Multi-target pursuit formation of multi-agent systems*

Yan Jing(闫 敬)[a], Guan Xin-Ping(关新平)[a)b)†], and Luo Xiao-Yuan(罗小元)[a)]

[a)] *Institute of Electrical Engineering, Yanshan University, Qinhuangdao* 066004, *China*
[b)] *School of Electronic and Electric Engineering, Shanghai Jiaotong University, Shanghai* 200240, *China*

(Received 24 June 2010; revised manuscript received 10 August 2010)

The main goal of this paper is to design a team of agents that can accomplish multi-target pursuit formation using a developed leader–follower strategy. It is supposed that every target can accept a certain number of agents. First, each agent can automatically choose its target based on the distance from the agent to the target and the number of agents accepted by the target. In view of the fact that all agents are randomly dispersed in the workplace at the initial time, we present a numbering strategy for them. During the movement of agents, not every agent can always obtain pertinent state information about the targets. So, a developed leader–follower strategy and a pursuit formation algorithm are proposed. Under the proposed method, agents with the same target can maintain a circle formation. Furthermore, it turns out that the pursuit formation algorithm for agents to the desired formation is convergent. Simulation studies are provided to illustrate the effectiveness of the proposed method.

Keywords: multi-agent systems, pursuit, formation, leader–follower

PACS: 89.20.Ff, 87.85.St, 89.65.Ef, 02.30.Em **DOI:** 10.1088/1674-1056/20/1/018901

1. Introduction

In recent years, there has been a growth in interest in the area of multi-agent systems (MASs).[1−5] These systems potentially consist of a large number of agents, such as unmanned underwater vehicles (UUV),[1] unmanned aerial vehicles (UAV), and unmanned ground vehicles (UGV). MASs provide numerous applications in various practical fields, such as building automation, intelligent transportation systems,[2] surveillance, terrain data acquisition, and underwater exploration. The advantages of MASs over single-agent ones include cost reduction, efficiency and robustness improvements.

The prerequisite for these agents is team cooperation for accomplishing predefined goals and requirements. One critical problem for cooperative control is to design appropriate protocols and algorithms, such that the group of agents can maintain a desired formation. Traditional methods in the study of formation control generally fall into three categories, namely, leader–follower approaches,[6,7] virtual structure approaches,[8] and behaviour-based approaches.[9] Other techniques that have already been applied to this problem include matrix theory,[10,11] algebraic graph theory,[12,13] receding horizon control (RHC),[2,14] and game theory.[15]

Meanwhile, cyclic pursuit strategy has also been investigated in the formation control of multi-agent systems. Based on cyclic pursuit strategy, a distributed cooperative controller was proposed to solve the target-capturing task in Refs. [16]–[18]. In these researches, n identical autonomous mobile agents were distributed in the workplace, and it was assumed that each agent can always obtain pertinent state information of its neighbours and the target. However, this assumption is impractical or unnecessary in many practical situations. To illustrate this, consider animals that forage or travel in groups. In many cases, few individuals have pertinent information, such as knowledge about the location of a food source, or of a migration route.[6] Also, when the number of agents is very large, it is costly to exchange the formation data to each agent. Furthermore, agents are assumed to be dispersed in a counterclockwise star formation at the initial time. Obviously, this assumption is strict, because in many cases agents are randomly dispersed.

If more than one target is located in the workplace, target allocation for agents becomes a new issue. In Ref. [19], a routing policy was proposed to solve the dynamic vehicle routing problem, in which there were multiple vehicles and multiple classes of demands. In Ref. [20], Cassandras *et al.* considered a class of discrete resource allocation problems which

*Project partially supported by the National Basic Research Program of China (Grant No. 2010CB731800), the Key Project of Natural Science Foundation of China (Grant No. 60934003), the National Natural Science Foundation of China (Grant No. 61074065) and Key Project for Natural Science Research of Hebei Education Department, China (Grant No. ZD200908).
†Corresponding author. E-mail: xpguan@ysu.edu.cn
© 2011 Chinese Physical Society and IOP Publishing Ltd

can be formulated as discrete optimisation problems. To the best of the authors' knowledge, target allocation for multi-agent systems in formation has been paid little attention. Most of the existing works on formation control are based on the common assumption that a group of agents pursues the same target. However, this assumption is strict in certain situations. For instance, when more than one target is considered in the workplace, agents will face a dilemma in choosing their targets. To some extent, how to choose the appropriate targets for multi-agent systems becomes a new issue. This issue requires that every agent can automatically choose an appropriate target and keep a pursuit formation with the other agents that have the same target.

In this paper, we develop a novel formation framework for multi-agent systems. This framework may also be considered as a pursuit–evasion game, in which agents are pursuers and targets are evaders. The target, if enclosed by agents with a circle formation, may be considered to be captured. There are two control objectives in this research: one is that every agent can automatically choose its target at the initial time; another is that agents with the same target can maintain a circle formation ultimately enclosing the target, namely, the targets should be captured by agents. Under the first control objective, a strategy of choosing a target is presented. In view of the condition that not every agent can always obtain pertinent state information about the targets, a developed leader–follower strategy is presented. We also provide a numbering strategy, which can break up the initial state assumption of the agents in Refs. [16]–[18]. Based on the leader–follower and numbering strategies, a distributed pursuit formation algorithm is proposed. Meanwhile, to avoid collision between agents, an additional control input is combined with the formation algorithm. Finally, the convergence of the pursuit formation algorithm for agents to the desired formation is also analysed.

The paper is organised as follows. In Section 2, system modeling and problem formulation are presented. In Section 3, we introduce the pursuit formation control strategy. Simulation studies are provided to illustrate the effectiveness of our method in Section 4. Conclusions and future work are given in Section 5.

2. System modeling and problem formulation

Multi-agent systems: define a set of agents as $\Omega = i\ \{i = 1, 2, \ldots, N\}$, where N is the number of agents. For agent i with two-dimensional (2D) coordinates, the position and input vectors are denoted by $p_i \in \mathbb{R}^2$ and $u_i \in \mathbb{R}^2$, respectively. The dynamics of agent i at time t is described by the following continuous-time equation

$$\dot{p}_i(t) = u_i(t). \tag{1}$$

For the dynamic system, the following assumptions are made.

Assumption 1 We consider the formation framework as a pursuit–evasion game where agents are pursuers and targets are evaders. Targets, if enclosed by agents with a circle formation, are assumed to be captured by them. Initially, targets and agents disperse (or hide) randomly in the workplace. Once detected by agents, targets will move away from them. Let t_0 denote this specified instant when targets are detected. Meanwhile, it is assumed that agents can gain (or detect) the state information about the targets at time t_0.

Remark 1 For the precision constraints on sensors and disturbances in a real environment, agents cannot always obtain pertinent state information about the targets after time t_0. This situation is similar to the traveling or foraging of animals in a group. Not all the members have pertinent information, such as knowledge about the location of a food source, or of a migration route during the movement process.

Assumption 2 Each target can only accept a certain number of agents.

Remark 2 To accomplish a complex task in a civil or military field, such as surveillance and traffic control, etc., a certain number of agents is enough.

Each agent has a limited communication capability and it can only communicate with agents within its neighbourhood. The neighbouring set[21,22] of agent i at time t is denoted as

$$N_i(t) = \{j : \|p_i - p_j\| < r, j \in [1, 2, \ldots, N], j \neq i\}, \tag{2}$$

where $\|\cdot\|$ is the Euclidean norm. We assume that all agents have an identical influence (or sensing) radius $r > 0$. During the course of motion, the relative distances between agents may vary with time, so

the neighbours of each agent may change. We define the neighbourhood graph $G(t) = \{V, E(t)\}$[23] to be an undirected graph consisting of a set of vertices $V = \{1, 2, \ldots, N\}$, whose elements represent agents in the group, and a set of edges $E(t) = \{(i, j)\}$, containing unordered pairs of vertices that represent neighbouring relations at time t.

3. Multi-target pursuit formation

In this section, we design a team of agents who can accomplish multi-target pursuit formation by using a developed leader–follower strategy. First, the strategy for choosing a target is presented. Second, we provide the leader–follower strategy and the pursuit formation algorithm for multi-agent systems.

3.1. Strategy for choosing a target

At time t_0, $m(m < N)$ targets and N agents are considered in the workplace, and an arbitrary target k can accept s_k agents, namely, $\sum_{k=1}^{m} s_k = N$. To illustrate the strategy of choosing an appropriate target for agent i, we define another index P_{ik} as

$$P_{ik} = P_{ik}^{\mathrm{d}} + P_{ik}^{\rho}, \tag{3}$$

where P_{ik}^{d} denotes the probability term about distance from agent i to target k, and P_{ik}^{ρ} denotes the probability used to show whether target k can still accept agent i. If target k can still accept agent i, $P_{ik}^{\rho} = 1$; otherwise, $P_{ik}^{\rho} = 0$. Let D_{ik} denote the relative distance between agent 1 and an arbitrary target $k\,(k \in [1, 2, ..., m])$. The values of P_{ik}^{d} are shown in Table 1.

Table 1. Values of P_{ik}^{d}.

distances	D_{i1}	$\geq$	$\cdots$	$\geq$	D_{ik}	$\geq$	$\cdots$	$\geq$	D_{im}
P_{ik}^{d}	$1/m$		$\cdots$		k/m		$\cdots$		1

From Assumption 1, we know that all the agents are static in the workplace and they can gain the state information about the targets at time t_0. Then we can describe the process of choosing a target for each agent as follows.

Initiation At time t_0, a recorder is used to store the number of agents which belong to an arbitrary target. There are m recorders that are used to store the number of agents with respect to m targets. The initial value of each number is set at 0.

(a) The recording process starts from agent 1. By calculating the distance D_{1k}, we can obtain the values of P_{1k}^{d} and $P_{1k}^{\rho} = 1$. Further, we can obtain P_{1k} by calculating the function in Eq. (3). Choosing the largest value for P_{1j}, we can say target j is the target of agent 1. The number to target j in the recorder adds 1.

(b) Subsequent steps can be deduced by analogy. Finally, the target of every agent is determined.

Remark 3 For each agent, three situations should be highlighted as shown in Fig. 1. (i): On the left of Fig. 1, we can see target j and k keep the same distance from agent i. Meanwhile, target k can accept agent i, but target j cannot. So the chosen target for agent i is k. Comparing P_{ik} with P_{ij}, we find $P_{ik} > P_{ij}$. (ii): In the middle of Fig. 1, target j and k can both accept agent i, but target j keeps a shorter distance with agent i than target k. So the chosen target for agent i is j. Comparing P_{ik} with P_{ij}, we find $P_{ij} > P_{ik}$. (iii): On the right of Fig. 1, target k can accept agent k, but target j cannot. Meanwhile, target j keeps a shorter distance with agent i than target k. So the chosen target for agent i is k. Then we can find $P_{ij} = P_{ij}^{\mathrm{d}} \leq 1 < 1 + P_{ik}^{\mathrm{d}} = P_{ik}$, namely, $P_{ik} > P_{ij}$. In brief, we can draw the conclusion that Eq. (3) can satisfy all the situations in practice.

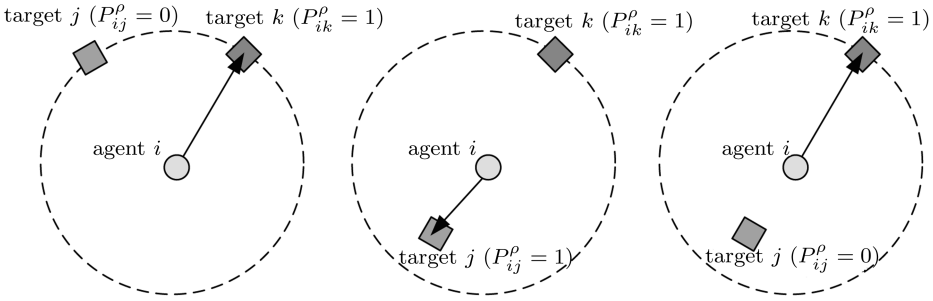


Fig. 1. Three situations for choosing the appropriate target.

3.2. The leader–follower strategy and pursuit formation algorithm

After choosing an appropriate target, each agent needs to keep a pursuit formation with the other agents which have the same target. To realise this process, we present a developed leader–follower strategy. The leaders are virtual agents whose communication and computing abilities are strong enough, and the agents can be considered as followers whose communication and computing abilities are limited. In other words, the formation information is divided into two independent parts, namely, global and local. The global information determines the position of the desired formation, and a small number of leaders can obtain the global formation information. The local information determines the relative positions of the followers with respect to the frame decided by the leaders.

Remark 4 In the leader–follower strategy, each agent requires information on all the subgroup-mates. Then, this leader–follower strategy may face the following challenges: (i) If the number of subgroup-mates is large, stricter requirements of computation ability will be needed. (ii) The leader should have anti-interference ability, because followers rely firmly on the leader. In view of these challenges, we assume that the leader should have strong computing and anti-interference abilities.

In this paper, there are m targets, namely, N agents will be divided into m groups based on the strategy of choosing a target. For simplicity, we only provide the control method used in one group of agents. The proposed method can then be extended to the remaining $m-1$ groups by updating the number of agents in every group. In the specified group, $n(n<N)$ agents are considered, and the initial position of the leader is the gravity centre of the positions for these agents. Agents in this group are driven to track the leader and maintain a circle formation enclosing it, and the leader is used to pursue the moving target. To finish the target tracking task for the leader, we can use the potential function method, first proposed by Khatib in Ref. [24]. Then the potential function can be defined by[25–27]

$$U=\frac{1}{2}\zeta_1 p_{lt}^{\mathrm{T}} p_{lt}, \tag{4}$$

where $\zeta_1>0$ is a scaling factor for the attraction, and $p_{lt}=p_t-p_l$ is the relative position vector from the leader to the target, p_t and p_l are the positions of the target and leader, respectively.

The vector p_{lt} can be described by $p_{lt}=[x_{lt},y_{lt}]^{\mathrm{T}}$, and $\|\cdot\|$ represents an Euclidean norm. $v_t\in\mathbb{R}^2$ and $v_l\in\mathbb{R}^2$ are the velocities of the target and leader, respectively. θ_t and θ_l are respectively the angles of v_t and v_l, and ψ is the angle of p_{lt}. The relative motion between leader and target is described by $\dot{p}_{lt}=[\dot{x}_{lt},\dot{y}_{lt}]^{\mathrm{T}}$, where $\dot{x}_{lt}=\|v_t\|\cos\theta_t-\|v_l\|\cos\theta_l$ and $\dot{y}_{lt}=\|v_t\|\sin\theta_t-\|v_l\|\sin\theta_l$.

Lemma 1 To make the leader track a moving target, the control input for the leader should be planned such that $\dot{p}_{lt}$ points to the negative gradient of U with respect to p_{lt}.

Proof First, we study the situation that the target is static in the workplace. To track a static target for the leader, the potential function can be defined as

$$\check{U}=\frac{1}{2}\check{\zeta}_1 p_{lt}^{\mathrm{T}} p_{lt}, \tag{5}$$

where $\check{\zeta}_1>0$ is a scaling factor for attraction; $p_{lt}=\check{p}_t-p_l$ is the relative position vector from the leader to the target, $\check{p}_t$ and p_l are the positions of target and leader, respectively.

We choose a Lyapunov function

$$H(p_l)=\check{U}=\frac{1}{2}\check{\zeta}_1 p_{lt}^{\mathrm{T}} p_{lt}\geq 0. \tag{6}$$

Differentiating $H(p_l)$ in Eq. (6) with respect to time t, we have

$$\dot{H}(p_l)=-\check{\zeta}_1 p_{lt}^{\mathrm{T}}\dot{p}_l. \tag{7}$$

To make the leader catch up with the static target, we can assume that $\dot{p}_l$ points to the negative gradient of $\check{U}$ with respect to p_l, namely,

$$\dot{p}_l=-\nabla_{p_l}\check{U}=\check{\zeta}_1(\check{p}_t-p_l). \tag{8}$$

Then Eq. (7) can be rewritten as

$$\dot{H}(p_l)=-\check{\zeta}_1^2\|p_{lt}\|^2\leq 0. \tag{9}$$

Note that $\dot{H}(p_l)=0$ if and only if $\|p_{lt}\|=0$, namely, $\dot{H}(p_{lt})=0$ only when the leader has caught up with the static target. Combining with the result $H(p_l)\geq 0$, we can obtain $\|p_{lt}\|\to 0$ when $t\to\infty$, namely, the assumption in Eq. (8) can guarantee the leader to catch up with the static target. Therefore, we can draw the conclusion that $\dot{p}_l$ should point to the negative gradient of $\check{U}$ with respect to p_l, if the leader requires to track a static target.

For a moving target, we can assume that the target is static relative to the leader. Then, the dynamic

environment can be transferred into a quasi-static environment, and the new state information can be expressed as follows: $\tilde{p}_l$ is the new position of the leader, $\dot{\tilde{p}}_l$ is its velocity, and $\tilde{U}$ denotes the new potential function of the leader. Based on the conclusion in the static environment, the control input of the leader should be planned such that $\dot{\tilde{p}}_l$ points to the negative gradient of $\tilde{U}$ with respect to $\tilde{p}_l$, if the leader requires to catch up with the quasi-static target.

Then, transfer the quasi-static environment into the dynamic environment, namely $p_{lt} = \tilde{p}_l$, $\dot{p}_{lt} = \dot{\tilde{p}}_l$, and $U = \tilde{U}$. We can draw the conclusion that $\dot{p}_{lt}$ should point to the negative gradient of U with respect to p_{lt}, if the leader requires to catch up with the moving target. This completes the proof. □

Based on Lemma 1, it is necessary to make

$$\dot{p}_{lt} = -\nabla p_{lt} U. \tag{10}$$

Then the following equation can be obtained

$$v_l = v_t + \zeta_1 p_{lt}, \tag{11}$$

which is equivalent to

$$\|v_l\| = (\|v_t\|^2 + 2\zeta_1 \|p_{lt}\| \|v_t\| \cos(\theta_t - \psi) + \zeta_1^2 \|p_{lt}\|^2)^{0.5}. \tag{12}$$

From Eq. (11), the velocity component v_l and v_t, which are perpendicular to p_{lt}, have the same value, namely,

$$\|v_l\| \sin(\theta_l - \psi) = \|v_t\| \sin(\theta_t - \psi), \tag{13}$$

which is equivalent to

$$\theta_l = \psi + \arcsin(\|v_t\| \sin(\theta_t - \psi) / \|v_l\|) \tag{14}$$

with $\|v_l\| \neq 0$.

Meanwhile, we should also consider another case where the leader is static. If the leader is static in the workplace, we can draw the following conclusion: the target is static in the workplace; meanwhile the leader has already caught up with the target. Then, we can assume $\theta_l = \theta_t$ if $\|v_l\| = 0$, namely, the leader's angle will be the same as the target's.

Based on the above discussion, the control input for the leader can be planned

$$u_l = v_l = (\|v_l\| \cos\theta_l, \|v_l\| \sin\theta_l)^{\mathrm{T}}, \tag{15}$$

where

$$\|v_l\| = (\|v_t\|^2 + 2\zeta_1 \|p_{lt}\| \|v_t\| \cos(\theta_t - \psi) + \zeta_1^2 \|p_{lt}\|^2)^{0.5},$$

$$\theta_l = \begin{cases} \psi + \arcsin(\|v_t\| \sin(\theta_t - \psi) / \|v_l\|), & \text{if } \|v_l\| \neq 0; \\ \theta_t, & \text{else.} \end{cases}$$

Lemma 2 Under the control input in Eq. (15), the leader can catch up with the target.

Proof We choose a Lyapunov function as

$$H(p_{lt}) = U = \frac{1}{2}\zeta_1 p_{lt}^{\mathrm{T}} p_{lt} \geq 0. \tag{16}$$

Differentiating $H(p_{lt})$ in Eq. (16) with respect to time t, we have

$$\dot{H}(p_{lt}) = \zeta_1 p_{lt}^{\mathrm{T}} \dot{p}_{lt}. \tag{17}$$

From Eq. (10), we have

$$\dot{H}(p_{lt}) = -\zeta_1^2 \|p_{lt}\|^2 \leq 0. \tag{18}$$

Note that $\dot{H}(p_{lt}) = 0$ if and only if $\|p_{lt}\| = 0$, namely, $\dot{H}(p_{lt}) = 0$ only when the leader has caught up with the target. Combining with the result $H(p_{lt}) \geq 0$, we can obtain $\|p_{lt}\| \to 0$ when $t \to \infty$, namely, the leader can catch up with the moving target. □

Next, we consider how to form a circle formation for the target-capturing task by the group of agents(or followers). In Refs. [16] and [17], a cyclic pursuit formation algorithm was proposed to solve the formation problem of multi-agent systems. Unfortunately, there it was assumed that agents were dispersed in a counterclockwise star formation at the initial time. Obviously, this assumption is strict, because in many cases agents are randomly dispersed. Taking the actual conditions into consideration, we provide a numbering strategy for the agents, in which they are randomly dispersed in the workplace. By using this numbering strategy, the formation control algorithm for agents is proposed. The process of numbering agents are described as follows.

Initiation At time t_0, n agents that have the same target are randomly numbered as $\tilde{1}, \tilde{2}, \ldots, \tilde{n}$, as shown Fig. 2(a). Let $p = (p_1, p_2, \ldots, p_n)^{\mathrm{T}}$ denote the position vector of the agents where $p_i = (x_i, y_i)^{\mathrm{T}}$. Define $A = \{i |\ y_i \geq 0, i \in [1, 2, \cdots, n]\}$ and $B = \{i |\ y_i < 0, i \in [1, 2, \cdots, n]\}$. It is assumed that the number of agents in set A is c ($c > 0$), then the number in set B is $n - c$. The angle θ_i between agent i and the leader can be described as

$$\theta_i = \begin{cases} \arccos \dfrac{x_i - x_l}{\sqrt{(x_i - x_l)^2 + (y_i - y_l)^2}}, & \text{if } \ i \in A; \\ 2\pi - \arccos \dfrac{x_i - x_l}{\sqrt{(x_i - x_l)^2 + (y_i - y_l)^2}}, & \text{if } \ i \in B; \end{cases} \tag{19}$$

where $x_l = (1/n)\sum_{i=1}^{n} x_i$, and $y_l = (1/n)\sum_{i=1}^{n} y_i$.

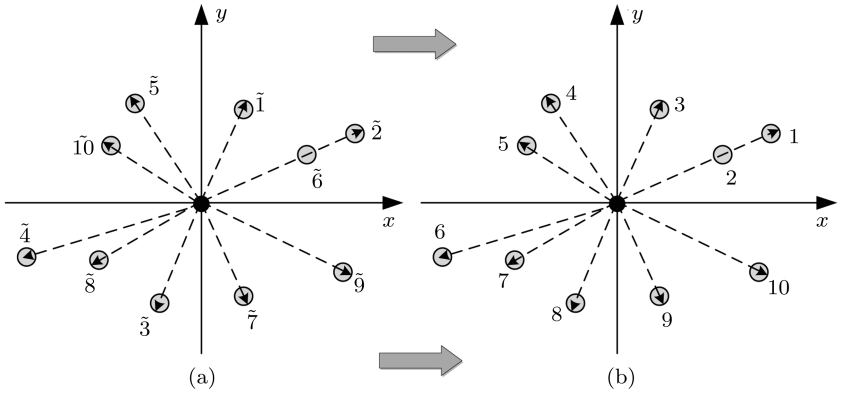


Fig. 2. The process of numbering agents. The black ball denotes the leader, and the grey ones denote agents.

Step 1 First, we number the agents in A. Compare θ_1 with θ_2, if $\theta_1 \geq \theta_2$, we consider agent 2 and compare θ_2 with θ_3. If $\theta_1 < \theta_2$, we continually compare θ_1 with θ_3. and so on. Finally, we can obtain the smallest angle that may be θ_h. Then agent h can be labeled as 1.

Step 2 Compare the rest $c-1$ agents in A, and number them by analogy, finally we can number all the agents in A as $1, 2, \ldots, c$;

Step 3 The process of numbering the agents in B is similar to the situation in A. The only difference is that the number record is described as $c+1, c+2, \ldots, n$. Finally, all the agents can be numbered, as described in Fig. 2(b).

Using the above numbering strategy, a distributed pursuit formation algorithm can be proposed. The detailed control objectives are described as follows:

(a) n agents enclose the leader at uniformly spaced angle and maintain this angle;

(b) each agent approaches the leader and maintains a distance R.

To illustrate the control objectives more explicitly, we show a desired formation pattern in Fig. 3, where $\phi_1 = \phi_2 = \cdots = \phi_6 = \pi/3$ and $\|r_i\| = \|p_i - p_l\| = R$, $i = 1, 2, \ldots, n$.

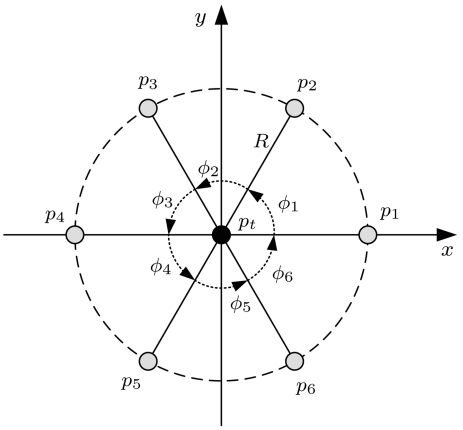


Fig. 3. A desired formation, in which the black ball denotes the leader and the grey ones denote the agents.

From Eq. (1) and noting that $r_i = p_i - p_l$ and $r_i = (\|r_i\|\cos\theta_i, \|r_i\|\sin\theta_i)^{\mathrm{T}}$, we have

$$\begin{aligned} u_i(t) &= \dot{r}_i(t) + \dot{p}_l(t) \\ &= \dot{r}_i(t) + u_l(t), \end{aligned} \tag{20}$$

subject to Eq. (15) and

$$\begin{aligned} &\dot{r}_i(t) \\ &= \begin{bmatrix} \cos\theta_i(t), & -\|r_i(t)\|\sin\theta_i(t) \\ \sin\theta_i(t), & \|r_i(t)\|\cos\theta_i(t) \end{bmatrix} \begin{bmatrix} \|\dot{r}_i(t)\| \\ \left\|\dot{\theta}_i(t)\right\| \end{bmatrix}, \end{aligned} \tag{21}$$

where $u_i(t)$ is the distributed control input for agent i at time t.

In Eq. (21), the adaptive control laws for agents i can be designed by

$$\begin{aligned} &\|\dot{r}_i(t)\| = \zeta_2(R - \|r_i(t)\|), \\ &\dot{\theta}_i(t) = \zeta_3 \varpi_i(t), \end{aligned} \tag{22}$$

where $\zeta_2 > 0$ and $\zeta_3 > 0$ are scaling factors, and $\varpi_i(t)$ can be planed such that

$$\begin{aligned} &\varpi_1(t) = -\theta_1(t), i = 1, \\ &\varpi_i(t) = \theta_1(t) - \theta_i(t) + 2\pi(i-1)/n, \quad i = 2, 3, \ldots, n. \end{aligned} \tag{23}$$

Theorem 1 Consider the specified group of n agents who have the same target. It is assumed that all the agents are randomly spaced in the workplace as shown in Fig. 2(a). By numbering the agents and using the control input in Eq. (20), all the agents will maintain a circle formation enclosing the leader, namely, $\|r_i(\infty)\| = R$, and $\phi_i = 2\pi/n$ for all the agents($i = 1, 2, 3, \ldots, n$).

Proof For agent 1, we can have $\dot{\theta}_1(t) = -\zeta_3\theta_1(t)$ where $\zeta_3 > 0$. Then, solving this differential equation, we have

$$\theta_1(t) = C_1 \, \mathrm{e}^{-\zeta_3 t}, \tag{24}$$

where C_1 is a constant.

It is obvious that $\theta_1(t) \to 0$ when $t \to \infty$, namely, the angle θ_1 converges to 0 finally.

For agent $i(i > 1)$, we can have $\dot{\theta}_i(t) = \zeta_3(\theta_1(t) - \theta_i(t) + 2\pi(i-1)/n)$. At arbitrary sampling constant, $\theta_1(t)$ can be considered constant. Then, solving this differential equation, we have

$$\theta_i(t) = C_2 \, \mathrm{e}^{-\zeta_3 t} + \theta_1(t) + 2\pi(i-1)/n, \tag{25}$$

where C_2 is a constant.

From Eqs. (24) and (25), we have $\theta_i(t) \to \theta_1(t) + 2\pi(i-1)/n \to 2\pi(i-1)/n$ when $t \to \infty$.

Further, it holds that $\theta_2(t) - \theta_1(t) = \theta_3(t) - \theta_2(t) = \cdots = \theta_n(t) - \theta_{n-1}(t)$ as $t \to \infty$. Therefore, $\phi_i = 2\pi/n$ for all the agents ($i = 1, 2, 3, \ldots, n$) when $t \to \infty$.

Next, we prove that all the agents maintain the same distance from the leader, namely, $\|r_i(\infty)\| = R$ for all the agents ($i = 1, 2, 3, \ldots, n$). Solving the differential equation $\|\dot{r}_i(t)\| = \zeta_2(R - \|r_i(t)\|)$ in Eq. (22), we have

$$\|r_i(t)\| = C_3 \, \mathrm{e}^{-\zeta_2 t} + R, \tag{26}$$

where C_3 is a constant.

Obviously, $\|r_i(t)\| \to R$ when $t \to \infty$, namely, $r_i(\infty) = R$ for all the agents ($i = 1, 2, 3, \ldots, n$). Then, the proof is complete. □

Corollary 1 Consider the system of N agents, in which agents are randomly spaced in the workplace. Extending the control method, which is used by arbitrary agent $i(i \le n)$, to all agents from 1 to N, then each agent from 1 to N will maintain a circle formation enclosing the target with the other agents who have the same target.

Proof By using the strategy of choosing a target, we obtain that each agent can automatically choose its target. Combining Lemma 2 with Theorem 1, we can draw the following conclusion: each agent from 1 to n can maintain a circle formation enclosing the target. Furthermore, we can extend the control method, which is used by arbitrary agent $i(i \le n)$, to all agents from 1 to N. The conclusion can be modified as: each agent from 1 to N will maintain a circle formation enclosing the target with the other agents which have the same target. □

To avoid collision between agents, we should also consider the interactions between agents. If the agents are close-spaced, a repulsive force should act on the agents, namely, there should no collision between them. Then we can use the potential function method to construct a potential field. Under the potential field, each agent can maintain a safe distance from its neighbours. Let $p_{ij} = \|p_i - p_j\|$ denote the actual distance between agent $i(i \le N)$ and agent j; $r^*(r^* < r)$ is the so-called threat distance of the neighbours. The definition of the interior potential function should satisfy:

(a) the function approaches infinity as $p_{ij} \to 0$;

(b) it cuts off at r^*.

Therefore, this differential potential function can be defined as

$$U_i^r = \sum_{j \in N_i(t)} \int_{r^*}^{p_{ij}} E(\tau) \mathrm{d}\tau \tag{27}$$

with

$$E(p_{ij}) = \begin{cases} -\dfrac{\zeta_4(p_{ij} - r^*)}{p_{ij}}, & p_{ij} \in (0, r^*], \\ 0, p_{ij} \in (r^*, \infty], \end{cases}$$

where $N_i(t)$ is the neighbourhood of agent i at time t. $\zeta_4 > 0$ is scaling factor for the interior potential.

Combining with the results in Eq. (20), the distributed control input of agent i at time t can be designed as follows:

$$\hat{u}_i(t) = u_i(t) + u_i^r(t), \tag{28}$$

where

$$u_i^r(t) = -\nabla_{p_i} U_i^r = \sum_{j \in N_i(t)} E(p_{ij}) \frac{(p_j - p_i)}{p_{ij}}.$$

Lemma 3[3] Under the control input in Eq. (28), there is no collision between agents.

Proof The proof is similar to the result in Ref. [3], and hence omitted.

4. Simulation results

This section presents the simulation studies of the proposed control scheme on a group of 35 agents and 3 targets. Initially, the targets and agents are randomly dispersed in the workplace. Once detected by agents, the targets will immediately move to avoid being captured by them. Under the control scheme in Section 3, all agents will achieve the required circle formation. The sampling time is $\delta = 0.01$s and the simulation is performed for $t = 47$s. The trajectories and velocities of the three targets are specified by

$$\begin{aligned} p_t^1 &= [3 + 0.06t, 6 - 0.06t]^{\mathrm{T}}, \ v_t^1 = [0.06, -0.06]^{\mathrm{T}}; \\ p_t^2 &= [3 + 0.065t, \ 7]^{\mathrm{T}}, \qquad v_t^2 = [0.065, \ 0]^{\mathrm{T}}; \\ p_t^3 &= [1, \ 5 + 0.1t]^{\mathrm{T}}, \qquad v_t^3 = [0, \ 0.01]^{\mathrm{T}}. \end{aligned}$$

Some parameters in the simulation are given in Table 2. The transmission range of each agent is $r = 2.5$, and the threat distance of the neighbour is $r^* = 0.05$ m. The initial positions of agents are chosen stochastically from a real white Gaussian noise of power 2dBW, which can be generated in MATLAB with the following command: $y = wgn(70, 1, 2)$.

Table 2. Parameters of each target.

	target 1	target 2	target 3
ζ_1	2	1	1.5
ζ_2	3	5	7
ζ_3	3	5	5
ζ_4	0.5	0.5	0.5
R	0.8	0.6	0.8
n	10	10	15

Figure 4 shows the initial positions of the agents and targets. After time t_0, agents begin to track the targets while maintaining the desired formations, which can be seen in Fig. 5. At time $t = 47$ s, all the agents have already formed the desired formations, as illustrated in Fig. 6.

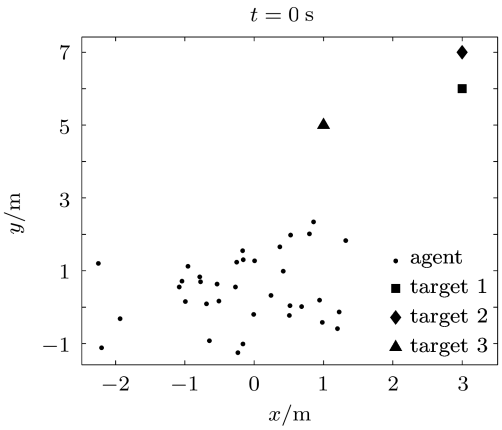


Fig. 4. The initial positions of the agents and targets.

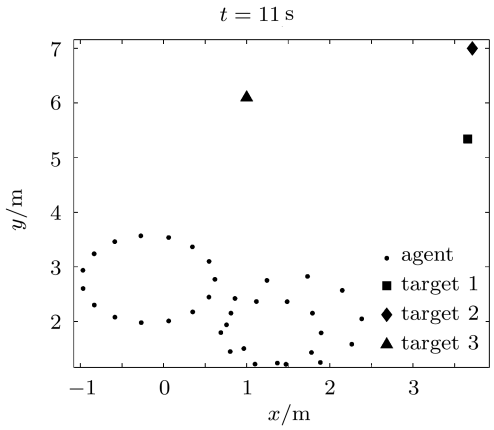


Fig. 5. The positions of the agents and targets at time $t = 11$ s.

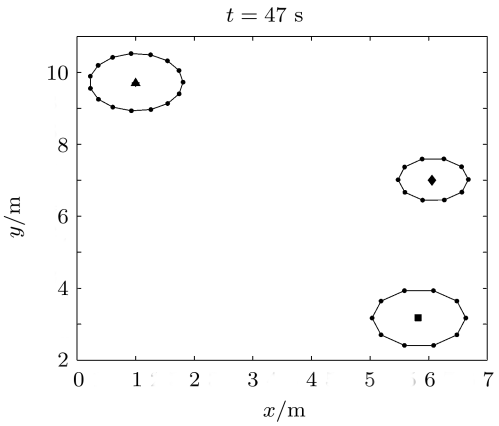


Fig. 6. The positions of the agents and targets at time $t = 47$ s.

From Table 2, we know that each agent should keep a desired position R with the target at a uniformly spaced angle. To show the results more clearly, we give the position and angle errors in Figs. 7 and 8, respectively. We define $\mathrm{Per}(1) = \left\|p_i - p_t^1\right\| - 0.8$ as the position error for the agents that belong to target 1, $\mathrm{Per}(2) = \left\|p_i - p_t^2\right\| - 0.6$ for the agents that belong to

target 2, and $\mathrm{Per}(3) = \left\|p_i - p_t^3\right\| - 0.8$ for the agents that belong to target 3. Further, we define $\mathrm{Aer}(1) = \|\theta_{i+1} - \theta_i\| - \pi/5$ as the angle errors for the agents that belong to target 1, $\mathrm{Aer}(2) = \|\theta_{i+1} - \theta_i\| - \pi/5$ for target 2, and $\mathrm{Aer}(3) = \|\theta_{i+1} - \theta_i\| - \pi/10$ for target 3. Through Figs. 7 and 8, it is obvious that all the errors approximately converge to zero.

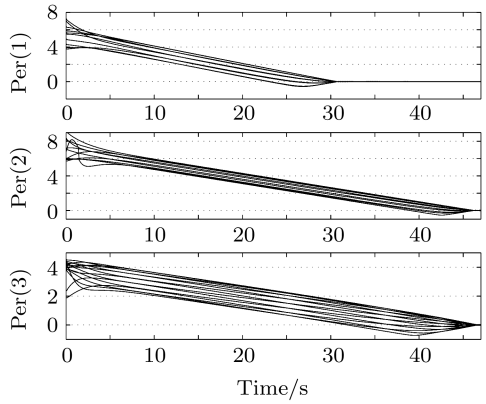


Fig. 7. The position errors for all the agents which belong to different targets.

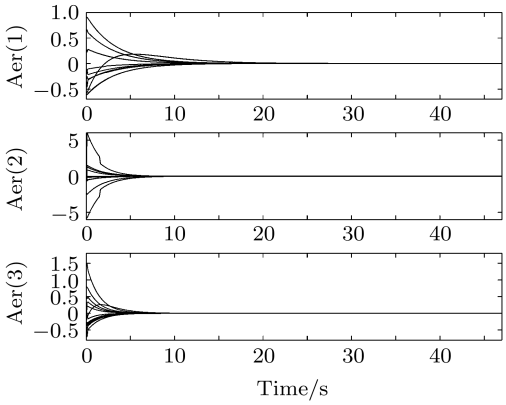


Fig. 8. The angle errors for all the agents which belong to different targets.

To avoid collision between agents, each agent should maintain a safe distance from its neighbours. Depicting all the relative distance between any two agents is a complex task (595 curves should be considered), so we only show some of them, namely, $i = 1, 6, 12, 18, 24, 30$. The relative distances between these agents are shown in Fig. 9. It is obvious that there is no collision between agents, because the distances are all greater than zero.

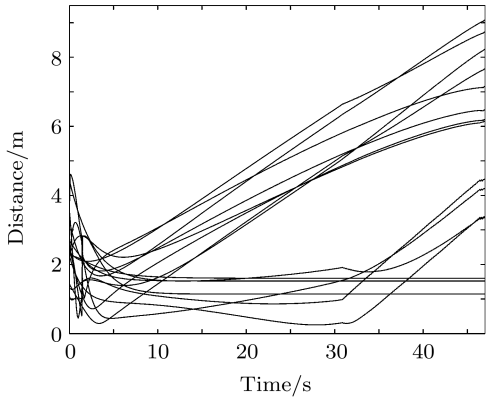


Fig. 9. The relative distances between the agents ($i = 1, 6, 12, 18, 24, 30$).

5. Conclusions and future work

We have presented a team of agents that can accomplish multi-target pursuit formation by using a developed leader–follower strategy. The formation information is divided into two independent parts, namely, global and local. The leaders decide the positions of the desired formations, and agents are used to maintain relative positions of the other agents by using the local information. Then, a numbering strategy and the distributed control algorithm are proposed. Under this control scheme, each agent from 1 to N can maintain a circle formation enclosing the target with the other agents which have the same target. Meanwhile, we also consider the collisions between agents, and the potential function method is used to guarantee that there is no collision between agents. Finally, simulation shows the effectiveness of the proposed method.

In this paper, the target assignment is static, i.e., each agent will select an invariable target to pursue. However, as the system evolves, each agent may select a different target to pursue according to certain optimal objectives. Separating the tasks of target selection and target enclosure in time may be a more interesting challenge. Although this paper does not consider systems with dynamic target assignment, we will consider this problem in more depth in our future work.

References

[1] Yuan H L and Qu Z H 2009 *IET Control Theory* A **3** 712

[2] Oliveira L B and Camponogara E 2010 *Transport. Res. C-Emer.* **18** 120

[3] Olfati-Saber R 2006 *IEEE Trans. Automat. Control.* **51** 401

[4] Kolling A and Carpin S 2010 *IEEE Trans. Robot.* **26** 32

[5] Xiao F, Wang L, Chen J and Gao Y P 2009 *Automatica* **45** 2605
[6] Couzin I D, Krause J, Franks N R and Levin S A 2005 *Nature* **433** 513
[7] Gu D B and Wang Z Y 2009 *IEEE Trans. Control Syst. Tech.* **17** 1211
[8] Lewis M A and Tan K H 1997 *Auton. Robot.* **4** 387
[9] Balch T and Arkin R C 1999 *IEEE Trans. Robot. Autom.* **14** 1999
[10] He Y, Zhang F M, Yang Y F and Li C F 2010 *Chin. Phys.* B **19** 060501R
[11] Qu Z H, Wang J and Hull R A 2008 *IEEE Trans. Autom. Control* **53** 894
[12] Luo X Y, Li S B and Guan X P 2009 *Chin. Phys.* B **18** 3104
[13] Ding W, Yan G F and Lin Z Y 2010 *Automatica* **46** 174
[14] Keviczky T, Borrelli F and Fregene K 2008 *IEEE Trans. Intell. Transp. Syst.* **16** 19
[15] Semsar-Kazerooni E and Khorasani K 2009 *Automatica* **45** 2205
[16] Marshall J A, Broucke M E and Francis B A 2006 *Automatica* **42** 3
[17] Kim T H and Sugie T 2007 *Automatica* **43** 1426
[18] Marshall J A 2005 *Coordinated Autonomy: Pursuit Formations of Multivehicle Systems* Ph. D. Thesis (Toronto: University of Toronto of Canada)
[19] Pavone M, Smith S L, Bullo F and Frazzoli E 2009 *Proceedings of American Control Confer.* St Louis, USA, June 10-12, 2009 p604
[20] Cassandras C G, Dai L Y and Panayiotou C G 1998 *IEEE Trans. Autom. Control* **43** 881
[21] Choi J, Oh S and Horowitz R 2009 *Automatica* **45** 2802
[22] Luo X Y, Li S B and Guan X P 2010 *Pattern Recogn. Lett.* **31** 800
[23] Tan F X, Guan X P and Liu D R 2008 *Chin. Phys.* B **17** 3531
[24] Khatib O 1986 *J. Robot. Res.* **5** 90
[25] Li H, Lin P and Zhang C X 2008 *Chin. Phys.* B **17** 4458
[26] Huang L 2009 *Robotic. Auton. Syst.* **57** 55
[27] Huang H and Fajen B R 2006 *Robot. Auton. Syst.* **54** 288