when the distance between the leader and the target is shorter than the target approaching radius $d_l^t = \|p_l - p_t\| \leq r^t$. Hence, this desired formation is designed based on the relative position between the target and the leader, as shown in Figure 6.

The position of the virtual nodes j ($j = 1, 2, \dots, N$) on the circle, whose central point is at the target's position p_t and whose radius is $d_l^t = \|p_l - p_t\|$, is computed as

$$q_j = p_l + Rq_j', j = 1, 2, \dots, N \quad (48)$$

The position of the virtual node j on the coordinate system $x'y'$ ($q_j' = (x_j', y_j')^T$) is computed as

$$\begin{pmatrix} x_j' \\ y_j' \end{pmatrix} = d_l^t \begin{pmatrix} \cos(2j\pi/N) \\ \sin(2j\pi/N) \end{pmatrix}, j = 1, 2, \dots, N \quad (49)$$

Let the virtual node owned by the leader be the first position in the circular desired formation. Substituting equation (49) into equation (48), we have the circular formation of the virtual node j as follows

$$\begin{pmatrix} x_j \\ y_j \end{pmatrix} = \begin{pmatrix} x_l \\ y_l \end{pmatrix} + R d_l^t \begin{pmatrix} \cos(2\zeta\pi/N) \\ \sin(2\zeta\pi/N) \end{pmatrix}, j = 1, 2, \dots, N; \zeta = j - 1 \quad (50)$$

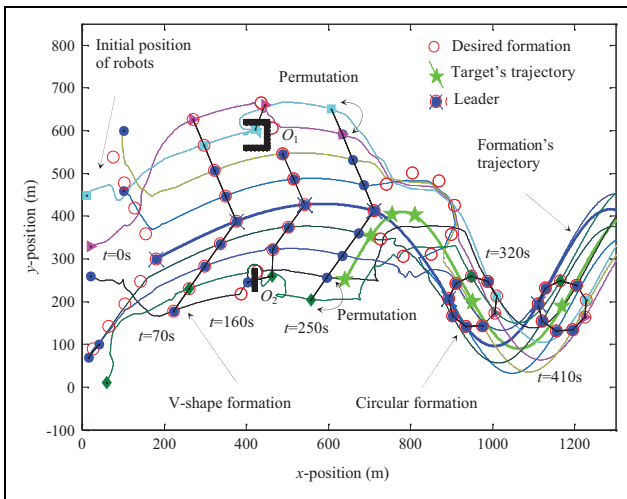
Equation (50) shows that the distributed virtual nodes j ($j = 1, 2, \dots, N$) on the circular desired formation are equidistant, and the distance from them to the target is $d_l^t = \|p_l - p_t\|$ (see Figure 6(b)). The desired radius is chosen such as the distances among agents guarantee collision avoidance when the circle formation is achieved.

Simulation results

In this section, we use the V-shape and circular desired formations, which are built in section *Case study*, to illustrate the proposed control algorithms. We assume that the initial velocities of the robots and the target are zero. The

Table 2. Parameter values.

| Parameter | Definition | Value |
|------------------------------------------------|------------------------------------------------|--------------|
| N | Number of robots | 9 |
| ϕ_d | Desired formation angle | $2\pi/3$ rad |
| r^r | Desired radius of circular formation | 60 m |
| r^t | Target approach radius | 100 m |
| r_r | Collision radius around each robot | 45 m |
| d | Desired distance between robots in the V-shape | 60 m |
| $\lambda, \lambda^*, \lambda_\alpha$ | Positive constants | 15, 10, 5 |
| $\varepsilon_1, \varepsilon_2, \varepsilon_3$ | Positive constants | 1, 0.5, 0.7 |
| $\kappa_i^{c1}, \kappa_i^{c2}$ | Constants for fast repulsion | 80, 12 |
| $\kappa_i^{o1}, \kappa_i^{o2}$ | Constants for fast obstacle avoidance | 90, 15 |
| $\kappa_i^{t1d}, \kappa_i^{t2}, \kappa_i^{t3}$ | Constants for approaching to target | 1, 0.6, 4 |
| $\kappa_i^{p1d}, \kappa_i^{p2}, \kappa_i^{p3}$ | Positive constants | 3, 4, 9 |
| $\kappa_i^f, \kappa_i^{tv}, \kappa_{ik}^c$ | Damping constants | 1, 1.2, 1.5 |

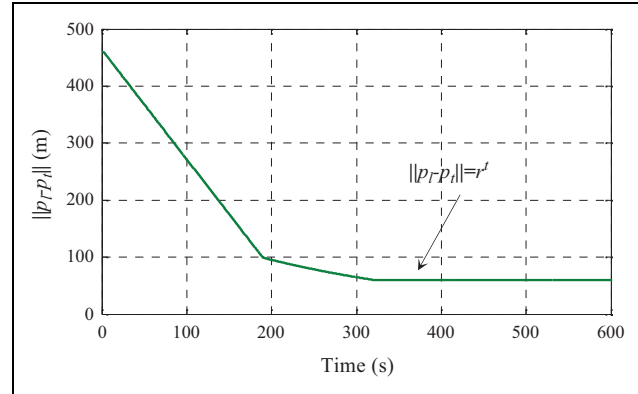
**Figure 7.** The evolution of a swarm following the desired formations under the influence of the dynamic environment while tracking a moving target.

initial positions of the robots are random. Each robot can sense the position of other robots within its sensing range as well as the positions of the target and obstacles. The target moves in a sine wave trajectory, which is defined as $p_t = (0.9t + 640, 160\sin(0.01t) + 250)^T$. The general parameters of the simulations are listed in Table 2.

Achievement of desired formation

For this simulation, the formation angle is selected as $\phi(t) = 2\pi/3$. The proposed algorithms were used to generate the desired formations (V-shape and circular shape) and control the robots to move toward the virtual nodes in the desired formation (algorithm 1).

The results in Figure 7 show that the desired formations were easily created. Robots, which have random

**Figure 8.** The distance between the leader and the target in case the leader is not hindered while tracking a moving target.

initial positions, quickly achieved the desired positions in these desired formations while tracking a moving target without collisions. The position permutations among the members in the formation appeared, but they did not influence the structure of the formation when tracking the target. As shown in Figure 7, at the initial time, one robot was chosen as the leader to drive its formation toward the target in a V-shape formation. At time $t = 70$ s, the V-shape formation was constructed, and it was kept until the square robot detected the obstacle O_1 . At time $t = 160$ s, while avoiding the obstacle O_1 , the virtual node, which was owned by the square robot, became a free node. Then, this virtual node attracted the triangular robot to become the active node at time $t = 200$ s. After escaping the obstacle, the square robot quickly approached the remaining free node of the desired formation, as shown in Figure 7. Similarly, the rhombus robot was permuted with other robots in the formation while avoiding the obstacle O_2 . At time $t = 250$ s, the V-shape formation changed to the circular formation to encircle the target. In this situation, the member robots became the free robots, and then they approached the desired circular formation. Figure 8 shows that this circular formation was kept around the target at the desired radius r^r at time $t = 320$ s. In other words, the leader's position is stable at the equilibrium point, at which $\|p_l - p_t\| = r^r$.

Next we consider the connection of a swarm when the leader was trapped in the complex obstacle (e.g. U-shape obstacle; see Figure 9). In this situation, the actual leader had to transfer its leadership to other members in the swarm, and managed to escape this obstacle. Figure 10 shows that, at time $t = 0$ s, the square robot was chosen as the leader, and its leadership was kept until it was trapped in the U-shape obstacle at time $t = 200$ s. While avoiding obstacle, the square leader transferred its leadership to the triangular robot, which was not faced with any obstacles and closest to the target. Then, this square leader became a free robot. It automatically found a way (lilac

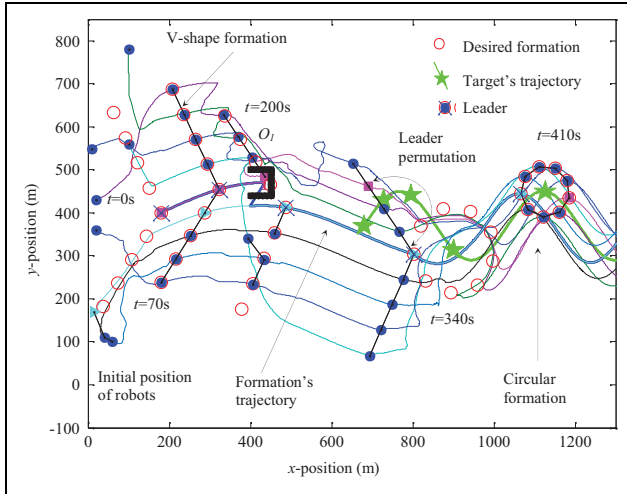


Figure 9. Path planning for a swarm following the desired formations while tracking a moving target with the leader permutation.

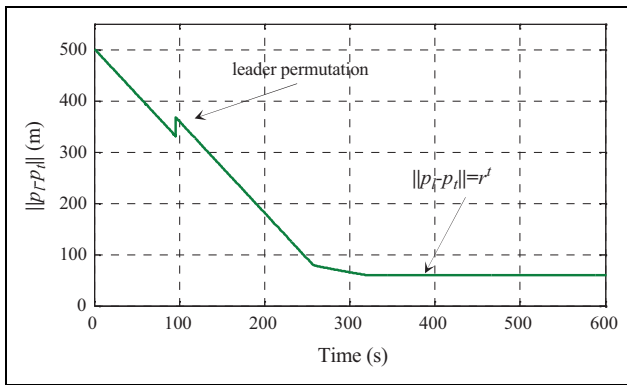


Figure 10. Distance between the leaders while tracking a moving target in case the leader is permuted.

way) to escape this U-shape obstacle in order to continue following its formation.

After receiving the leadership, the triangular robot reorganized a new formation and continued to lead this formation to track the target. The distance between the new leader and the target shrunk until it achieved the active radius of the circular desired formation $\|p_i - p_t\| = r^\tau$. Then, this distance was maintained to encircle the moving target (see Figure 10). Moreover, Figure 9 shows that the position permutation between the square leader and the triangular leader did not influence the structure of the formation.

The robustness of the swarm in noisy environment

In this subsection, we examine the robustness of the formation under the influences of noise and the change of the formation angle. Distances and velocities of agents are affected by noises, whose profiles are the Gaussian

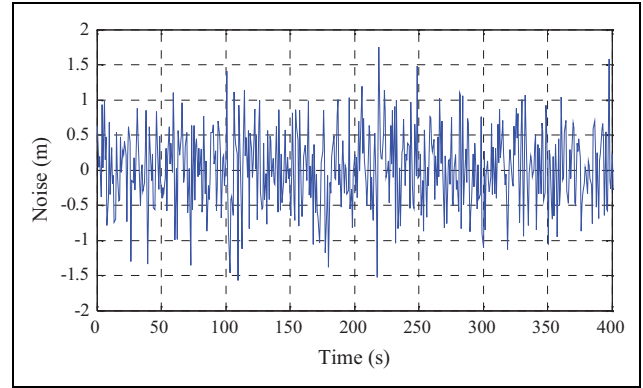


Figure 11. Measurement noise.

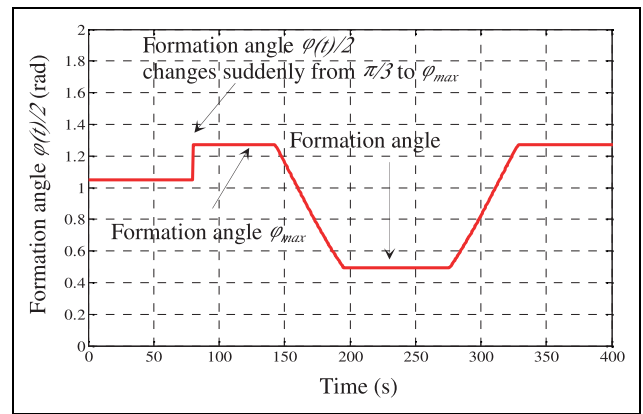


Figure 12. Formation angle $\phi(t)/2$ while tracking a moving target.

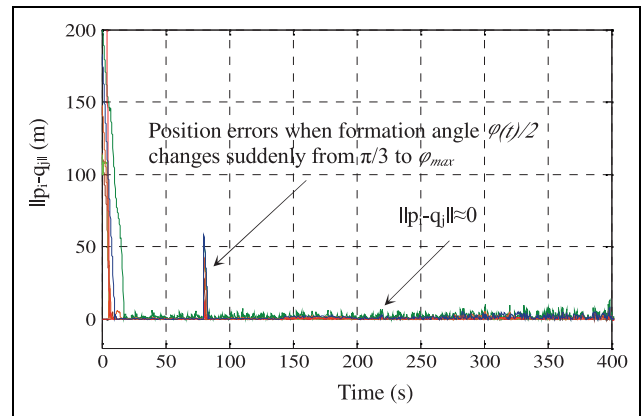


Figure 13. Position errors $\|p_i - q_j\|$ between the positions of the agents and their desired positions under the effect of the noise.

function with zero mean and variance of 1 (see Figure 11). The formation angle $\phi(t)$ is used for simulations as $\phi(t) = 2\pi/3 + 1.6\sin(0.2t)$ (see Figure 12).

Figure 14 shows that robot i was always close to active node j in the desired formation, and its formation was

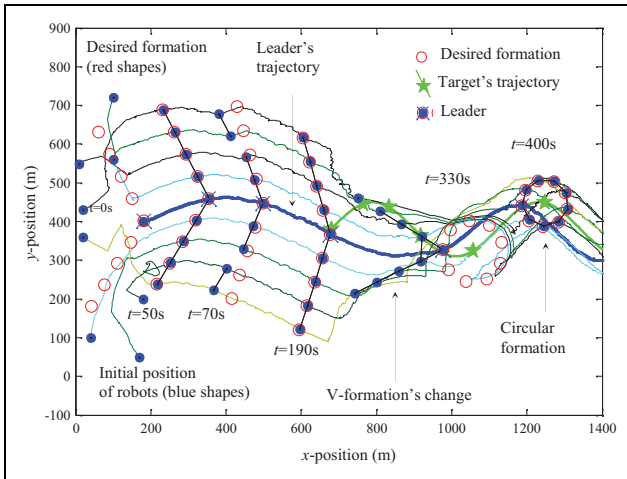


Figure 14. The influences of noises and the different formation angles on the swarm's trajectory while tracking a moving target.

maintained following the desired formations (V-shape and circular formation) under the influence of the noisy environment and the changes of the formation angle $\phi(t)/2$. The position error between each robot i and active node j , at which this robot i was occupying, is small (see Figures 12 and 13). Figure 14 also reveals that from the random initial positions, the free robots quickly found their desired position in the desired V-shape formation. Then, they tracked a moving target in a stable V-shape formation. At time $t = 70$ s, under the influence of the sudden change of the formation angle from $\phi = 2\pi/3$ to $\phi = (\pi - 0.6)$, the formation was broken, but it was quickly redesigned to continue to track the moving target. In contrast, when the formation angle $\phi(t)$ changed slowly, the formation of robots was always maintained following the desired V-shape formation with the small position errors (see Figures 13 and 14). Moreover, the simulation results show that the noise had influences on the position errors of the robot formation, but they only caused small formation changes, as shown in Figure 13.

Conclusion

In this article, a new method for decentralized formation control of autonomous robots has been presented, in which the swarm tracks a moving target in a dynamic and noisy environment. The robot system can form predefined formations such as V-shape or circular shape while avoiding collisions among agents of the swarm. The new mechanism which combines the rotational and the repulsive forces can drive robots to quickly escape complex obstacles such as those with concave shapes. The theoretical analysis of the proposed algorithm is given. In our future research, the proposed approach would serve as a framework for addressing the formation control of multiagent systems in two-dimensional space such as unmanned aerial vehicles and underwater robots.

Authors' note

The views, opinions, findings, and conclusions reflected in this publication are solely those of the authors and do not represent the official policy or position of the USDOT/OSTR, or any State or other entity.


Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was partially supported by the National Aeronautics and Space Administration under grant number NNX15AI02H issued through the NVSGC-RI program under sub-award no 19-21, the RID program under sub-award no 19-29, and the NVSGC-CD program under sub-award no 18-54. This work was also partially supported by the US Department of Transportation, Office of the Assistant Secretary for Research and Technology (USDOT/OST-R) under grant number 69A3551747126 through INSPIRE University Transportation Center (<http://inspire-utc.mst.edu>) at Missouri University of Science and Technology.

ORCID iD

Thang Nguyen  <https://orcid.org/0000-0001-6814-4885>

References

1. Yang H and Zhang F. Geometric formation control for autonomous underwater vehicles. In: *2010 IEEE International Conference on Robotics and Automation*, Anchorage, AK, USA, 3–7 May 2010, pp. 4288–4293. IEEE.
2. Zhang S, Yan W, and Xie G. Consensus-based leader-following formation control for a group of semi-biomimetic robotic fishes. *Int J Adv Robot Syst* 2017; 14(4): 1729881417720784.
3. Pham HX, La HM, Feil-Seifer D, et al. A distributed control framework for a team of unmanned aerial vehicles for dynamic wildfire tracking. In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, 24–28 September 2017, pp. 6648–6653. IEEE.
4. Palacios FM, Quesada ESE, Sanahuja G, et al. Test bed for applications of heterogeneous unmanned vehicles. *Int J Adv Robot Syst* 2017; 14(1): 1729881416687111.
5. Olfati-Saber R. Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Trans Automat Contr* 2006; 51(3): 401–420.
6. Jadbabaie A, Lin J, and Morse AS. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans Automat Contr* 2003; 48(6): 988–1001.
7. La HM, Lim R, and Sheng W. Multirobot cooperative learning for predator avoidance. *IEEE Trans Contr Syst Trans* 2015; 23(1): 52–63.
8. La HM and Sheng W. Flocking control of a mobile sensor network to track and observe a moving target. In: *2009 IEEE*

- International Conference on Robotics and Automation*, Kobe, Japan, 12–17 May 2009, pp. 3129–3134. IEEE.
9. Olfati-Saber R. Distributed tracking for mobile sensor networks with information-driven mobility. In: *2007 American Control Conference*, New York, NY, USA, 9–13 July 2007, pp. 4606–4612. IEEE.
 10. La HM and Sheng W. Distributed sensor fusion for scalar field mapping using mobile sensor networks. *IEEE Trans Cybern* 2013; 43(2): 766–778.
 11. Ogren P, Fiorelli E, and Leonard NE. Cooperative control of mobile sensor networks: adaptive gradient climbing in a distributed environment. *IEEE Trans Automat Contr* 2004; 49(8): 1292–1302.
 12. Dai Y, Choi KS, and Lee SG. Adaptive formation control and collision avoidance using a priority strategy for nonholonomic mobile robots. *Int J Adv Robot Syst* 2013; 10(2): 140.
 13. Nguyen T, La HM, Le TD, et al. Formation control and obstacle avoidance of multiple rectangular agents with limited communication ranges. *IEEE Trans Contr Network Syst* 2017; 4(4): 680–691.
 14. Nguyen T, Han TT, and La HM. Distributed flocking control of mobile robots by bounded feedback. In: *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Monticello, IL, USA, 27–30 September 2016, pp. 563–568. IEEE.
 15. Nguyen T, La HM, Azimi V, et al. Bounded distributed flocking control of nonholonomic mobile robots, *Swarm Intelligence - Volume 1: Principles, current algorithms and methods*, Chap. 11, pp. 297–321, September 2018. IET.
 16. Nguyen T and La HM. Distributed formation control of nonholonomic mobile robots by bounded feedback in the presence of obstacles. *2017 IEEE International Conference on Real-time Computing and Robotics (RCAR)*, Okinawa, Japan, 14–18 July 2017, pp. 206–211. IEEE.
 17. Han TT, La HM, and Dinh BH. Flocking of mobile robots by bounded feedback. In: *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, Fort Worth, TX, USA, 21–25 August 2016, pp. 689–694. IEEE.
 18. Santiaguillo-Salinas J and Aranda-Bricaire E. Containment problem with time-varying formation and collision avoidance for multiagent systems. *Int J Adv Robot Syst* 2017; 14(3): 1729881417703929.
 19. Saska M, Vonásek V, Krajník T, et al. Coordination and navigation of heterogeneous MAV-UGV formations localized by a ‘hawk-eye’-like approach under a model predictive control scheme. *Int J Robot Res* 2014; 33(10): 1393–1412.
 20. Saska M, Krajník T, Vonásek V, et al. Fault-tolerant formation driving mechanism designed for heterogeneous MAV-UGVs groups. *J Intell Robot Syst* 2014; 73(1): 603–622.
 21. Do KD and Lau MW. Practical formation control of multiple unicycle-type mobile robots with limited sensing ranges. *J Intell Robot Syst* 2011; 64(2): 245–275.
 22. Ge SS and Cui YJ. New potential functions for mobile robot path planning. *IEEE Trans Robot Autom* 2000; 16(5): 615–620.
 23. La HM and Sheng W. Dynamic target tracking and observing in a mobile sensor network. *Robot Auton Syst* 2012; 60(7): 996–1009.
 24. Dang AD and Horn J. Intelligent swarm-finding in formation control of multi-robots to track a moving target. *Int J Comput Elect Autom Contr Inform Eng* April 2014; 8(4): 561–567. <http://waset.org/Publications?p=88>.
 25. Dang AD, La HM, and Horn J. Distributed formation control for autonomous robots following desired shapes in noisy environment. In: *2016 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, Baden-Baden, Germany, 19–21 September 2016, pp. 285–290. IEEE.
 26. Dang AD and Horn J. Formation control of autonomous robots to track a moving target in an unknown environment. In: *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)*, Bali, Indonesia, 5–10 December 2014, pp. 896–901. IEEE.
 27. Dang AD and Horn J. Path planning for a formation of autonomous robots in an unknown environment using artificial force fields. In: *2014 18th International Conference on System Theory, Control and Computing (ICSTCC)*, Sinaia, Romania, 17–19 October 2014, pp. 773–778. IEEE.
 28. Dang AD and Horn J. Formation control of autonomous robots following desired formation during tracking a moving target. In: *2015 IEEE 2nd International Conference on Cybernetics (CYBCONF)*, Gdynia, Poland, 24–26 June 2015, pp. 160–165. IEEE.
 29. Liu Y and Passino KM. Stable social foraging swarms in a noisy environment. *IEEE Trans Automat Contr* 2004; 49(1): 30–44.
 30. Gazi V and Passino KM. Stability analysis of swarms. *IEEE Trans Automat Contr* 2003; 48(4): 692–697.
 31. Hou SP, Cheah CC, and Slotine JJE. Dynamic region following formation control for a swarm of robots. In: *2009 IEEE International Conference on Robotics and Automation*, Kobe, Japan, 12–17 May 2009, pp. 1929–1934. IEEE.
 32. Cheah CC, Hou SP, and Slotine JJE. Region-based shape control for a swarm of robots. *Automatica* 2009; 45(10): 2406–2411.
 33. Eren T. Formation shape control based on bearing rigidity. *Int J Contr* 2012; 85(9): 1361–1379.
 34. Fu-guang D, Peng J, Xin-qian B, et al. AUV local path planning based on virtual potential field. In: *2005 IEEE International Conference Mechatronics and Automation*, Niagara Falls, Ontario, Canada, 29 July–1 August 2005, pp. 1711–1716, Vol. 4. IEEE.
 35. Das AK, Fierro R, Kumar V, et al. A vision-based formation control framework. *IEEE Trnas Robot Autom* 2002; 18(5): 813–825.
 36. Spletzer JR and Fierro R. Optimal positioning strategies for shape changes in robot teams. In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 18–22 April 2005, pp. 742–747. IEEE.

37. Gould LL and Heppner F. The V-formation of Canada geese. *The Auk* July 1974; 91(3): 494–506. <http://www.jstor.org/stable/4084469>.
38. Badgerow JP. An analysis of function in the formation flight of Canada geese. *The Auk* October 1988; 105(4): 749–755. <http://www.jstor.org/stable/4087389>.
39. Wang C, Xie G, and Cao M. Forming circle formations of anonymous mobile agents with order preservation. *IEEE Trans Autom Contr* 2013; 58(12): 3248–3254.
40. Lee G, Yoon S, Chong NY, et al. A mobile sensor network forming concentric circles through local interaction and consensus building. *J Robot Mech* 2009; 21(4): 469–477.
41. Okamoto M and Akella MR. Novel potential-function-based control scheme for non-holonomic multi-agent systems to prevent the local minimum problem. *Int J Syst Sci* 2015; 46(12): 2150–2164.
42. Okamoto M and Akella MR. Avoiding the local-minimum problem in multi-agent systems with limited sensing and communication. *Int J Syst Sci* 2016; 47(8): 1943–1952.
43. Song C, Liu L, and Feng G. Coverage control for mobile sensor networks with input saturation. *Unmanned Syst* 2016; 04(01): 15–21.
44. Do K. Formation control of multiple elliptical agents with limited sensing ranges. *Automatica* 2012; 48(7): 1330–1338.
45. Shi Y, Li R, and Teo K. Cooperative enclosing control for multiple moving targets by a group of agents. *Int J Contr* 2015; 88(1): 80–89.
46. Nguyen MT, La HM, and Teague KA. Collaborative and compressed mobile sensing for data collection in distributed robotic networks. *IEEE Transactions on Control of Network Systems* 2018; 5(4): 1729–1740.
47. Briñon-Arranz L, Seuret A, and Canudas-de-Wit C. Cooperative control design for time-varying formations of multi-agent systems. *IEEE Trans Autom Contr* 2014; 59(8): 2283–2288.
48. Yu X and Liu L. Cooperative control for moving-target circular formation of nonholonomic vehicles. *IEEE Trans Autom Contr* 2017; 62(7): 3448–3454.
49. Freeman R and Kokotovic PV. *Robust nonlinear control design: state-space and Lyapunov techniques*. Berlin: Springer Science & Business Media, 2008.
50. Khalil HK. *Nonlinear systems*, 3rd ed. Upper Saddle River: Prentice Hall, 2002, p. 9.
51. Olfati-Saber R. *Flocking for multi-agent dynamic systems: algorithms and theory*. Technical Report Tech. Rep. 2004-005, California Inst. Technol., Pasadena, CA, 2004.
52. Slotine JJE and Li W. *Applied nonlinear control*. Vol. 199. Englewood Cliffs: Prentice hall, 1991.
53. Yao J, Ordóñez R, and Gaz V. Swarm tracking using artificial potentials and sliding mode control. *J Dynam Syst Meas Contr* 2007; 129(5): 749–754.
54. Gazi V, Fidan B, Ordóñez R, et al. A target tracking approach for nonholonomic agents based on artificial potentials and sliding mode control. *J Dynam Syst Meas Contr* 2012; 134(6): 061004–13.
55. Lial ML and Hungerford TW. *Mathematics with applications*. Boston: Addison-Wesley, 2002.