IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 31, 2019, accepted June 25, 2019, date of publication June 28, 2019, date of current version July 15, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2925633

Collision Avoidance Method for Self-Organizing Unmanned Aerial Vehicle Flights

YANG HUANG[1], JUN TANG[2], AND SONGYANG LAO[1]
[1]College of Systems Engineering, National University of Defense Technology, Changsha 410072, China
[2]Technical Innovation Cluster on Aeronautical Management, Universitat Autònoma de Barcelona, 08201 Sabadell, Spain

Corresponding author: Jun Tang (jun.tang@e-campus.uab.cat)

This work was supported in part by the National Natural Science Foundation of China, China, under Grant 71601181, in part by the Young Talents Lifting Project, China, under Grant 17JCJQQT048, in part by the Huxiang Young Talents, China, under Grant 2018RS3079, and in part by the Complex Situational Cognitive Technology under Grant 315050202.

ABSTRACT Autonomous unmanned aerial vehicle (UAV) swarm flights have been investigated widely. In the presence of a high airspace density and increasingly complex flight conditions, collision avoidance between UAV swarms is very important; however, this problem has not been fully addressed, particularly among self-organizing flight clusters. In this paper, we developed a method for avoiding collisions between different types of self-organized UAV clusters in various flight situations. The Reynolds rules were applied to self-organized flights of UAVs and a parameter optimization framework was used to optimize their organization, before developing a collision avoidance solution for UAV swarms. The proposed method can self-organize the flight of each UAV swarm during the overall process and the UAV swarm can continue to fly according to the self-organizing rules in the collision avoidance process. The UAVs in the airspace all make decisions according to their individual type. The UAVs in different UAV swarms can merge in the same space while avoiding collisions, where the UAV's self-organized flight process and collision avoidance process are very closely linked, and the trajectory is smooth to satisfy the actual operational needs. The numerical and experimental tests were conducted to demonstrate the effectiveness of the proposed algorithm. The results confirmed the effectiveness of this approach where self-organized flight cluster collision avoidance was successfully achieved by the UAV swarms.

INDEX TERMS Collision avoidance, parameter optimization, Reynolds rules, self-organized, unmanned aerial vehicle cluster.

I. INTRODUCTION

Owing to scientific and technological development, the roles of unmanned aerial vehicles (UAVs) have become increasingly important in both military and civilian fields. The inherently small and flexible nature of UAVs allows them to perform well in many areas, and they are particularly useful in environments and situations that are not suitable for humans. UAV swarm collaboration will become commonplace to meet the increasingly complex requirements of future missions. Communication and control become increasing difficult as the number of UAVs increases; thus, it is important to achieve self-organization behavior during UAV flights. In self-organizing cluster flights, communication and messaging can occur between the UAVs; however, external instructions cannot be sent. Thus, individual UAVs must analyze the available information and make their own decisions. However, a swarm of UAVs might merge from different directions during a self-organized UAV flight. Therefore, a collision avoidance method is required to prevent collisions assuming self-organized flight conditions.

Improvements and optimization are required for the UAV self-organizing flight model, and a collision avoidance method is needed for UAV swarms comprising multiple UAVs in the self-organized flight mode. Many methods are currently employed for self-organizing UAV flights, such as the Reynolds rules [1], pigeon flock algorithm, and goose swarm algorithm. In the field of UAV collision avoidance, the approaches employed include the artificial potential field method [2], [3], ant colony algorithm [4], [5], genetic algorithm [6], [7], particle swarm optimization algorithm [8], [9], Markov decision method [10], and dynamic programming method [11], [12].

UAV collision avoidance methods have been studied for decades and various strategies have been proposed for collision avoidance by self-organized drones, but each method

The associate editor coordinating the review of this manuscript and approving it for publication was Haluk Eren.

TABLE 1. Self-organizing uav flight anti-collision methods.

Model	Physical experiment	Target parameter optimization	UAV type	Leader agent based
Benedetti, et al.[13]	N	N	/	Y
Qiu, et al.[14]	N	Y	/	N
Di, et al.[15]	N	Y	/	N
Braga, et al.[16]	N	N	Q	N
Kownacki, et al.[17]	Y	N	F	Y
Clark, et al.[18]	Y	Y	F	N
Sajwan, et al.[19]	N	N	/	N
Quintero, et al.[20]	Y	Y	F	Y
Espelosín, et al.[21]	N	N	/	N
Hauert, et al.[22]	Y	N	F	N
Hildenbrandt, et al.[23]	N	N	/	N
Han, et al.[24]	N	N	/	Y
Vásárhelyi, et al.[25]	Y	Y	Q	N

has practical difficulties. Thus, to facilitate collision avoidance in self-organizing UAV flights in practical applications, we developed a method by optimizing the parameters of the force formula used in the Reynolds rules. In the following, we review relevant research into the UAV self-organized flight collision avoidance problem.

Table 1 lists 14 previous studies that proposed models for UAV self-organized flight collision avoidance, where the key characteristics of these studies are indicated as: physical experiment, target parameter optimization, UAV type, and leader agent based. In particular, ''physical experiment'' indicates whether a physical experiment was conducted, ''target parameter optimization'' indicates whether parameter optimization was performed based on the flight target, ''UAV type'' denotes the type of UAV, i.e., fixed-wing UAV(F) or quadrotor UAV(Q), while ''/'' indicates that the type of UAV was not specified, and ''leader agent based'' shows whether there was a leader in the UAV swarm.

Qiu and Duan [14] proposed a distributed optimization control framework for UAV clusters, which transformed the multi-objective optimization problem into a problem that can be solved using a single UAV. A distributed cluster control algorithm for UAVs based on an improved multipath I/O was proposed, which allows a UAV swarm to exhibit stable flight in a complex environment. Di *et al.* [15] proposed a two-layer control framework to address the collaborative surveillance problems using multiple UAVs, and distributed back horizon optimization for planning UAV movements in cluster flight situations. Braga *et al.* [16] used the Reynolds cluster rules to drive a swarm of UAVs, and then conducted physical experiments to demonstrate the feasibility of the method.

Kownacki *et al.* applied the two basic Reynolds rules and combined them with leadership features to achieve the collective flight of fixed-wing UAVs. Clark and Jacquesva [18] conducted a UAV cluster flight test at NASA's Dryden Flight Research Center as an important step in the development of a deployable distributed boid system. Sajwan *et al.* [19] clustered the UAVs in a leader–follower manner and solved each follower's control problem in the context of stochastic optimal control, where the problem was solved offline by dynamic programming to minimize the expectations within a limited range. Qiu and Duan [14] designed a distributed cluster control algorithm based on the dove cluster and a coordinated obstacle avoidance model for managing a situation where a UAV swarm flies through an environment containing obstacles without additional information. Hauert *et al.* [22] studied the trade-off between the communication range and flight dynamics in a simulation within the Reynolds cluster. Hildenbrandt *et al.* [23] combined conventional coordination rules based on separation, attraction, and alignment with the details of ostrich behavior to provide new insights into the complex clustering mechanisms employed by ostriches and other birds. Han *et al.* [24] proposed a new concept called ''soft control'' for regulating the collective behavior of self-organizing, multi-agent systems and demonstrated a natural method for intervening in distributed systems. Vásárhelyi *et al.* [25] solved the problem of seamless navigation by a UAV swarm in a narrow space by addressing and resolving issues in terms of constrained motion and communication capabilities, delays, disturbances, or obstacles. They proposed carefully selected sequence parameters and an evolutionary optimization framework for a fitness function, and they used 30 self-organizing UAVs in field experiments. This was the largest airborne outdoor system without central control reported in previous studies, and they demonstrated successful avoidance by clusters of collective UAVs with effective avoidance control.

In the present study, we developed an optimized, self-organizing flight method to ensure stable flight and collision avoidance by UAV clusters. Our approach does not require a leader to self-organize the flight and it achieves self-organized flight collision avoidance for multiple UAV swarms by optimizing the resulting trajectory for fixed-wing UAVs and quadrotor UAVs, which is difficult for other methods. Our method does not require a leader, which means that there is no mutual control between the drones and a flight decision is made by assessing the airspace situation. The core idea of this method involves optimizing UAV flight parameters by defining a fitness function for optimizing UAV flight. We studied the collision avoidance method in a self-organized UAV swarm flight and designed a new collision avoidance algorithm for self-organized flight collision avoidance. The method was validated in simulations using MATLAB.

The remainder of this paper is organized as follows. In Section II, we present the traditional Reynolds flocking model. In Section III, we describe the optimization method based on the Reynolds rules and a collision avoidance method for self-organizing flight. In Section IV, we present the simulation results and their analysis. We give our conclusions and discuss future research in Section V.

II. TRADITIONAL REYNOLDS FLOCKING MODEL

It should be noted that we do not consider the specific flight control problem for the UAV and our method only plans the trajectory of the UAV. Thus, we address the collision avoidance problem for UAV clusters with normal communication between UAVs regardless of the time delay for communication between UAVs.

We write $U =< P, V, A >$ to represent the UAV cluster system. $P = (\vec{p}_1, \vec{p}_2, \cdots, \vec{p}_i, \cdots, \vec{p}_N)$, $\vec{p}_i \in R^n$ are defined as the position vector of $\mathrm{UAV_i}$; $V = (\vec{v}_1, \vec{v}_2, \cdots, \vec{v}_i, \cdots, \vec{v}_N)$ are defined as the speed vector of $\mathrm{UAV_i}$; $A = (\vec{a}_1, \vec{a}_2, \cdots, \vec{a}_i, \cdots, \vec{a}_N)$, $\vec{a}_i \in R^n$ indicate the acceleration vector of $\mathrm{UAV_i}$. The formula for the motion of a UAV is as follows.

$$\begin{cases} \dot{\vec{p}}_i(t) = \vec{v}_i(t) \\ \dot{\vec{v}}_i(t) = \vec{a}_i(t) \end{cases} \tag{1}$$

According to this formula, the control of the motion by a UAV cluster may depend on the acceleration control. The speed and acceleration of the UAV are constrained by the following formulae:

$$\vec{v}_i(t) = \begin{cases} \vec{v}_i(t), & \left\|\vec{v}_i(t)\right\| \le v_{\max} \\ v_{\max} \cdot \dfrac{\vec{v}_i(t)}{\left\|\vec{v}_i(t)\right\|}, & \left\|\vec{v}_i(t)\right\| > v_{\max} \end{cases} \tag{2}$$

$$\vec{a}_i(t) = \begin{cases} \vec{a}_i(t), & \left\|\vec{a}_i(t)\right\| \le a_{\max} \\ a_{\max} \cdot \dfrac{\vec{a}_i(t)}{\left\|\vec{a}_i(t)\right\|}, & \left\|\vec{a}_i(t)\right\| > a_{\max}, \end{cases} \tag{3}$$

where $v_{\max}$ is the limited maximum speed value and $a_{\max}$ is the limited maximum acceleration value.

During the actual flight of an UAV, the change in flight angle needs to be limited owing to the flight performance limitations of the UAV.

The amount of change in the angle of an UAV is also limited, as shown in Eq. (4), where $\theta_{\max}$ is the maximum change in the angle of the UAV.

$$\frac{\vec{v}_i(t) \cdot \vec{v}_i(t-1)}{\|\vec{v}_i(t)\| \cdot \|\vec{v}_i(t-1)\|} > \cos\theta_{\max} \tag{4}$$

Reynolds first introduced a distributed flock behavior model in 1987 [1]. This model is based on the phenomenon of cluster flights according to the behavior of bird flocks. Subsequent research into flock modeling led to the definition of three simple rules for simulating flight decisions and state updates for a single individual in a flock of birds. The clustering algorithm based on the bird flight rules proposed by Reynolds can be summarized as follows.

R1 - Alignment: individuals attempt to align their velocity to the average velocity of nearby flock mates.

R2 - Cohesion: individuals attempt to match the average position of nearby flock mates.

R3 - Separation: individuals attempt to avoid collisions with nearby flock mates.

However, the Reynolds rules are simply a simulation of flock behavior and model details need to be optimized further for application to a real UAV system.

III. OPTIMIZED FLOCKING MODEL WITH COLLISION AVOIDANCE

The proposed flocking model is based on the Reynolds rules and the optimized model is based on a previously described model [25]. As shown in Figure 1, self-organized UAV flight cluster collision avoidance is based on the Reynolds rules, where the first advance involves providing an update by calculating the combined effect of different forces. None of the different UAV swarms in the simulation are controlled by ground control stations, but instead they rely solely on information sent by other UAVs for analysis and decision making. In Section A, we introduce the basic Reynolds cluster rules and some added buffer functions to make the cluster rules more suitable for actual flight. In Section B, we introduce the parameter optimization method for the cluster flight rules as well as the design of the corresponding objective function and iterative search of a random simulation scene to find the various force parameters that best match the objective function. In Section C, we describe the self-organizing collision avoidance method for UAV swarms.

A. OPTIMIZED FLOCKING MODEL

1) REPULSION

To keep the focal UAV, $\mathrm{UAV_s}$, separate from the other UAVs, each UAV receives a collision avoidance force known as the *separation* force, which is defined as follows:

$$\vec{f}_{sep}(t) = \frac{1}{n}\sum_{i=1}^{n}\vec{f}_{s,i}(t) = \frac{1}{n}\sum_{i=1}^{n} k \cdot \psi_i(t) \cdot \frac{\vec{p}_s(t) - \vec{p}_i(t)}{\left\|\vec{p}_s(t) - \vec{p}_i(t)\right\|} \tag{5}$$

where $\psi_i(t)$ is defined as:

$$\psi_i(t) = \begin{cases} \dfrac{1}{\sqrt{\left\|\vec{p}_s(t) - \vec{p}_i(t)\right\|}} - \dfrac{1}{\sqrt{d_{sep}}}, \\ \quad if \ \left\|\vec{p}_s(t) - \vec{p}_i(t)\right\| \le d_{sep} \\ 0, \quad if \ \left\|\vec{p}_s(t) - \vec{p}_i(t)\right\| > d_{sep}, \end{cases} \tag{6}$$

and n indicates the serial number of each remaining UAV other than the focal UAV ($\mathrm{UAV_s}$). In Eq. (6), $\psi_i(t)$ is the nonlinear gain of pairwise repulsion and $\left\|\vec{p}_s(t) - \vec{p}_i(t)\right\|$ is the distance between $\mathrm{UAV_s}$ and $\mathrm{UAV_i}$. The symbol d_{sep} defines the maximum interaction range at which UAVs start to repel

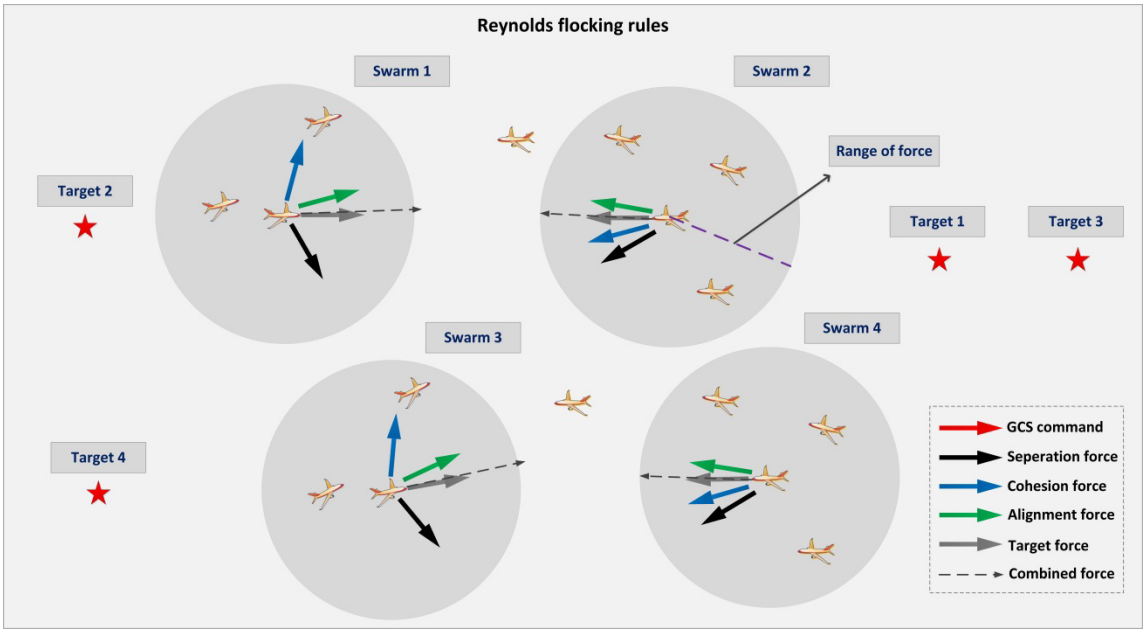


FIGURE 1. Self-organized flight concept diagram based on Reynolds rules.

each other and k is a coefficient that affects the magnitude of the repellent force.

2) VELOCITY ALIGNMENT

UAV clusters are collectively managed by individuals matching their average cluster velocity according to Eq. (7):

$$\vec{f}_{ali}(t)=\frac{1}{n}\sum_{i=1}^{n}\vec{f}_{a,i}(t)=\frac{1}{n}\sum_{i=1}^{n}\vec{v}_{a,i}(t)=\frac{1}{n}\sum_{i=1}^{n}(\vec{v}_i(t)-\vec{v}_a(t)) \tag{7}$$

where $\vec{v}_{a,i}(t)$ represents the speed of $\mathrm{UAV_i}$ relative to $\mathrm{UAV_a}$ and n represents the number of UAVs other than $\mathrm{UAV_a}$.

The flocking algorithm needs to handle possibly large velocity differentials at the same time, so a braking curve known as a velocity decay function (denoted by D(.)) is applied to the flocking algorithm. The decay function aims to provide constant acceleration at high speed and exponential acceleration at low speed, as shown in Eq. (8):

$$D(r,a,p)=\begin{cases}0, & \text{if } r\le 0\\ rp, & \text{if } o<rp<a/p\\ \sqrt{2ar-a^2/p^2}, & otherwise\end{cases} \tag{8}$$

where r is the distance between a UAV and a desired stopping point, a denotes the expected acceleration, and p represents the maximum speed difference allowed at a certain separation distance.

To construct the velocity alignment pattern, the speed difference that can be tolerated at a certain relative distance is given by Eq. (9).

$$v_{ij}^{tor\,\max}=\max\left(v^{tor},D\left(p_{ij}-p_0^{tor},a^{tor},k^{tor}\right)\right) \tag{9}$$

Thus, the velocity alignment term calculated for $\mathrm{UAV_i}$ with respect to $\mathrm{UAV_j}$ is:

$$\vec{v}_{ij}^{\,tor}(t)=\begin{cases}Q^{tor}\left(v_{ij}-v_{ij}^{tor\,\max}\right)\cdot\dfrac{\vec{v}_i(t)-\vec{v}_j(t)}{\left\|\vec{v}_i(t)-\vec{v}_j(t)\right\|}, & \text{if } v_{ij}>v_{ij}^{tor\,\max}\\ 0, & \text{otherwise}\end{cases} \tag{10}$$

where v^{tor} represents the amount of velocity slack required to allow for a certain amount of speed difference, which is independent of the relative distance, p_0^{tor} indicates the distance of $\mathrm{UAV_i}$ relative to the stop point in front of $\mathrm{UAV_j}$, and k^{tor} and a^{tor} represent the linear gain and acceleration parameters for a pair of consistent speed changes, respectively. In Eq. (10), Q^{tor} represents the linear coefficient of the speed fault reduction and v_{ij} represents the absolute value of the speed difference between $\mathrm{UAV_i}$ and $\mathrm{UAV_i}$.

3) FLOCK CENTERING

The UAV cluster needs to be as close to the center of nearby UAVs as possible, and thus the cohesion force is given by Eq. (11):

$$\vec{f}_{cen}(t)=\frac{1}{n}\sum_{i=1}^{n}\vec{f}_{c,i}(t)=\frac{1}{n}\sum_{i=1}^{n}\left(\vec{p}_i(t)-\vec{p}_c(t)\right) \tag{11}$$

where n represents the number of UAVs other than $\mathrm{UAV_c}$.

4) FLIGHT TARGET POSITION

To reach the desired position, the UAV cluster should know the general direction of the flight. We define the general

direction as:

$$\vec{f}_{tar}(t) = \vec{p}_{des} - \vec{p}_i(t) \tag{12}$$

where $\vec{p}_{des}$ represents the target end point vector of the UAV.

5) FINAL COMBINED FORCE FORMULAE

Thus, the final combined force formulae are:

$$\vec{v}_i(t) = \vec{f}_{steer} + \vec{v}_i(t-1) \tag{13}$$

$$\vec{p}_i(t) = \vec{p}_i(t-1) + \vec{v}_i(t) \tag{14}$$

where:

$$\vec{f}_{steer}(t) = \omega_1 \cdot \vec{f}_{sep}(t) + \omega_2 \cdot \vec{f}_{ali}(t) + \omega_3 \cdot \vec{f}_{cen}(t) + \omega_4 \cdot \vec{f}_{tar}(t) - \vec{v}_i(t-1) \tag{15}$$

In Eq. (15), ω_1, ω_2, ω_3, and ω_4 represent the weight coefficients of each force.

B. FLOCKING OPTIMIZATION ALGORITHM

1) FLIGHT EFFECT EVALUATION FUNCTION

A statistic is employed to measure the anti-collision effect during the self-organizing flight of a UAV swarm, which is then organized according to the statistic for parameter optimization. Let T be the entire simulation duration, $d_{ij}(t)$ is the distance between UAV_i and UAV_j at time t, and d_{col} refers to the danger zone around a UAV. In this situation, the collision risk is defined as shown in Eq. (16), where $\Theta(\cdot)$ is the Heaviside step function.

$$\xi_{col} = \frac{1}{T} \cdot \frac{1}{N(N-1)} \sum_{t=1}^{T} \sum_{i=1}^{N} \sum_{i=j} \Theta\left(d_{ij}(t) - d_{col}\right) \tag{16}$$

$$\Theta(x) = \begin{cases} 0, & x < 0 \\ 1, & x > 0 \\ 1/2, & x = 0 \end{cases} \tag{17}$$

The next key element for self-organizing flight is the velocity alignment motion. The UAV cluster velocity correlation is measured as shown by Eq. (17).

$$\xi_{cor} = \frac{1}{T} \cdot \frac{1}{N(N-1)} \sum_{t=1}^{T} \sum_{i=1}^{N} \sum_{i=j} \frac{\vec{v}_i(t) \cdot \vec{v}_j(t)}{\left|\vec{v}_i(t)\right| \cdot \left|\vec{v}_j(t)\right|} \tag{18}$$

Finally, we require that the UAV cluster move at a designated speed. Let $\vec{v}_{flock}$ be the designated speed and it is measured according to Eq. (18).

$$\xi_{vel} = \frac{1}{T} \cdot \frac{1}{N} \left| \sum_{t=1}^{T} \sum_{i=1}^{N} \left(\vec{v}_i(t) - \vec{v}_{flock}\right) \right| \tag{19}$$

The global function can be defined using the following three types of transfer functions.

The first type is a monotonically growing function represented as:

$$F_1(\xi, \xi_0, d) = 1 - S(\xi, \xi_0, d) \tag{20}$$

where $S(\xi, \xi_0, d)$ is a sigmoid function:

$$S(x, x_0, d) = \begin{cases} 1, & \text{if } x < x_0 - d \\ \frac{1}{2}\left(1 - \cos\left(\frac{\pi}{d}(x - x_0)\right)\right), & \text{if } x_0 - d < x < x_0 \\ 0, & otherwise \end{cases} \tag{21}$$

The second transfer function is a sharp peak transfer function, as follows.

$$F_2(\xi, z) = \frac{z^2}{(\xi - z)^2} \tag{22}$$

Finally, we construct the following single objective fitness function, which includes all of the related flocking behaviors.

$$F = F_{cor} \cdot F_{col} \cdot F_{speed} \tag{23}$$

$$\begin{cases} F_{cor} = \Theta(\xi_{cor}) \cdot \xi_{cor} \\ F_{col} = F_2(\xi_{col}, a_{up}) \\ F_{\text{speed}} = F_1(\xi_{vel}, v_{flock}, v_{up}) \end{cases} \tag{24}$$

C. COLLISION AVOIDANCE BETWEEN SWARMS OF UAVs

When different UAV swarms merge together, the probability of collisions will increase significantly. The separation force defined above is no longer effective because its direction is almost opposite to the direction of speed. Therefore, new rules are necessary to achieve the anti-collision targets between different UAV swarms.

In the presence of other UAV swarms, the force responsible for preventing collisions is a force that overlaps with other forces as little as possible; otherwise, the force used for collision prevention would be diluted. To meet this requirement, we employ a three-dimensional (3D) geometry-based approach where we calculate the closest proximity position of the two opposing flying UAVs, and the corresponding applied force is then obtained according to the speed and position information for the UAVs. The magnitude of the force is the same as the previous separation force but its direction differs.

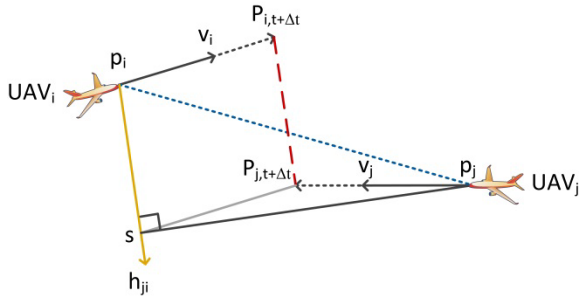


FIGURE 2. Method for calculating the nearest relative distance between two UAVs.

As shown in Figure 2, the key step involves calculating the time taken to arrive at the closest point of approach (CPA). CPA occurs at the moment when the two UAVs arrive at their closest point.

Algorithm 1 Self-Organizing UAV Flight Cluster Collision Avoidance Algorithm

Input: initial position, target position, initial velocity of UAVs, number of UAVs N, simulation step length S, labels of the UAVs
Output: trajectories of all the UAVs
while *flock* **do**
  if label is the same $\&\&D_{ij} < D_{safe1}$
    Steering_force = Repulsion_force + Alignment_force + Cohesion_force + Target_force – Current_velocity;
  elseif label is different $\&\&D_{ij} < D_{safe2}$
    Steering_force = Repulsion_force;
  end
  Updated_velocity = Current_velocity + Steering_force;
  Updated_position = Current_position + Updated_velocity;
end

First, we calculate the CPA between $\mathrm{UAV_i}$ and $\mathrm{UAV_j}$. The distance between $\mathrm{UAV_i}$ and $\mathrm{UAV_j}$ is defined according to Eqs (25)–(27):

$$d_{CPA,ij} = \frac{\left|\vec{h}_{ij} \cdot \vec{p}_{ij}\right|}{\left|\vec{h}_{ij}\right|} \tag{25}$$

$$\vec{h}_{ij} = \vec{v}_i \times \vec{v}_j \tag{26}$$

$$\vec{p}_{ij} = \vec{p}_j - \vec{p}_i \tag{27}$$

where $\vec{p}_i = (x_i, y_i, z_i)$ and $\vec{p}_j = (x_j, y_j, z_j)$ are the positions of $\mathrm{UAV_i}$ and $\mathrm{UAV_j}$, respectively, and $\vec{v}_i = (v_{i,x}, v_{i,y}, v_{i,z})$ and $\vec{v}_j = (v_{j,x}, v_{j,y}, v_{j,z})$ are their velocities.

We then determine the time required to reach CPA, which is defined as Δt. The formula $\vec{p}_{ij,t+\Delta t} \cdot \vec{v}_{ij} = 0$ holds when the UAVs arrive at the CPA. Assuming that the locations of the UAVs at time $t + \Delta t$ are as given by Eqs (28) and (29):

$$\vec{p}_{i,t+\Delta t} = \left(x_i + v_{i,x} \cdot \Delta t, y_i + v_{i,y} \cdot \Delta t, z_i + v_{i,z} \cdot \Delta t\right) \tag{28}$$

$$\vec{p}_{j,t+\Delta t} = \left(x_j + v_{j,x} \cdot \Delta t, y_j + v_{j,y} \cdot \Delta t, z_j + v_{j,z} \cdot \Delta t\right), \tag{29}$$

where t is the current moment, then we can calculate the relative position vectors for the UAVs at time $t + \Delta t$.

$$\begin{aligned}\overline{p}_{ij,t+\Delta t} = (&x_j - x_i + (v_{j,x} - v_{i,x}) \cdot \Delta t, y_j \\ &- y_i + (v_{j,y} - v_{i,y}) \cdot \Delta t, z_j - z_i + (v_{j,z} - v_{i,z}) \cdot \Delta t)\end{aligned} \tag{30}$$

Therefore, the time interval Δt can be calculated as follows.

$$\Delta t = \left|\frac{(x_j - x_i)(v_{j,x} - v_{i,x}) + (y_j - y_i)(v_{j,y} - v_{i,y}) + (z_j - z_i)(v_{j,z} - v_{i,z})}{(v_{i,x} - v_{j,x})^2 + (v_{i,y} - v_{j,y})^2 + (v_{i,z} - v_{j,z})^2}\right| \tag{31}$$

Based on the information described above, we propose a formula for calculating the anti-collision force for $\mathrm{UAV_i}$, as follows.

$$\vec{f}_i = \begin{cases} C \cdot \dfrac{\vec{h}_{ij}}{|\vec{v}_i| \cdot |\vec{v}_j|} \cdot \dfrac{d_{CPA,ij}}{|\vec{p}_{ij}|}, & \text{if } \vec{h}_{ij} \cdot \vec{p}_{ij} < 0 \\ C \cdot \dfrac{\vec{h}_{ij}}{|\vec{v}_{ij}| \cdot |\vec{v}_j|} \cdot \dfrac{d_{CPA,ij}}{|\vec{p}_{ij}|}, & \text{if } \vec{h}_{ij} \cdot \vec{p}_{ij} \geq 0 \end{cases} \tag{32}$$

Similarly, the anti-collision force for $\mathrm{UAV_j}$ can be written as follows.

$$\vec{f}_j = \begin{cases} C \cdot \dfrac{\vec{h}_{ji}}{|\vec{v}_i| \cdot |\vec{v}_j|} \cdot \dfrac{d_{CPA,ij}}{|\vec{p}_{ij}|}, & \text{if } \vec{h}_{ji} \cdot \vec{p}_{ji} < 0 \\ C \cdot \dfrac{\vec{h}_{ji}}{|\vec{v}_i| \cdot |\vec{v}_j|} \cdot \dfrac{d_{CPA,ij}}{|\vec{p}_{ij}|}, & \text{if } \vec{h}_{ji} \cdot \vec{p}_{ji} \geq 0, \end{cases}$$

$$\vec{h}_{ji} = \vec{v}_j \times \vec{v}_i \tag{33}$$

The basic idea of this method is that the direction of the force is perpendicular to the relative position vector of the UAV when it reaches CPA. This approach gives the UAV the most effective control over the relative distance beyond the safe distance to the CPA.

Algorithm 1 is the collision avoidance algorithm for a cluster of self-organizing UAVs, where the core step involves distinguishing the UAV repulsive force from different UAV swarms.

IV. SIMULATION AND RESULTS

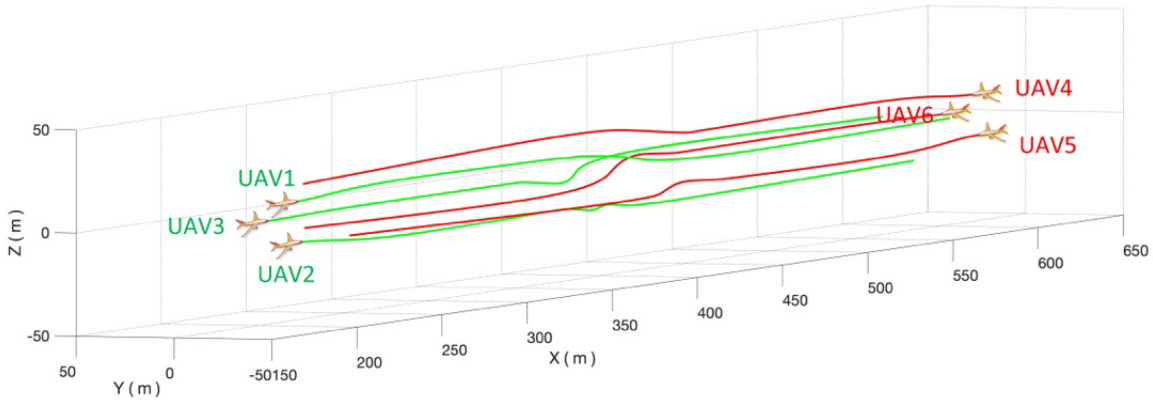
Scripts were written in Matlab to generate four different simulation scenarios. Each UAV swarm was assumed to be heading toward its own target position along a straight line at 5 m/s. The minimum safe separation distance, d0, was defined as 15 m. The interval time, t0, for discrete points was set as 0.1 s. The corresponding weight coefficients w1, w2, w3, and w4 were calculated as 427, 0.225, 0.452, and 548, respectively. The maximum variation range for the pitch angle, A, was 0.74, and the minimum value, B, was −0.74. We assumed that each UAV could receive information from the other UAVs and that the simulation calculation times could be ignored. The simulations were performed using an HP EliteBook laptop, with an Intel i5 processor at 2.6 GHz and 4 GB of RAM.

A. UAVs IN A SWARM FLYING TOWARD EACH OTHER

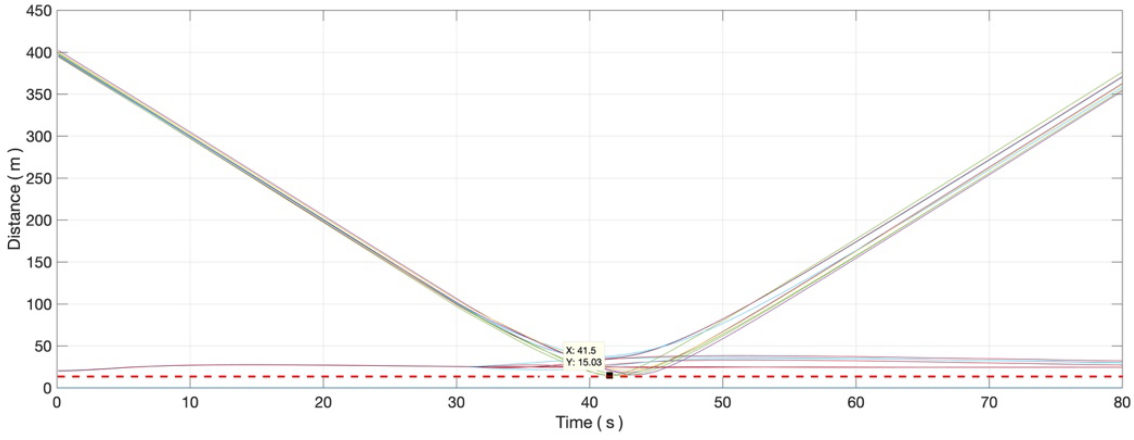
We assumed that two swarms of UAVs approached each other from opposite directions, and thus collisions were imminent. In the simulations, each UAV swarm contained three UAVs and each UAV swarm was self-organized to form its own team

TABLE 2. Initial UAV states for the first flight scenario.

Number	UAV1	UAV2	UAV3	UAV4	UAV5	UAV6
Initial position of UAV (m, m, m)	(200, –17.3, 10)	(204, –17.3, –10)	(205, 0, 0)	(600, –17.3, 14)	(603, –17.3, 6)	(601, 0, 4)
Initial speed of UAV (m/s, m/s, m/s)	(5, 0, 0)	(5, 0, 0)	(5, 0, 0)	(–5, 0, 0)	(–5, 0, 0)	(–5, 0, 0)
Initial angle of UAV (rad, rad, rad)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(π, 0, 0)	(π, 0, 0)	(π, 0, 0)
UAV flight end point (m, m, m)	(1000, –8.7, 0)	(1000, –8.7, 0)	(1000, –8.7, 0)	(–200, –8.7, 0)	(–200, –8.7, 0)	(–200, –8.7, 0)



a



b

FIGURE 3. Simulation results for scenario one.

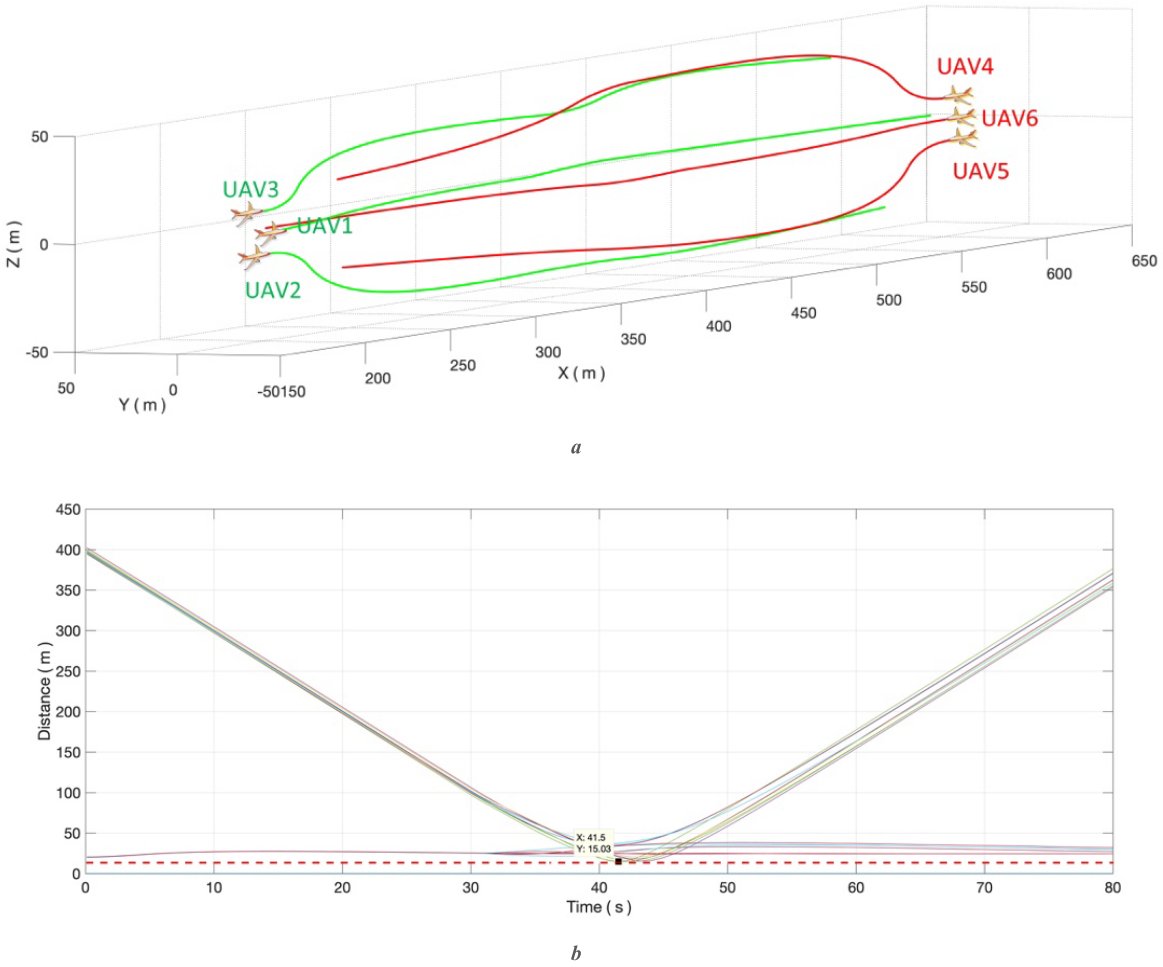
at the target location. The initial state and target status for each UAV are shown in Table 2.

In Figure 3*a*, each UAV trajectory is displayed in the form of a 3D curve. The green tracks represent the trajectories of the first UAV swarm and the red tracks represent the trajectories of the second swarm. Figure 3*a* shows that the trajectories of the two swarms changed continually during the collision avoidance process. The UAVs in the two swarms flew from the scheduled starting position in a self-organized manner until the end of the flight. During the entire collision avoidance process, the two UAV swarms adaptively generated collision avoidance movements within the collision area, where the distributions of their positions in space were uniform and their trajectories satisfied the actual flight demands.

Figure 3*b* shows the relative distance between each UAV pair. The minimum distance between any two UAVs was 15.03 m, which was above the lower limit of the safe separation distance.

B. UAV SWARM FLIGHT SCENARIO THAT MIGHT CAUSE A DOMINO EFFECT

A second scenario was generated to detect whether collision avoidance could be achieved in a high traffic density situation



a

b

FIGURE 4. Simulation results for scenario two.

TABLE 3. Initial UAV states for the second flight scenario.

Number	UAV1	UAV2	UAV3	UAV4	UAV5	UAV6
Initial position of UAV (m, m, m)	(200, 0, 10)	(204, 0, –10)	(205, 0, 0)	(600, 0, 14)	(603, 0, 6)	(601, 0, 4)
Initial speed of UAV (m/s, m/s, m/s)	(5, 0, 0)	(5, 0, 0)	(5, 0, 0)	(–5, 0, 0)	(–5, 0, 0)	(–5, 0, 0)
Initial angle of UAV (rad, rad, rad)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(π, 0, 0)	(π, 0, 0)	(π, 0, 0)
UAV flight end point (m, m, m)	(1000, –8.7, 0)	(1000, –8.7, 0)	(1000, –8.7, 0)	(–200, –8.7, 0)	(–200, –8.7, 0)	(–200, –8.7, 0)

where the airspace was limited and the UAV vertical separation distances were very small. Figure 4 shows that two swarms of approaching UAVs were present and their collisions were imminent. Unlike scenario 1, the UAV formations in each swarm were arranged in a vertical manner and this formation could lead to a collision domino effect when the UAV swarms met. Table 3 shows the initial state and the target status of each UAV.

The trajectory of each UAV is presented in the form of a 3D curve in Figure 4*a*. The green track represents the trajectory of the first UAV swarm and the red track represents that of the second swarm. In this simulation scenario, the UAVs generally adopted horizontal strategies for collision avoidance because vertical collision avoidance movements could have resulted in a domino effect. This scenario verified the anti-collision effect of self-organizing UAV clusters in a vertically stacked situation. Figure 4*b* shows the relative distances between each UAV pair, where the minimum distance between any two UAVs was 15.02 m, which was higher than the minimum safe separation distance.

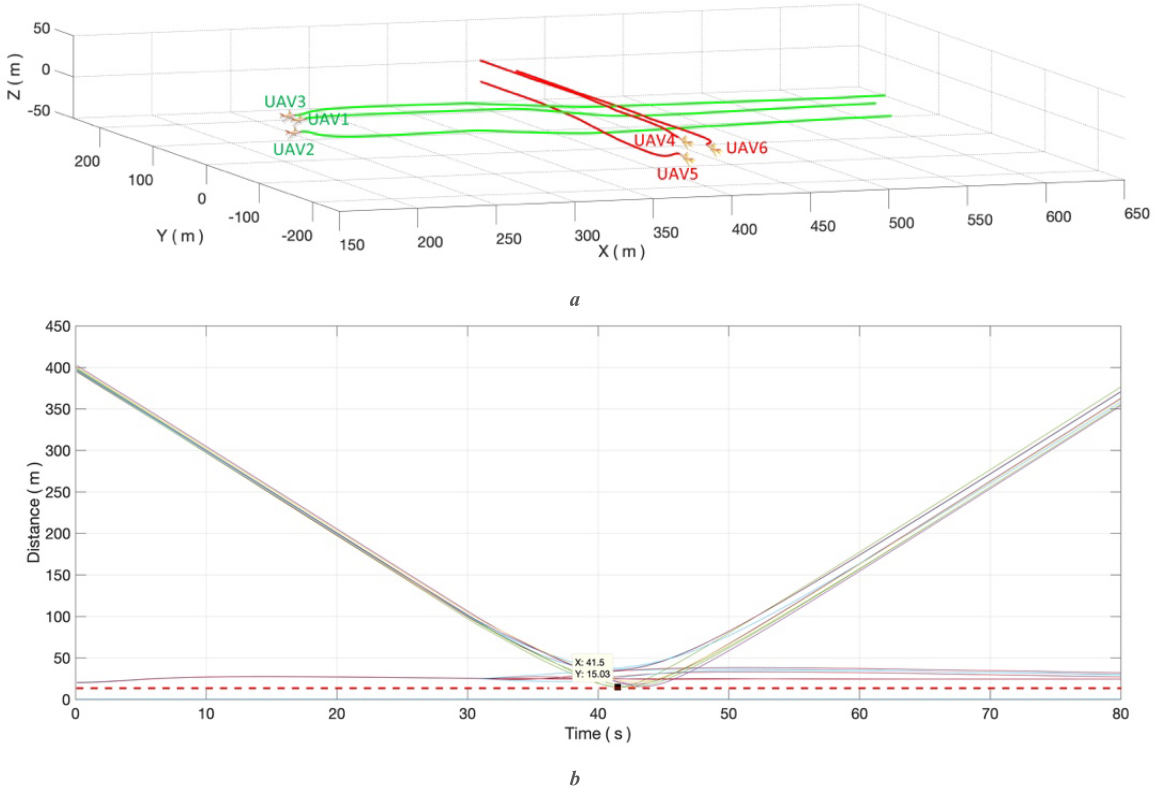


FIGURE 5. Simulation results for scenario three.

TABLE 4. Initial UAV states for the third flight scenario.

Number	UAV1	UAV2	UAV3	UAV4	UAV5	UAV6
Initial position of UAV (m, m, m)	(200, –17.3, 10)	(204, –17.3, –10)	(205, 0, 0)	(382.7, –200, 11)	(382.7, –202, –9)	(400, –205, 1)
Initial speed of UAV (m/s, m/s, m/s)	(5, 0, 0)	(5, 0, 0)	(5, 0, 0)	(0, 5, 0)	(0, 5, 0)	(0, 5, 0)
Initial angle of UAV (rad, rad, rad)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(π/2, 0, 0)	(π/2, 0, 0)	(π/2, 0, 0)
UAV flight end point (m, m, m)	(1000, –8.7, 0)	(1000, –8.7, 0)	(1000, –8.7, 0)	(400, 591.3, 0)	(400, 591.3, 0)	(400, 591.3, 0)

C. UAV SWARM VERTICAL CROSS FLIGHT

This scenario was designed to test whether UAV swarm collisions could be avoided effectively in the vertical direction. Figure 5 shows that two UAV swarms approached each other at a vertical angle and collisions were imminent. The simulation showed that the method could readily ensure collision avoidance in this scenario and the anti-collision trajectories were very smooth, thereby meeting the realistic flight requirements. This scenario supported the effectiveness of the anti-collision method when UAV swarms meet at a vertical angle. The initial state and target status for each UAV are shown in Table 4.

The trajectory of each UAV is represented by a 3D curve in Figure 5*a*. The green track represents the trajectory of the first UAV swarm and the red track is the trajectory of the second swarm. Figure 5*b* shows the relative distance between each pair of UAVs, where the minimum distance between any two UAVs was 15.04 m, which was above the minimum safe separation distance.

D. THREE DIFFERENT DIRECTIONS OF UAV SWARM GATHERING SCENE

This scenario tested the effect of the anti-collision method when several UAVs arrived in the same airspace from different directions. Figure 5 shows that three swarms of UAVs were flying from three different directions and collisions were imminent. This scenario tested whether the algorithm was still effective at avoiding collisions when the UAV swarms merged from three different directions. The simulation results showed that although the airspace density was high, the

TABLE 5. Initial UAV states for the fourth flight scenario.

Number	UAV1	UAV2	UAV3	UAV4	UAV5	UAV6	UAV7	UAV8	UAV9
Initial position of UAV (m, m, m)	(200, –17.3, 10)	(204, –17.3, –10)	(205, 0, 0)	(600, –17.3, 14)	(603, –17.3, 6)	(601, 0, 4)	(382.7, –200, 11)	(382.7, –202, –9)	(400, –205, 1)
Initial speed of UAV (m/s, m/s, m/s)	(5, 0, 0)	(5, 0, 0)	(5, 0, 0)	(–5, 0, 0)	(–5, 0, 0)	(–5, 0, 0)	(0, 5, 0)	(0, 5, 0)	(0, 5, 0)
Initial angle of UAV (rad, rad, rad)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(π, 0, 0)	(π, 0, 0)	(π, 0, 0)	(π/2, 0, 0)	(π/2, 0, 0)	(π/2, 0, 0)
UAV flight end point (m, m, m)	(1000, –8.7, 0)	(1000, –8.7, 0)	(1000, –8.7, 0)	(–200, –8.7, 0)	(–200, –8.7, 0)	(–200, –8.7, 0)	(400, 591.3, 0)	(400, 591.3, 0)	(400, 591.3, 0)

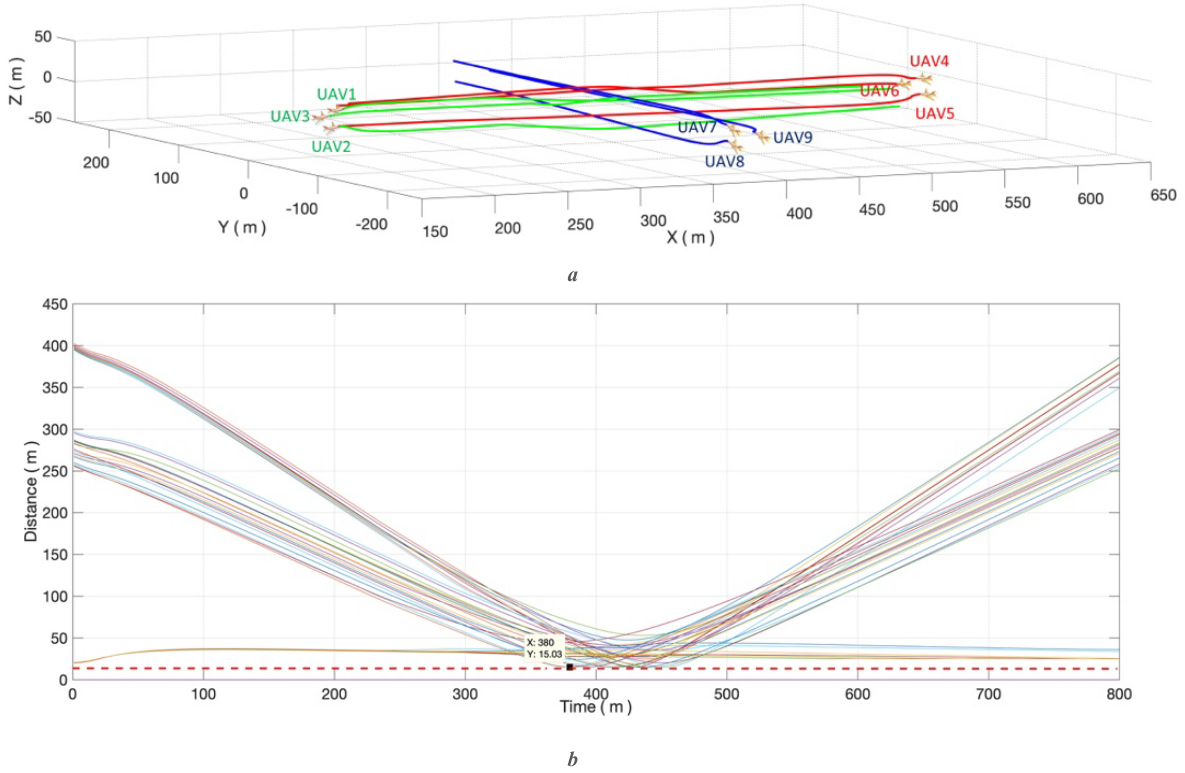


FIGURE 6. Simulation results for scenario four.

anti-collision effect was very good and the flight paths did not require excessive deviations, thereby meeting the realistic flight requirements. The initial state and target status of each UAV is shown in Table 5.

The trajectory of each UAV is represented by a 3D curve in Figure 6*a*. The green track represents the trajectory of the first UAV swarm, the red track is that for the second swarm, and the blue track is that for the third swarm. Figure 6*b* shows the relative distance between each pair of UAVs, where the minimum distance between any two UAVs was 15.03 m, which was higher than the minimum safe separation distance.

E. SIMULATION ANALYSIS AND COMPARISON OF VARIOUS SCENARIOS

Each simulation scenario showed that all of the UAVs were located greater than the safe distance, thereby demonstrating that collisions were successfully avoided in the drone swarm. The relative distance maps indicated that the collision avoidance effect was very good for the UAVs, and each UAV swarm continued to move toward the target position after completing the collision avoidance process. The relative distances of the UAVs in the same swarm or in different swarms were also maintained in a smooth manner during the entire collision avoidance process. However, in the fourth scenario, the relative distances of the UAVs at the CPA varied more than those in the first three scenarios because the airspace traffic density was very high in the same airspace at the moment corresponding to the CPA, and each UAV swarm was at risk of colliding with the UAV swarm in both directions.

The mixed state of the UAV swarm was observed using the relative distance curves for the UAVs and by assessing the topology of the UAV swarm from the side. For each scene,

two types of relative distance curves were obtained, where one tended to form a straight line and other tended to form a curve. The former represents the relative distances of the UAVs inside the swarm and the latter within the different swarms. The relative distance curve inside the swarm was smooth and it varies little, and thus the UAVs inside the swarm had a smooth and compact topology throughout the simulation.

V. CONCLUSION AND FUTURE WORK

In this study, the self-organized flight of UAV swarms was optimized based on the Reynolds rules. A new fitness function was proposed and the parameters were optimized using a CMA-ES optimization algorithm. A new cluster anti-collision algorithm was also proposed. The simulation results proved that the proposed self-organized UAV swarm flight collision avoidance method performed in a safe and effective manner with smooth avoidance trajectories, thereby meeting the realistic flight requirements. The main contributions of our study are summarized as follows.

1) Self-organizing flight rules were modeled for a UAV swarm based on the traditional Reynolds rules and suitable constraints were added. A fitness function was constructed to optimize the rules in the model to ensure that the self-organizing UAV swarm flight collision avoidance behavior was aligned with the actual flight requirements.
2) The traditional self-organizing flight rules are not effective for addressing the collision avoidance problem between UAV clusters, so we derived new anti-collision rules between UAVs to improve the collision avoidance efficiency and obtain smooth avoidance trajectories. In the simulations, the UAVs completed their anti-collision maneuvers without changing speed, which is suitable for meeting the actual flight requirements.
3) Our proposed method was verified in four simulation experiments, where UAV swarms approached face-to-face as well as extreme collision avoidance by UAV swarms in the vertical direction, vertical angle convergence to allow anti-collision by UAV swarms approaching in the horizontal direction, and collision avoidance by UAV swarms approaching the same airspace from three different directions.

In future research, we will consider factors such as communication delays to further optimize the rules model as well as the drone speed to make the model more flexible. In addition, actual flight experiments will be conducted using fixed-wing UAVs to validate our proposed method.

REFERENCES

[1] C. W. Reynolds, "Flocks, herds, and schools: A distributed behavioral model," in *Proc. ACM*, Aug. 1987, pp. 25–34.
[2] A. K. Pamosoaji and K.-S. Hong, "A path-planning algorithm using vector potential functions in triangular regions," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 43, no. 4, pp. 832–842, Jul. 2013. doi: 10.1109/TSMCA.2012.2221457.
[3] Y. C. Chang and Y. Yamamoto, "On-line path planning strategy integrated with collision and dead-lock avoidance schemes for wheeled mobile robot in indoor environments," *Ind. Robot*, vol. 35, no. 5, pp. 421–434, Aug. 2008. doi: 10.1108/01439910810893590.
[4] S. Perez-Carabaza, E. Besada-Portas, J. A. Lopez-Orozco, and J. M. de la Cruz, "Ant colony optimization for multi-UAV minimum time search in uncertain domains," *Appl. Soft Comput.*, vol. 62, pp. 789–806, Jan. 2017. doi: 10.1016/j.asoc.2017.09.009.
[5] H.-B. Duan, X.-Y. Zhang, J. Wu, and G.-J. Ma, "Max-min adaptive ant colony optimization approach to multi-UAVs coordinated trajectory replanning in dynamic and uncertain environments," *J. Bionic Eng.*, vol. 6, no. 2, pp. 161–173, Jun. 2009. doi: 10.1016/S1672-6529(08)60113-4.
[6] V. Y. Pehlivanoglu, "A new vibrational genetic algorithm enhanced with a Voronoi diagram for path planning of autonomous UAV," *Aerosp. Sci. Technol.*, vol. 16, no. 1, pp. 47–55, Jan./Feb. 2012. doi: 10.1016/j.ast.2011.02.006.
[7] O. K. Sahingoz, "Generation of bezier curve-based flyable trajectories for multi-UAV systems with parallel genetic algorithm," *J. Intell. Robotic Syst.*, vol. 74, nos. 1–2, pp. 499–511, Apr. 2014. doi: 10.1007/s10846-013-9968-6.
[8] Y. Liu, X. J. Zhang, X. M. Guan, and D. Delahaye, "Adaptive sensitivity decision based path planning algorithm for unmanned aerial vehicle with improved particle swarm optimization," *Aerosp. Sci. Technol.*, vol. 58., pp. 92–102, Nov. 2016. doi: 10.1016/j.ast.2016.08.017.
[9] J. L. Foo, J. Knutzon, and V. Kalivarapu, "Path planning of unmanned aerial vehicles using b-splines and particle swarm optimization," *J. Aerosp. Comput., Inf., Commun.*, vol. 58, pp. 271–290, Nov. 2009. doi: 10.2514/1.36917.
[10] S. Ragi and E. K. P. Chong, "UAV path planning in a dynamic environment via partially observable Markov decision process," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 49, no. 4, pp. 2397–2412, Oct. 2013. doi: 10.1109/TAES.2013.6621824.
[11] M. Li, R. Z. Chen, W. L. Zhang, and D. Li, "A stereo dual-channel dynamic programming algorithm for UAV image stitching," *Sensors*, vol. 17, no. 9, p. 2060, Sep. 2017. doi: 10.3390/s17092060.
[12] M. Radmanesh and M. Kumar, "Flight formation of UAVs in presence of moving obstacles using fast-dynamic mixed integer linear programming," *Aerosp. Sci. Technol.*, vol. 50, pp. 149–160, Mar. 2016. doi: 10.1016/j.ast.2015.12.021.
[13] M. De Benedetti and F. D'Urso, G. Fortino, F. Messina, G. Pappalardo, and C. Santoro, "A fault-tolerant self-organizing flocking approach for UAV aerial survey," *J. Netw. Comput. Appl.*, vol. 96, pp. 14–30, Oct. 2017. doi: 10.1016/j.jnca.2017.08.004.
[14] H. Qiu and H. Duan, "Pigeon interaction mode switch-based UAV distributed flocking control under obstacle environments," *ISA Trans.*, vol. 71, pp. 93–102, Nov. 2017. doi: 10.1016/j.isatra.2017.06.016.
[15] B. Di, R. Zhou, and H. Duan, "Potential field based receding horizon motion planning for centrality-aware multiple UAV cooperative surveillance," *Aerosp. Sci. Technol.*, vol. 46, pp. 386–397, Oct./Nov. 2015. doi: 10.1016/j.ast.2015.08.006.
[16] R. G. Braga, R. C. da Silva, A. C. B. Ramos, and F. Mora-Camino, "Collision avoidance based on reynolds rules: A case study using quadrotors," in *Proc. Inf. Technol.—New Generations*, 2018, pp. 773–780. doi: 10.1007/978-3-319-54978-1_96.
[17] C. Kownacki and D. Ołdziej, "Flocking algorithm for fixed-wing unmanned aerial vehicles," in *Proc. Adv. Aerosp. Guid., Navigat. Control*, 2015, pp. 415–431. doi: 10.1007/978-3-319-17518-8_24.
[18] J. B. Clark and D. R. Jacques, "Flight test results for UAVs using boid guidance algorithms," *Procedia Comput. Sci.*, vol. 8, pp. 232–238, Dec. 2012. doi: 10.1016/j.procs.2012.01.048.
[19] M. Sajwan, D. Gosain, and S. Surani, "Flocking behaviour simulation: Explanation and enhancements in boid algorithm," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 4, p. 5539, Jul. 2014.
[20] S. A. P. Quintero, G. E. Collins, and J. P. Hespanha, "Flocking with fixed-wing UAVs for distributed sensing: A stochastic optimal control approach," in *Proc. Amer. Control Conf.*, Jun. 2013, pp. 2025–2031.
[21] J. Espelosín, L. Acosta, and D. Alonso, "Path planning approach based on flock dynamics of moving particles," *Appl. Soft Comput.*, vol. 13, no. 4, pp. 2159–2170, Apr. 2013. doi: 10.1016/j.asoc.2012.12.015.
[22] S. Hauert, S. Leven, M. Varga, F. Ruini, A. Cangelosi, J.-C. Zufferey, and D. Floreano, "Reynolds flocking in reality with fixed-wing robots: Communication range vs. maximum turning rate," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, San Francisco, CA, USA, Sep. 2011, pp. 25–30.

[23] H. Hildenbrandt, C. Carere, and C. K. Hemelrijk, "Self-organized aerial displays of thousands of starlings: A model," *Behav. Ecology*, vol. 21, no. 6, pp. 1349–1359, Nov./Dec. 2009. doi: 10.1093/beheco/arq149.
[24] J. Han, M. Li, and L. Guo, "Soft control on collective behavior of a group of autonomous agents by a shill agent," *J. Syst. Sci. Complex.*, vol. 19, no. 1, pp. 54–62, Mar. 200. doi: 10.1007/s11424-006-0054-z.
[25] G. Vásárhelyi, C. Virágh, G. Somorjai, T. Nepusz, A. E. Eiben, and T. Vicsek, "Optimized flocking of autonomous drones in confined environments," *Sci. Robot.*, vol. 3, no. 20, Jul. 2018, Art. no. eaat3536. doi: 10.1126/scirobotics.aat3536.

YANG HUANG received the B.S. and M.S. degrees in information and communication engineering from the National University of Defense Technology, Changsha, China, in 2013 and 2017, respectively, where he is currently pursuing the Ph.D. degree with the Science and Technology on Information Systems Engineering Laboratory. During the B.S. degree, he studied the system engineering and the operation research. After finishing the B.S. degree, he began to study the self-organizing flight theory of UAVs and the collision avoidance problem of UAV clusters. His research interests include the air traffic management and unmanned aerial vehicle systems, such as collision avoidance of multi-UAV systems, especially the self-organized UAV collision avoidance problem.

JUN TANG is currently an Assistant Professor with the Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology. He was dedicated to the Ph.D. researches with the Technical Innovation Cluster on Aeronautical Management, Universitat Autònoma de Barcelona. His research interests include logistic systems, causal modeling, state space, air traffic management, and discrete event simulation. He is the winner of William Sweet Smith Prize, in 2015. He has been very active in the simulation community, organizing as the General Co-Chair for several international conferences. He acted as one of the main participants playing an important role in the FP7 European Project INnovative TEchnologies and Researches for a new Airport Concept toward Turnaround coordinatION (INTERACTION) with collaboration of Airbus.

SONGYANG LAO received the B.S. degree in information system engineering and the Ph.D. degree in system engineering from the National University of Defense Technology, Changsha, China, in 1990 and 1996, respectively. He joined as a Faculty Member with the National University of Defense Technology, in 1996, where he is currently a Professor with the School of Information System and Management. He was a Visiting Scholar with Dublin City University, Ireland, from 2004 to 2005. His current research interests include image processing and video analysis, and human–computer interaction.