Dynamic Evasive Strategy of UAV Swarm Active Interception

Jiaxin Li[1], Jindong Zhu[1,2], Yunfei Liu[2], and Xiaowei Fu[1(✉)]

[1] Northwestern Polytechnical University, Xian 710129, China
fxw@nwpu.edu.cn
[2] AVIC Shenyang Aircraft Design and Research Institute, Shenyang 110035, China

Abstract. Aiming to maximize the interception time from the beginning of the confrontation to the attacker is successfully rounded up by the defenders, an evasive strategy for the attacker is designed. Under the different situations such as one defender, two defenders, and multiple defenders, a motion control method is designed to avoid the interception of defenders dynamically by changing the heading of the attacker. Finally, the simulation results show that the method has a good effect to maximize the interception time.

Keywords: UAV swarm · Attack and defense · Dynamic evasive strategy

1 Introduction

In recent years, the problems of UAV swarm attack and defense have gradually become an important research topic. Among them, the UAV defense problem about how a certain area can be protected effectively and an enemy can be rounded up is one of the hot research topics. Hamilton-Jacobi-Isaacs equation method (HJI) and its numerical approximation [2] are the most commonly used methods, although they can effectively solve the problem of attack and defense, their calculation cost is high and the computational complexity of this method will increase exponentially with the increase of the number of UAVs. Efficient solutions based on model predictive control (MPC) and distributed algorithms [3] are also the commonly used methods, although the solutions can reduce the computational complexity to a certain extent, their calculation cost is still a little high. Using geometric relations to solve the attack and defense problems can greatly reduce the computational complexity, but are usually limited to relatively simple scenarios with no obstacles or an equal constant speed for all UAVs, like UAV attack and defense strategy based on Apollonius circle [4], UAV defense strategy based on the Voronoi polygon [1] and so on. Among these strategies, the UAV swarm defense strategy based on the Voronoi polygon proposed can guarantee that the defenders successfully round up the attacker within a limited time in a convex polygon region.

To counter the UAV defense strategy based on the Voronoi polygon, it is necessary to design an evasive strategy. This paper designs this strategy for the attacker in the UAV swarm defense strategy based on the Voronoi polygon [1], to achieve the goal of

Z. Ren et al. (Eds.): *Proceedings of 2021 5th Chinese Conference on Swarm Intelligence and Cooperative Control*, LNEE 934, pp. 251–260, 2023.
https://doi.org/10.1007/978-981-19-3998-3_25

maximize the interception time from the beginning of the confrontation to the attacker is successfully rounded up by the defenders.

2 Scene Design and Problem Formulation

The scenario is shown in Fig. 1. The movement area of all UAVs is a convex polygon region, denoted as $D \in R^2$. There is a circular protected region $E \in D$, and the center of the protected region is O. There are two types of UAVs: defender and attacker, the attacker aims to enter the protected area E, and the defenders aim to prevent the attacker from entering protected area E and to round up the attacker. The attacker and the defenders are moving at an equal constant speed. Only when the attacker flies into the defenders' detection ranges, the defenders can detect the position information of the attacker. So does the attacker.

Before the defenders discover the attacker, the defenders use the parallel search method to search the region [7], and the division of the search area adopts the anchoring area division method [5–7]. Once detecting the attackers, the defenders use the UAV swarm defense strategy based on the Voronoi polygon to round up the attacker [1]. The attacker moves directly towards the protected area before detecting the defenders, and if it detects any of the defenders, the attacker will make a decision based on the defenders which are already detected. If it is believed to be threatened, the attacker immediately evades the defenders to move towards the protected area.

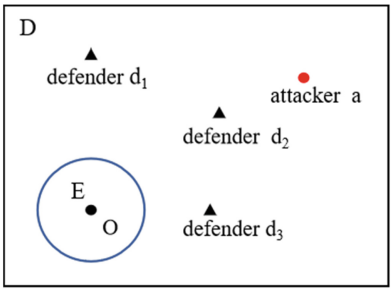

Fig. 1. Scene design diagram

There are N defenders $d^i, i = 1, \ldots, N$, and one attacker a. The dynamics of these UAVs are defined as:

$$\dot{\mathbf{x}}_k(\mathbf{t}) = \mathbf{u}_k(\mathbf{t})\ k = a, d^i \tag{1}$$

where $k = a$ represents the attacker, $k = d^i$ represents the defender d^i. $\mathbf{x_k(t)}$ is the UAV's position at the time t, $\mathbf{x}_k(\mathbf{0})$ is the initial position of the UAV. $\mathbf{u}_k(\mathbf{t})$ is the speed of the UAV, and there is a maximum speed limit: $\|\mathbf{u}_k(\mathbf{t})\| \leq v_{\max}, \forall t \geq 0, \forall i = 1, \ldots, N$, $v_{\max}$ is the maximum speed of the UAVs.

For any time t, the distance between the attacker and the defender $\{d^i\}_{i=1}^N$ is:

$$d_{\min}(t) = \min_{i=1,\ldots,N} \left\| \mathbf{x_{di}(t) - x_a(t)} \right\| \tag{2}$$

The condition for the defenders to intercept the attacker successfully is:

$$d_{\min}(T) \leq r_c \tag{3}$$

where $r_c > 0$ is the intercept radius which is manually specified. $T < \infty$ is the time when the defenders intercept the attacker successfully.

3 UAV Evasive Strategy Based on Speed Obstacle Method

3.1 Speed Obstacle Method

The speed obstacle method is mainly applied to UAV obstacle avoidance [9]. UAVs can use this method to avoid static or moving obstacles. First, a threat zone is defined. The threat zone is a circular area centered on the location of the obstacle d^i, as shown in Fig. 2. If the straight line with the relative speed of the obstacle and a UAV intersects the threat zone, the obstacle is considered as a threat to the UAV, and the UAV will make corresponding movement variations to avoid the obstacle. If the obstacle is not a threat to the UAV, the UAV continues to fly as it did before.

Each defender d^i is considered as an obstacle (The defender d^i is referred to as an obstacle in the following paragraphs). As shown in Fig. 2, x_{d^i} and x_a is the position of the obstacle d^i and the attacker. u_a is the speed of the attacker. u_{d^i} is the speed of the obstacle d^i. $\mathbf{u_r} = \mathbf{u_a} - \mathbf{u_{di}}$ is the relative speed between the attacker and the obstacle d^i. l is the line with the relative speed.

As shown in Fig. 2, $\odot x_{d^i}$ is the threat zone of the obstacle d^i. d is the radius of the threat zone and is artificially specified according to the threat range (Such as missile killing range, interception radius, etc.) of the obstacle d^i. When the line l with the relative speed intersects the threat zone, the obstacle d^i is considered to be a threat to the attacker.

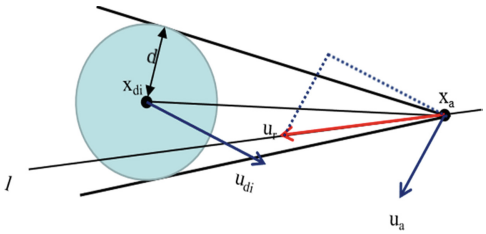

Fig. 2. Velocity barrier model

As shown in Fig. 3, follow the right-hand rule to establish a plane cartesian coordinate system. x_{d^i} is the origin coordinates, and $x_{d^i}x_a$ is the direction of the X-axis. $\mathbf{i}$ is the unit vector in the positive direction of the X-axis. $\mathbf{j}$ is the unit vector in the positive direction of the Y-axis. α is the angle between $\mathbf{u_r}$ and the X-axis. β is the angle between $\mathbf{u_r}$ and the boundary of the threat zone. ε is the angle between $\mathbf{u_r}$ and $\mathbf{u_{di}}$. θ_a is the angle between $\mathbf{u_a}$ and the X-axis. θ_{d^i} is the angle between $\mathbf{u_{di}}$ and the X-axis. D is the initial distance between attacker and the obstacle d^i. $d_{\min}$ is the shortest distance from x_{d^i} to the line with $\mathbf{u_r}$.

By geometric relationships, the angle γ between $x_a x_{d^i}$ and the boundary of the threat zone is:

$$\gamma = \arcsin(d/D) \tag{4}$$

The angle α between $\mathbf{u_r}$ and the X-axis is:

$$\alpha = \arcsin(d_{\min}/D) \tag{5}$$

The speeds of the attacker and the obstacle d^i are:

$$\mathbf{u_{di}} = u_{d^i}(\cos\theta_{d^i}\mathbf{i} + \sin\theta_{d^i}\mathbf{j}) \tag{6}$$

$$\mathbf{u_a} = u_a(\cos\theta_a\mathbf{i} + \sin\theta_a\mathbf{j}) \tag{7}$$

The relative speed $\mathbf{u_r}$ is:

$$\mathbf{u_r} = \mathbf{u_a} - \mathbf{u_{di}} = (u_a\cos\theta_a - u_{d^i}\cos\theta_{d^i})\mathbf{i} + (u_a\sin\theta_a - u_{d^i}\sin\theta_{d^i})\mathbf{j} \tag{8}$$

If $\alpha \geq \gamma$, the obstacle d^i is not a threat to the attacker. if $\alpha < \gamma$, the obstacle d^i is a threat to the attacker.

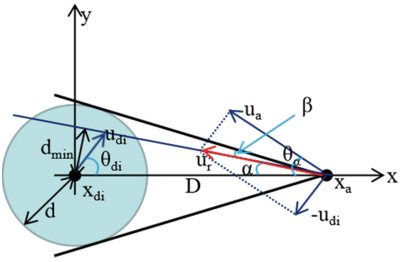

Fig. 3. Flight status diagram

3.2 Attacker's Avoidance of Dynamic Obstacles

When the attacker finds an obstacle which is a threat, the attacker can change the heading as little as possible. So the attacker should adjust the heading which can let $\alpha = \gamma$.

In the actual confrontation, the attacker is likely to face more than one obstacle which are the threats. Therefore, the following is discussed separately according to the number of obstacles.

One Obstacle. As shown in Fig. 4, $\mathbf{u'_a}$ is the speed of the attacker after the heading is changed. $\mathbf{u'_r}$ is the relative speed between the attacker's speed $\mathbf{u'_a}$ and the obstacle d^i's speed $\mathbf{u_{di}}$. ε' is the angle between $-\mathbf{u_{di}}$ and $\mathbf{u'_a}$. α' is the angle between $\mathbf{u'_r}$ and the X-axis. θ'_a is the angle between $\boldsymbol{u'_a}$ and the X-axis. When the angle between $\boldsymbol{u'_a}$ and the positive direction of the X-axis is more than 180°, $\theta'_a < 0$. When the angle between $\boldsymbol{u'_a}$ and the positive direction of the X-axis is less than 180°, $\theta'_a > 0$.

When the angle between $\mathbf{u'_a}$ and the positive direction of the X-axis is less than the angle between $\mathbf{u'_r}$ and the positive direction of the X-axis (as shown in Fig. 4a and Fig. 4b), the following relations in these cases are obtained:

$$\frac{|\mathbf{u'}_a|}{\sin\varepsilon'} = \frac{|-\mathbf{u_{di}}|}{\sin(\alpha' - \theta'_a)} \tag{9}$$

$$\theta'_a = \alpha - \arcsin\frac{|-\mathbf{u_{di}}|\sin\varepsilon'}{|\mathbf{u'_a}|} \tag{10}$$

$$\Delta\theta_a = \theta'_a - \theta_a \tag{11}$$

where $\Delta\theta_\alpha$ is the heading variation of the attacker.

When the angle between $\mathbf{u'_a}$ and the positive direction of the X-axis is more than the angle between $\mathbf{u'_r}$ and the positive direction of the X-axis (as shown in Fig. 4c and Fig. 4d), the following relations in these cases can be obtained:

$$\frac{|\mathbf{u'_a}|}{\sin\varepsilon'} = \frac{|-\mathbf{u_{di}}|}{\sin(\alpha' + \theta'_a)} \tag{12}$$

$$\theta'_a = \arcsin\frac{|-\mathbf{u_{di}}|\sin\varepsilon'}{|\mathbf{u'_a}|} - \alpha \tag{13}$$

$$\Delta\theta_a = \theta'_a - \theta_a \tag{14}$$

where $\Delta\theta_\alpha$ is the heading variation of the attacker.

The attacker's speed should eventually be:

$$\mathbf{u'_a} = |\mathbf{u_a}|(\cos\theta'_a\mathbf{i} + \sin\theta'_a\mathbf{j}) \tag{15}$$

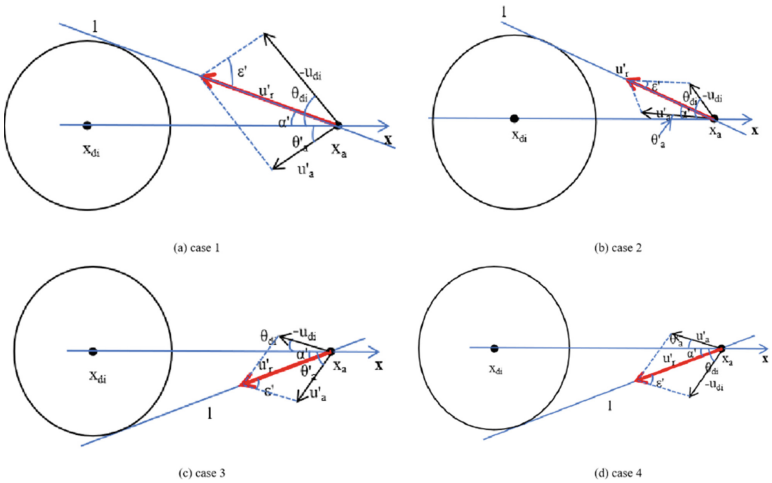

Fig. 4. The cases in which the attacker faces an obstacle

Two Obstacles. As shown in Fig. 5, x_{d^1} and x_{d^2} are the positions of these two obstacles d^1 and d^2. x_1 is represented the X-axis of the plane Cartesian coordinate system whose origin coordinates is x_{d^1}. x_2 is represented the X-axis of the plane Cartesian coordinate system whose origin coordinates is x_{d^2}. $\mathbf{u_{r^1}}$ and $\mathbf{u_{r^2}}$ (the red arrows) are the relative speeds between the attacker and the two obstacles d^1 and d^2 respectively. $\mathbf{u'_{r^1}}$ and $\mathbf{u'_{r^2}}$ (the black arrows) are the relative speeds between the attacker and the two obstacles d^1 and d^2 after the heading of the attacker is changed.

The heading of the attacker can be changed clockwise or counterclockwise to avoid the obstacles d^1 and d^2 respectively. When the angle between the relative speed $\mathbf{u_r}$ and the positive direction of the X-axis is more than 180°, the heading of the attack should be changed clockwise. Otherwise, the heading of the attack should be changed counterclockwise. For example, as shown in Fig. 5b, the angle between the relative speed $\mathbf{u_{r^1}}$ and the positive direction of the x_1 is more than 180°. The heading of the attacker should be changed clockwise. According to change the attacker's heading, the line with the relative speed between the attacker and the obstacle d^1 can be tangent to the threat zone in the figure (the blue tangent line). According to the directions of the heading of the attacker are changed to avoid the threat d^1 and d^2, the four cases can be divided.

The Heading Variations of the Attacker for the Two Obstacles d^1 and d^2 are in the Same Direction. The heading variations $\Delta\theta_{a^1}$ and $\Delta\theta_{a^2}$ of the attacker are calculated after the heading is changed to avoid the threat d^1 and d^2 respectively. As shown in Fig. 5a, the headings of the attacker should be both adjusted clockwise. As shown in Fig. 5b, the heading of the attacker should be both adjusted counterclockwise. When $\Delta\theta_{a^1} > \Delta\theta_{a^2}$, the heading variation of the attacker is $\Delta\theta_{a^1}$ to avoid d^1 and d^2 simultaneously. Otherwise, when $\Delta\theta_{a^1} < \Delta\theta_{a^2}$, the heading variation of the attacker is $\Delta\theta_{a^2}$.

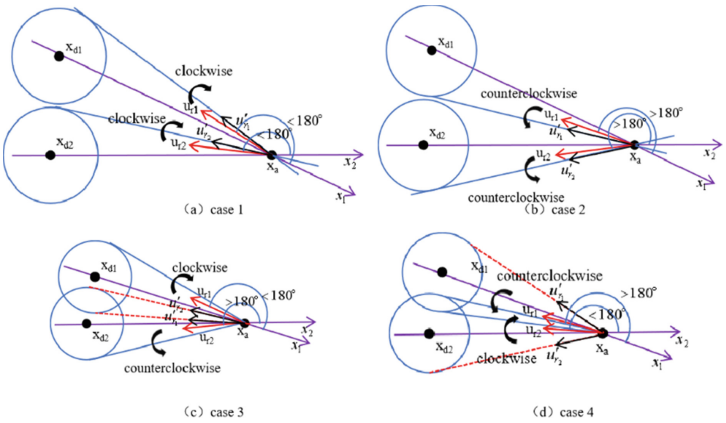

Fig. 5. The cases in which the attacker faces two obstacles

The Heading Variations of the Attacker for the Two Obstacles are in the Different Directions. As shown in Fig. 5c, the heading of the attack should be changed counterclockwise to avoid the threat d^1 and the heading of the attack should be changed clockwise to avoid

the threat d^2. As shown in Fig. 5d, the heading of the attack should be changed clockwise to avoid the threat d^1 and the heading of the attack should be changed counterclockwise to avoid the threat d^2. By changing the attacker's heading, the line with the relative speed between the attacker and the obstacle d^1 or d^2 is tangent to the threat zone in the figure. The tangent line is the red dotted line in the figure instead of the blue tangent line, because the heading variations of the attacker should be minimized. $\Delta\theta'_{a^1}$ and $\Delta\theta'_{a^2}$ are the two heading variations of the attacker. When $\Delta\theta'_{a^1} > \Delta\theta'_{a^2}$, the heading variation of the attacker is $\Delta\theta'_{a^2}$. When $\Delta\theta_{a^1} < \Delta\theta_{a^2}$, the heading variations of the attacker is $\Delta\theta'_{a^1}$.

The attacker's speed should eventually be:

$$\mathbf{u'_a} = |\mathbf{u_a}|(\cos\theta'_{a^k}\mathbf{i} + \sin\theta'\mathbf{j}) \quad k = 1, 2 \tag{16}$$

Multiple Obstacles. when the attacker faces more than two obstacles which are the threats, there are two cases the attacker may face.

The Distances Between all the Obstacles are Less than or Equal to $4d$. As shown in Fig. 6a, the attacker simply chooses one obstacle which is nearest to the attacker to avoid.

At Less One of the Distances Between all the Obstacles is more than $4d$. The attacker will fly towards the midpoint of the two obstacles that the distance between them is longest (the line marked by the red double arrow in Fig. 6b) between all the obstacles.

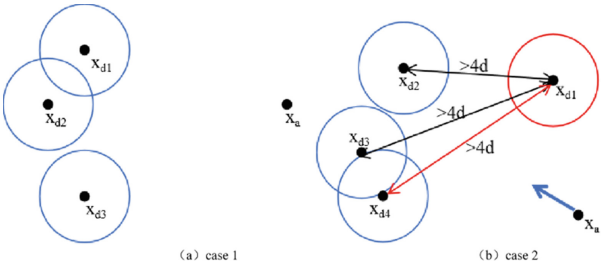

Fig. 6. The cases in which the attacker faces multiple obstacles

3.3 Algorithm

The main algorithm of the attacker avoidance strategy based on the speed obstacle method is shown in Fig. 7.

The attacker flies directly towards the protected area. When the attacker detects the defender, the attacker determines whether the defender is a threat. If there is a threat, the attacker adjusts the heading. If there is no threat, the attacker will fly towards the center of the protected area.

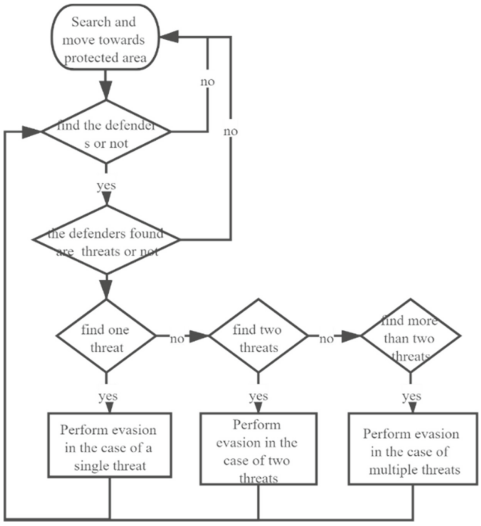

Fig. 7. The specific flow of attacker avoidance strategy based on speed obstacle method

4 The Simulation Results

Given a square area $D = [0,200] \times [0,200]$ (One unit length represents 1 km). the center O of the protected area E is (40,40), and the radius of the protected area E is 25. The speed of all UAVs is 0.9 Ma and all UAVs' maximum heading variation is 5°. $d = 5$ is the radius of the threat zone. It is assumed that the attacker will only fly towards the protected area from the angle (15°,75°) of the positive direction of the X-axis. The detection range of the UAVs is a fan. The detection angle is ±60°, and the detection distance is 20 km.

As shown in Fig. 8, there have 4 defenders and 1 attacker. The figure shows the snapshots at different times of simulation. The attacker is represented by a blue dot and its trajectory is represented by a blue line, and the detection range is represented by a blue fan. Defenders are represented by green, red, yellow, and black triangles, and the detection ranges of defenders are represented by green, red, yellow, and black fans. The trajectory of the defender which uses the area reduction strategy [1] is represented by the green line, the trajectory of the defender which uses the distance reduction strategy [1] is represented by the yellow line, and the trajectory of the defender which uses the pursuit strategy [8] is represented by the red line. The division of the defender's search area adopts the anchoring area division method [5–7] with the yellow dotted line. The attacker's initial position is (180,80).

As shown in Fig. 8(b), the attacker detects the defender d_1 and considers the defender d_1 is a threat. So the attacker avoids the threat d_1. At the same time, the defender d_1 also finds the attacker. Because there is no overlap between the Voronoi polygon V_a of the attacker and the protected area E, the defender d_2 executes the distance reduction strategy, and the defenders d_3 and d_4 execute the pursuit strategy.

Figure 8(c) shows that the attacker finds the defender d_2 and considers the defender d_2 is a threat. So the attacker avoids the threat d_2.

In Fig. 8(d), the defenders successfully capture the attacker. Defenders' duration of detecting the attacker is 42 s which is about 22.5% of the total time. And the total time is the time from the beginning of the confrontation to the attacker is successfully rounded up by the defenders.

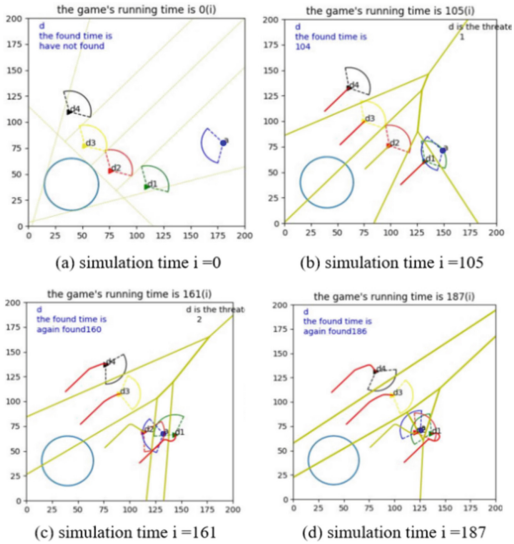

Fig. 8. Simulation screenshots

A total of 330 groups of simulations were did to compare the interception time whether the attacker adopts the evasive strategy or not. And there are about 92.4% group of experiments maximum 5.84 s for the interception time if the attacker adopts the evasive strategy.

5 Conclusion

Aiming at the defense strategy of the UAV swarm based on the Voronoi polygon, this paper designs an evasive strategy for the attacker based on the speed obstacle method. The simulation results show that the evasive strategy can maximize the interception time from the beginning of the confrontation to the attacker is successfully rounded up by the defenders. In the next step, the dynamic avoidance method which multiple attackers are involved can be studied. The avoidance strategy can also be studied for the situation that the speed of the attacker and the speed of the defender are different and the detection range of the attacker and the detection range of the defender are inconsistent.

Acknowledgments. This work was supported by the Aviation Science Foundation under Grant 2020Z023053001.

References

1. Deng, Z.Q., Kong, Z.D.: Multi-agent cooperative pursuit-defense strategy against one single attacker. IEEE Robot. Autom. Lett. **5**(4), 5772–5778 (2020)
2. Isaacs, R.: Differential Games: A Mathematical Theory With Applications to Warfare and Pursuit. J. Wiley and Sons (1965)
3. Ataei, A., Paschalidis, I.C.: Quadrotor deployment for emergency response in smart cities: a robust MPC approach. In: 2015 54th IEEE Conference on Decision and Control (CDC), pp. 5130–5135. IEEE, Osaka, Japan (2015)
4. Ramana, M.V., Kothari, M.: Pursuit-evasion games of high speed evader. J. Intell. Rob. Syst. **85**(2), 293–306 (2016)
5. Dai, J., Xu, F., Chen, Q.F.: Multi-UAV cooperative search on region division and path planning. Acta Aeronautica ET Astronautica Sinica **41**(S1), 148–156 (2020)
6. Hert, S., Lumelsky, V.: Polygon area decomposition for multiple-robot workspace division. Int. J. Comput. Geom. Appl. **8**(4), 437–466 (1998)
7. Yu, S., Zhou, R., Xia, J., et al.: Decomposition and coverage of multi-UAV cooperative search area. J. Beijing Univ. Aeronaut. Astronaut. **41**(1), 167–173 (2015)
8. Pierson, A., Wang, Z., Schwager, M.: Intercepting rogue robots: an algorithm for capturing multiple evaders with multiple pursuers. IEEE Robot. Autom. Lett. **2**(2), 530–537 (2017)
9. Zhang, H.H., Gan, X.S., Li, A., et al.: UAV obstacle and track recovery strategy based on velocity obstacle method. Syst. Eng. Electron. **42**(8), 1759–1767 (2020)