Article

Starling-Behavior-Inspired Flocking Control of Fixed-Wing Unmanned Aerial Vehicle Swarm in Complex Environments with Dynamic Obstacles

Weihuan Wu 1,2, Xiangyin Zhang 1,2,* and Yang Miao 3

1 Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China
2 Engineering Research Center of Digital Community, Ministry of Education, Beijing 100124, China
3 Faculty of Materials and Manufacturing, Beijing University of Technology, Beijing 100124, China
* Correspondence: xy_zhang@bjut.edu.cn

Abstract: For the sake of accomplishing the rapidity, safety and consistency of obstacle avoidance for a large-scale unmanned aerial vehicle (UAV) swarm in a dynamic and unknown 3D environment, this paper proposes a flocking control algorithm that mimics the behavior of starlings. By analyzing the orderly and rapid obstacle avoidance behavior of a starling flock, a motion model inspired by a flock of starlings is built, which contains three kinds of motion patterns, including the collective pattern, evasion pattern and local-following pattern. Then, the behavior patterns of the flock of starlings are mapped on a fixed-wing UAV swarm to improve the ability of obstacle avoidance. The key contribution of this paper is collective and collision-free motion planning for UAV swarms in unknown 3D environments with dynamic obstacles. Numerous simulations are conducted in different scenarios and the results demonstrate that the proposed algorithm improves the speed, order and safety of the UAV swarm when avoiding obstacles.

Keywords: starlings; collective motion; fixed-wing UAV swarm; local following; obstacle avoidance

1. Introduction

Group formation is widespread, such as in schools of fish, flocks of birds and swarms of honeybees when they perform seasonal migration, travel to food sources or avoid predators [1–3]. A paradigmatic case of flocks of birds is provided by starlings [4,5], which are a highly gregarious species. Grouping has several benefits: it reduces the chance of an individual being attacked and decreases the risk of being exposed to hazards. Collective motions in flocks have intrigued scientists in various research fields, including control engineering and computer science. The study of collective motion began in the 1980s with the development of computer simulations. In 1987, Reynolds proposed a classical collective model [6] (called the Boid model), which is consisted of three rules: (1) Separation: each agent avoids colliding with nearby neighbors; (2) Cohesion: each agent stays close to nearby neighbors; (3) Alignment: each agent matches velocity with nearby neighbors. In 1995, Vicsek proposed a collective model [7] (called the Vicsek model) with self-propelled particles driven by constant velocity. The direction of particles is obtained by calculating the average direction of their neighbors. The Vicsek model accomplishes the self-organization and velocity consensus of multi-agent systems. In 2002, Couzin proposed a collective model [8] in three-dimensional space that divides the perceptual area of agents into three different parts. These three parts are the exclusion area, the alignment area and the attractive area, which correspond to the separation, cohesion and alignment rules. In 2006, Olfati-Saber presented a flocking algorithm [9] for free-flocking and constrained flocking. Each agent in constrained flocking has a control input consisting of a formation term, an obstacle avoidance term and a lead-following term. The flocking algorithm results in self-organized flocking behavior. In 2007, Tanner proposed a flocking algorithm [10] in...
fixed and switching networks. The network of multi-agent systems remains constantly connected regardless of switching topology. In the last decade, researchers discovered that most collective behavior of animals has the following advantages: decentralization, following the sample rules, flexibility. Therefore, a great deal of research is also inspired by the collective behavior of animals. Duan et al. [11] presented a novel swarm intelligence optimizer based on the collective behavior of pigeons, after which authors applied this pigeon-inspired optimization for solving air robot path planning problems. Zhang et al. [12] proposed an enhanced fruit fly optimization algorithm based on quantum theory and the collective behavior of fruit flies. In addition, the proposed algorithm was also adopted to unmanned aerial vehicle path planning problems in the three-dimensional environment. Zhou et al. [13] explored the problem of bionic flight control in an aircraft formation. By analyzing the formation flight mechanism of wild geese during migration, this research work designed a tight configuration for formation and a control method for multiple aircrafts. Xie et al. [14] proposed a starling swarm coordination algorithm inspired by the collective behavior of starlings. The algorithm adopted the thought of decentralization and self-organization to realize the behavior evolution process of collective motion from disorder to order.

In recent years, the unmanned aerial vehicle (UAV) has become widely used in military and civilian fields such as agriculture by virtue of the development of autonomy [15–17]. Compared with a single UAV, multiple UAVs are able to execute missions such as surveillance and reconnaissance simultaneously, which greatly improves execution efficiency [18]. Obstacles are one of the biggest threats to flight for multi-agent systems when executing missions. Most existing methods of obstacle avoidance are divided into known environment and unknown environment according to whether there is prior information. The typical motion planning method in a known environment includes the A* search algorithm [19], particle swarm optimization (PSO) [20,21] algorithm and model prediction control (MPC) method [22]. The reactive method for obstacle avoidance includes the artificial potential field (APF) method [23,24]. These methods are intuitive and convenient to implement, but the generated path may fall into local minima in some cases. Additionally, a deep-reinforcement-learning-based reactive online decision-making mechanism is applied in [25] to figure out the problem of obstacle avoidance, but this research only involves a single UAV. Some research [26–28] only considers a sparse obstacle environment or only stays in a two-dimensional environment, which is a disparate scenario compared with the environment in which a UAV swarm executes missions in reality. Unlike an insect swarm or a terrestrial animal (e.g., horse or wolf) swarm that consist of hundreds of individuals, a flock of starlings always consists of thousands of members, which makes starling flocks more valuable to explore. The collective motion of starlings emerges as the result of the local interaction between nearby neighbors [29], instead of relying on centralized coordination. Additionally, a flock of starlings is able to evade threats in an orderly fashion [30]. In groups, potential threats may be spotted earlier and information may be spread faster than among solitary individuals [31]. Enlightened by the collective evasion of starlings [30,32], a turning control is adopted to avoid obstacles in this paper. The UAV swarm turning control denotes that after a UAV in the group has turned, other UAVs reach a consensus state of turning to avoid threats through their perception and judgment even though they have not detected the threat. Therefore, fast information transmission in the UAV swarm is vital for obstacle avoidance.

In this paper, a control framework is proposed to solve the obstacle avoidance problem of a UAV swarm in an unknown environment with dynamic threats. This distributed control framework mimics behavioral patterns of the flocking of starlings when they evade threats. In order to improve the performance of the UAV swarm, three major patterns of flocks of starlings have been applied to the UAV swarm. Based on these patterns, the UAV swarm can aggregate and reach a consistent state immediately in the environment without obstacles; when the UAV swarm encounters obstacles, the turning curvature of the path and the consistency of the UAV swarm can be satisfied simultaneously.
The rest of the paper is organized as follows. Section 2 discusses the behavior mechanism of starlings and presents the behavioral patterns of the flock of starlings in detail. Section 3 defines the model of the UAV and mapping behavioral patterns on the UAV swarm. In Section 4, simulations and comparisons are depicted to demonstrate the effectiveness of the proposed algorithm. Conclusions are drawn in Section 5.

2. Model of Starling Behavior
2.1. The Behavioral Mechanism of Starlings

When a flock of starlings confronts threats, individuals near the threat become alert and take action. Because the flock of starlings is too large, it is inevitable that individuals far from the threat will not be aware of danger in the first place. Therefore, fast transmission of information in groups can increase the survival rate of the whole flock. Those individuals that detect threats earlier are selected as local leaders of their neighbor unit to transmit information [33]. The information transmission appears as a wave of agitation in the starling flock [34]. As shown in Figures 1 and 2, the movement of a dark band called an agitation wave generates from the position of the threat and propagates away from it. Biological research about collective behavior indicates that individuals on the far side of the group will be alerted earlier than if they were alone.

![Dark band](image1)

Figure 1. A flock of starlings in the sky [35] copyright: 543476037420.

![Dark band](image2)

Figure 2. Schemes follow the same formatting [5] copyright: © 2015, The Author(s).

Flocks of starlings have the capability to gather immediately through local information interaction. In the starling flock, the orientation of each starling is not only decided by their self-information but also decided by their neighbors [29]. Each starling reaches the velocity consensus with its neighbors, so that the entire flock will form a uniform movement.
As shown in Figure 3, a flock of starlings will change orientation to avoid collision with threats. The geometric shape of the flock on the plane remains approximately constant during turns.

![Figure 3. Schematic of the turning flock on the horizontal plane.](image)

2.2. Behavioral Patterns Based on Behavioral Mechanisms of Starlings

Based on analyses in Section 2.1, there are three major patterns in flocks of starlings [36]: the collective pattern, evasion pattern and local-following pattern. In obstacle-free areas, the tendency of each starling to imitate its neighbors produces a global collective state, i.e., the flock of starlings in the collective pattern. In the obstacle environment, starlings that have detected obstacles will switch to the evasion pattern to avoid obstacles, while those starlings that are not aware of threats will follow the starlings that have switched to the evasion pattern [36]. Fast information transmission allows coordination of motion and effective response to the complex environment.

2.2.1. Collective Pattern

In reality, constrained by a limited sensing ability, starlings can only communicate with neighbors in a fixed distance (as shown in Figure 4, the red marker represents the individual \(i\) and pink markers represent neighbors of individual \(i\)). The neighbor set of the individual \(i\) is denoted by

\[
N_i(t) = \{ j : \|x_i - x_j\| \leq R, j = 1, 2, \ldots, N, j \neq i \}
\]

(1)

where \(x_i, x_j \in \mathbb{R}^3\) are the position of individual \(i\) and \(j\), respectively. \(\|\cdot\|\) represents the Euclidean distance and \(R\) is the sensing radius of each individual.

![Figure 4. Schematic of the neighbor set, the red marker represents the individual \(i\) and pink markers represent neighbors of individual \(i\).](image)

According to research and data about biological collective behavior, an individual in a flock of starlings also follows cohesion, alignment and separation rules. Each starling
shares velocity and location information with its neighbors to decide the actions that need to be performed at that moment, which can be described as follows:

\[ f^\text{col}_i = f^\text{pos}_i + f^\text{vel}_i \]  

where the neighbor interaction term \( f^\text{col}_i \in \mathbb{R}^3 \) consists of the position coordination term \( f^\text{pos}_i \in \mathbb{R}^3 \) and the velocity coordination term \( f^\text{vel}_i \in \mathbb{R}^3 \).

If a flock of starlings gathers too closely, individuals will repel each other to avoid collision. However, starlings will also gather more closely when the distance between two neighbors is too large. Thus, the position coordination term is described as follows:

\[ f^\text{pos}_i = \sum_{j \in N(i)} \left( \frac{1}{\|x_i - x_j\|} - \frac{1}{\|x_i - x_j\|^3} \right) \]  

The alignment rule states that the velocity of a starling should keep consistent with its neighbors to guarantee the order of the flocking system. Thus, the velocity alignment term is defined as follows:

\[ f^\text{vel}_i = -\sum_{j \in N(i)} (v_i - v_j) \]  

where \( v_i, v_j \in \mathbb{R}^3 \) are the velocity of individual \( i \) and \( j \), respectively.

2.2.2. Evasion Pattern

When a flock of starlings encounters obstacles, individuals avoid colliding with obstacles by changing their orientation, which is called the tangential navigation schema [37]. As shown in Figure 5, the dark grey circled area represents the real obstacle while the radius of the light grey circle area is the expected distance from individual \( i \) to the center of obstacle. The red arrows in Figure 5b represent the motion direction of the individual \( i \). The velocity vector \( v_i \) of the individual \( i \) is in the area UAB, which indicates that if the individual \( i \) keeps the original heading angle \( \theta_i^{-1} \), a collision with the obstacle will happen.

![Figure 5](image_url)

**Figure 5.** Diagram of the threat. (a) is in x-y plane, while (b) is in the 3D space, the dark grey circled area represents the real obstacle while the radius of the light grey circle area is the expected distance from individual \( i \) to the center of obstacle. The red arrows in (b) represent the motion direction of the individual \( i \).

Consequently, the individual \( i \) will align its heading angle to the tangent direction of the virtual scope of the obstacle. Predefine a parameter to describe the direction of rotation as follows:

\[ \epsilon_r = \begin{cases} 
-1, & \text{for clockwise rotation} \\
1, & \text{for counter-clockwise rotation} 
\end{cases} \]
Supposing that the turning angle of individual $i$ is $\Delta \theta_i$, the new heading angle pointed at the desired point can be calculated as follows:

$$\theta_i^d = \theta_i^{d \leftarrow 1} + c_r \cdot \Delta \theta_i$$  \hspace{1cm} (6)

The position of virtual point $x_i^d \in \mathbb{R}^3$ is a projection of the position of the desired point $x_i^d \in \mathbb{R}^3$ on the $x$-axis. So, $x_i^d$ is calculated as follows:

$$x_i^d = x_i + \begin{bmatrix} \cos \theta_i^d & -\sin \theta_i^d & 0 \\ \sin \theta_i^d & \cos \theta_i^d & 0 \\ 0 & 0 & 1 \end{bmatrix} (x_i^d - x_i)$$  \hspace{1cm} (7)

Therefore, the pattern of evasion is donated as follows:

$$f_i^{ev} = -\frac{x_i - x_i^d}{\|x_i - x_i^d\|}$$  \hspace{1cm} (8)

2.2.3. Local-Following Pattern

In the free motion of the flocking system, the information flow is mainly conducted by averaging states of all the nearby neighbors. However, when an emergency occurs, a more efficient method to reduce the risk is to follow a single local leader within the neighbor set [38,39]. The single local leader is the individual that has detected threats and has already taken action (as shown in Figure 6).

![Figure 6. Schematic of the selection of local leader, the red marker represents the individual $i$ and pink markers represent neighbors of individual $i$.](image)

In order to select an individual as the local leader, the influence of neighbor $j$ on individual $i$ (the evaluation index) needs to be calculated as follows:

$$C_{ij} = c_1 \frac{1}{\|x_i - x_j\|} \cdot c_2 \left( \frac{\vec{v}_i}{\|\vec{v}_i\|} - \frac{\vec{v}_j}{\|\vec{v}_j\|} \right)$$  \hspace{1cm} (9)

where $c_1$ and $c_2$ are the coefficients of the influence of position and velocity, respectively. According to the value of the evaluation index, select the individual $l_i$ with the most drastic change in motion state in the neighbor set $N_i(t)$ as follows:

$$l_i = \{ \max_{C_{ij}, C_{ij} \geq C^*} \}$$  \hspace{1cm} (10)

In addition, a threshold value of $C^*$ is designed to decide whether to switch to the local-following pattern:

$$C^* = e^{-a\phi_i}$$  \hspace{1cm} (11)

where $\phi_i$ is the order parameter of individual $i$ and its neighbors. The coefficient $a$ is able to adjust the sensitivity of the group to the order parameter. Indeed, the larger the value of
α is, the more sensitive of the flock is, which indicates that the flock is more likely to adopt the local-following pattern. The definition of $\phi_i$ is as follows:

$$\phi_i = \frac{1}{N_i + 1} \left| \sum_{j=1}^{N_i} \frac{v_j}{||v_j||} \right|$$

(12)

where $N_i$ is the number of individuals in the neighbor set $N_i(t)$.

Thus, the pattern of local-following is presented as follows:

$$f_i^{lof} = -(v_i - v_l)$$

(13)

3. Starling-Behavior-Inspired Flocking Control for UAV Swarm

In this section, the model of the fixed-wing UAV is set up first. The rest of section describes the interaction among fixed-wing UAVs and the obstacle avoidance process that mimic flocks of starlings in detail. In order to simplify the setting of the model, suppose that each UAV is equipped with a GPS, a wireless communication device and a sensor measuring distance (assume that the measurements are without noise), so that each UAV is able to locate and communicate with other UAVs within the sensing radius, as well as detect obstacles.

3.1. Model of UAV

Consider a multi-UAV system with $n$ UAVs. Let $x_i, y_i, h_i, \psi_i, V_i$ denote the $x$ position, $y$ position, altitude, heading angle and forward velocity of $i$-th ($i = 1, 2, \ldots, n$) UAV in the global coordinate system, respectively. The basic kinematic model of a fixed-wing UAV in a three-dimensional space is described as follows:

$$\dot{x}_i = V_i \cos \psi_i$$

$$\dot{y}_i = V_i \sin \psi_i$$

$$\dot{V}_i = \frac{1}{\tau_v} (V_c^f - V_i)$$

$$\dot{\psi}_i = \frac{1}{\tau_{\psi}} (\psi_c^f - \psi_i)$$

$$\dot{h}_i = -\frac{1}{\tau_h} \dot{h}_i + \frac{1}{\tau_h} (h_c^f - h_i)$$

(14)

where $V_c^f, \psi_c^f, h_c^f$ are the command inputs of velocity, heading angle and altitude to the corresponding autopilots, respectively. $\tau_v, \tau_{\psi}, (\tau_h, \tau_h)$ are positive time constants for the velocity, heading angle and altitude response with respect to the corresponding command inputs, respectively.

In a practical setting, a real fixed-wing UAV’s $V_i, \psi_i, h_i$ should satisfy the following constraints:

$$v_{\min} \leq V_i \leq v_{\max}$$

$$|\psi_i| \leq \frac{n_{\max}}{V_i}$$

$$\lambda_{\text{glide}} \leq \dot{h}_i \leq \lambda_{\text{climb}}$$

(15)

where $v_{\min}, v_{\max}$ are the minimum and maximum forward velocity; $n_{\max}$ is the maximum lateral overload; $\lambda_{\text{climb}} > 0$ is the maximum climbing velocity; $\lambda_{\text{glide}} < 0$ is the minimum gliding velocity; $g = 10 \text{ m/s}^2$ is the gravitational acceleration. Due to constraints of velocity, the fixed-wing UAV cannot hover or fly backwards.

3.2. Collision Prediction Mechanism

In a multi-obstacle environment, there may exist an obstacle that will not be collided with if the UAVs do not change their orientation, which implies that the UAVs need not take action to avoid this obstacle. Therefore, a collision prediction mechanism is introduced to predict whether a collision with an obstacle will occur. Based on assumptions made at the
begin of this section, a UAV $i$ is able to sense the distance between itself and an obstacle as follows:

$$ R_{i}^{obs} = \|q_i - q_{m}^{obs}\| $$

(16)

where $q_{m}^{obs} \in \mathbb{R}^{3}$ is the position of $m$-th obstacle ($m = 1, 2, \ldots, k$, $k$ is the number of obstacles).

When the desired distance $R_{d}$ between the UAV and obstacle is less than $R_{i}^{obs}$, the collision prediction will start. As shown in Figure 7, the virtual points A and B are the edge points of scope of the obstacle. Supposing that the radius of the virtual zone of the obstacle is $r_{obs}$, the mathematical description of the obstacle effect region is as follows:

$$ UA = \sqrt{US^{2} - SA^{2}} $$

$$ \Delta \phi = \arctan\frac{SA}{UA} $$

(17)

where $UA$, $US$ and $SA$ represent the length of line segment UA, US and SA, respectively.

![Figure 7](image_url)

**Figure 7.** The collision prediction between the UAV and obstacle. (a) is in the 3D space, while (b) is in the $x$-$y$ plane, the red marker represents the individual $i$, the virtual points A and B are the edge points of scope of the obstacle, $UA$, $US$ and $SA$ represent the length of line segment UA, US and SA, respectively.

If the heading angle $\psi_i$ of the UAV $i$ in the range of $[\phi - \Delta \phi, \phi + \Delta \phi]$, it is necessary to consider the impact of the obstacle. Therefore, the UAV $i$ needs to take action to avoid this obstacle.

### 3.3. Mapping of the Intelligent Behavioral Patterns of Starlings

Suppose a system that consists of $n$ UAVs in a three-dimensional Euclidean space. The dynamics of the UAVs are modeled as second-order integrators as follows:

$$ \begin{cases} \dot{q}_i = p_i \\ p_i = u_i, i = 1, 2, \ldots, n \end{cases} $$

(18)

where $q_i$, $p_i$ and $u_i \in \mathbb{R}^{3}$ denote the position, velocity and control input vectors of the $i$-th UAV, respectively.

UAVs select their neighbors according to the principle called nearest-neighbor distance (as shown in Equation (1)). The desired geometric model of the flock requires that each UAV in the group keep the same distance from all its neighbors:

$$ \|q_i - q_j\| = d, \forall i, j \in N_i(t) $$

(19)

where $d$ is a positive constant that represents the distance between UAV $i$ and UAV $j$. 
Based on the flocking algorithm [9] and behavioral pattern of starlings [36], the cooperative control law in a multi-obstacle environment is designed as follows:

\[ u_i = f^{el}_i + f^{obs}_i + f^{reg}_i \]  

where \( f^{el}_i \) is used to control UAVs to maintain the flocking geometry; \( f^{obs}_i \) is adopted to control UAVs to avoid obstacles; \( f^{reg}_i \) is applied to control UAVs to follow the navigation function.

Mapping the behavioral pattern mechanisms of starlings on a fixed-wing UAV. The formation of the UAV swarm is inspired by the collective patterns of starlings, which is described as follows:

\[
f^{el}_i = c_3 \sum_{j \in N_i(t)} \left( \frac{1}{\|q_i - q_j\|} - \frac{1}{\|q_i - q_{j_L}\|} \right) - c_4 \left( \frac{\beta}{\sum_{j \in N_i(t)} \|q_i - q_j\|} \right) - \beta \sum_{j \in N_i(t)} (p_i - p_j) + (1 - \beta) \cdot (p_i - p_{l_i}) \]  

where \( c_3, c_4 \) are positive coefficients; \( p_{l_i} \in \mathbb{R}^3 \) is the velocity of the local leader in the neighbor set (refer to Equations (9)–(12) to select a local leader); \( \beta \) is the weight coefficient. If there is a local leader, individuals in the neighbor set will only follow the local leader and ignore the influence of other neighbors (\( \beta = 0 \)). Otherwise, the individual considers the movements of all the neighbors in the neighbor set (\( \beta = 1 \)).

The evasion pattern of the UAV is used to avoid an obstacle as follows:

\[
f^{obs}_i \xrightarrow{\text{mapping}} f^{obs}_i = -\gamma \cdot c_5 \left( \frac{q_{l_i} - q_i}{\|q_{l_i} - q_i\|} \right) \]  

where \( c_5 \) is a positive control gain. When the UAV \( i \) is avoiding the obstacle, \( \gamma = 1 \). Otherwise, \( \gamma = 0 \).

A virtual leader is set for the sake of making all UAVs move together. The navigational tracking force is introduced as follows:

\[
f^{reg}_i = -(1 - \gamma) \cdot \left[ c_6(q_i - q_{l_L}) + c_7(p_{l_L} - p_i) \right] \]  

where \( c_6, c_7 \) are positive coefficients; \( q_{l_L}, p_{l_L} \in \mathbb{R}^3 \) are the position and velocity of the virtual leader, respectively.

As illustrated in Figure 8, the control framework for the obstacle avoidance of the UAV swarm is composed of two major parts, the fixed-wing UAV model and the proposed algorithm, and two auxiliary parts, the control instruction solver and the state converter.

**Figure 8.** The block diagram of the control system for the UAV swarm.
The control instruction solver is used to transfer the control signal vector \( u_i = [u_i^x \quad u_i^y \quad u_i^h]^T \) to the control command of autopilots \( [V_i^c \quad \psi_i^c \quad h_i^c]^T \). The specific solution equation is as follows:

\[
\begin{bmatrix}
V_i^c \\
\psi_i^c \\
h_i^c
\end{bmatrix} =
\begin{bmatrix}
\tau_v \cos \psi_i & \tau_v \sin \psi_i & 0 \\
-\tau_v \sin \psi_i & \tau_v \cos \psi_i & 0 \\
0 & 0 & \tau_h
\end{bmatrix}
\begin{bmatrix}
u_i^x \\
u_i^y \\
u_i^h
\end{bmatrix}
+ 
\begin{bmatrix}
V_i \\
\psi_i \\
h_i + \frac{\tau_h}{\tau_v}
\end{bmatrix}
\]

(24)

The state converter is defined as follows:

\[
\begin{bmatrix}
q_i \\
p_i
\end{bmatrix} =
\begin{bmatrix}
x_i \\
y_i \\
h_i
\end{bmatrix}
\]

(25)

The pseudo-code of the starling-behavior-inspired flocking control algorithm for UAVs is shown in Algorithm 1.

Algorithm 1. Starling-behavior-inspired flocking control algorithm for UAVs

/*Initialization*/
Set initial parameters of the proposed algorithm and the model of fixed-wing UAV
Generate the position \( x_i, y_i, h_i \), heading angle \( \psi_i \) and velocity \( V_i, h_i \) of UAV \( i \) randomly

/*Begin*/
for \( i = 1 \) to \( n \)
   for \( j = 1 \) to \( n \)
      Select neighbors according to Equation (1)
      if UAV \( j \) is the neighbor of UAV \( i \)
         UAV \( i \) interact with UAV \( j \) according to Equation (21)
         Calculate \( C_{ij} \) according to Equation (9)
      end if
      Calculate \( \phi_i \) according to Equation (12)
      Find the local leader \( l_i \) according to Equations (10) and (11)
   end for
   Calculate \( R_{obs}^i \) to Equation (16)
   if \( R_d \leq R_{obs}^i \)
      Execute collision prediction according to Equation (17)
      if \( \phi - \Delta \phi \leq \theta_i \leq \phi + \Delta \phi \)
         Set parameter \( \gamma = 1 \)
         Calculate the control signal of evasion pattern of UAV \( i \) according to Equation (22)
      end if
      if there exist a local leader \( l_i \)
         Set parameter \( \beta = 0 \)
         UAV \( i \) follows the local leader according to Equation (21)
      end if
   end if
   Follow the virtual leader according to Equation (23)
/*Limitation*/
Set limitations according to Equation (15)
Update the position \( q_i \) and velocity \( p_i \) according to Equations (24) and (25)
end for

3.4. Conversion of Patterns

In Section 2, behavioral patterns that emerge in a flock of starlings have been described carefully. However, when mapping behavioral patterns on a swarm of fixed-wing UAVs, the conversion of these patterns is essential to maintain the motion stability of the swarm of fixed-wing UAVs. In a state without external stimuli, the UAV swarm will be in the collective pattern. When a UAV has entered the scope of an obstacle (i.e., \( R_d \leq R_{obs}^i \)), the collision prediction mechanism is applied to predict whether the UAV will collide with the
obstacle or not. If the collision prediction mechanism has determined that the UAV will hit the obstacle (i.e., $\phi - \Delta \phi \leq \psi_i \leq \phi + \Delta \phi$), this UAV will activate the evasion pattern to avoid the obstacle. In a neighbor unit, the UAV performing a sudden turning action (which indicates this UAV may have detected threats, i.e., $\max C_{ij} \geq C^*$) will be selected as a local leader to lead other UAVs in this unit to avoid potentially dangerous obstacles.

A brief figure about conversion of patterns of a single UAV $i$ is depicted in Figure 9.

**Figure 9.** The conversion of patterns.

### 4. Simulation Results and Analysis

In this section, several simulation results of a fixed-wing UAV swarm in a 3D obstacle environment have been presented. For the performance assessment of the proposed algorithm, the basic flocking algorithm [9] (the method of obstacle avoidance is APF) and the proposed algorithm are tested in an obstacle-dense environment, a dynamic obstacle environment and an obstacle environment with both static and dynamic obstacles. Controlled objects used in all simulations are a UAV swarm consisting of $n = 50$ small fixed-wing UAVs. Each UAV has a range sensor with a limited measuring range $R$ without having prior information about the experimental scenario. The UAV swarm tracks the trajectory of the virtual leader to reach the destination. The start point of the virtual leader is $q_1(0) = [0 \, 0 \, 0]^T$. The initial states are set randomly as $x_i(0) \in [0, 10]$, $y_i(0) \in [0, 10]$, $v_i(0) \in [0, 10]$, $\dot{h}_i(0) \in [0, 1]$, $\psi_i(0) \in [-\pi/2, \pi/2]$.

| Parameter | Value |
|-----------|-------|
| Sensing range, $R$ | 15 m |
| Desired distance between neighboring UAVs, $d$ | 5 m |
| Desired distance between UAV and obstacle, $R_d$ | 15 m |
| Coefficient for the selection of local leader, $\alpha$ | 2.5 |
| Control gains of local-following pattern, $c_1, c_2$ | 0.5, 2 |
| Step-size, $\Delta t$ | 0.025 s |
| Control gains of collective pattern, $c_3, c_4$ | 1, 4 |
| Control gains of evasion pattern, $c_5$ | 1 |
| Control gains of virtual leader-follower, $c_6, c_7$ | 1, 2 |

**Table 1.** Parameters in the proposed algorithm.

| Parameter | Value |
|-----------|-------|
| Velocity time constant, $\tau_v$ | 5 |
| Heading angle time constant, $\tau_\psi$ | 0.75 |
| Altitude time constant, $\tau_h$ | 0.3, 1 |
| Minimum and maximum velocity, $v_{\text{min}}, v_{\text{max}}$ | 5 m/s, 15 m/s |
| Maximum lateral overload, $n_{\text{max}}$ | 5 g |
| Maximum climbing and gliding velocity, $\lambda_{\text{climb}}, \lambda_{\text{glide}}$ | $-5$ m/s, 5 m/s |

**Table 2.** Parameters of the model of the fixed-wing UAV.
4.1. Performance Metrics

To evaluate the performance of the proposed algorithm during flight in an obstacle environment, several metrics \([40, 41]\) are defined as follows:

1. **Order parameter**: it captures the coordination of the motion of the UAV swarm and represents the velocity alignment degree of all UAVs in the swarm.

\[
\Phi = \frac{1}{n(n-1)} \sum_{i,j} v_i \cdot v_j \frac{1}{\|v_i\| \|v_j\|}
\]  

(26)

where \(\Phi\) varies from \([0, 1]\). \(\Phi = 1\) indicates that the UAV swarm is in an ordered state while \(\Phi = 0\) indicates that the UAV swarm is in a chaotic state.

2. **Safety metrics**: it measures the risk of collision between the UAV swarm and obstacles and assesses the ability of the UAV swarm to avoid collisions with the obstacles.

\[
\Phi_s = 1 - \frac{n_{obs}}{n}
\]

(27)

where the number of UAV that enters the obstacle zone is \(n_{obs} = \left| \{ (i,m) \land R_{obs}^i < r_{obs} \} \right| \). \(\Phi_s = 1\) reveals that no UAVs enter the obstacle zone while \(\Phi_s = 0\) reveals that the whole UAV swarm enters the obstacle zone.

3. **Tracking error**: it evaluates the tracking performance of the UAV swarm. The tracking error of position is error between the position of UAV swarm and virtual leader, which can be described as follows:

\[
e_q = \| \langle q \rangle - q_L \| \\
\langle q \rangle = \frac{1}{N} \sum_{i=1}^{N} q_i
\]

(28)

The tracking error of altitude is error between the altitude of UAV swarm and virtual leader, which can be described as follows:

\[
e_h = \| \langle h \rangle - h_L \| \\
\langle h \rangle = \frac{1}{N} \sum_{i=1}^{N} h_i
\]

(29)

4.2. Simulation in Obstacle-Dense Environment

First, the performance of the proposed algorithm is preliminarily tested in an obstacle-dense environment. The destination of the virtual leader is \(\begin{bmatrix} 220 & 220 & 160 \end{bmatrix}^T\) and the velocity of the virtual leader is \(p_L(t) = \begin{bmatrix} 6 & 6 & 4 \end{bmatrix}^T\). The mission requires the fixed-wing UAV to track the trajectory of the virtual leader in an obstacle-dense environment with cylinder obstacles (the radius of each cylinder obstacle is 10 m and the number of obstacles \(k = 12\)). Flight paths in the 3D space and x-y plane are shown in Figure 10. In Figure 10a–d, it is obvious to see that each UAV is randomly distributed, i.e., the UAV swarm is in a disorder state at the beginning of the simulation. The UAV swarm stays in the collective pattern before entering the obstacle zone. Therefore, the UAV swarm reaches the consistent state immediately. In addition, it can be seen that the flight paths of the proposed algorithm have a smaller curvature compared with those of the basic flocking algorithm, which is beneficial to the stable operation of the UAV swarm.

The motion state histories of each UAV and the virtual leader are shown in Figure 11, including the velocity, heading angle and altitude. As shown in Figure 11, the time–response curve of velocity, heading angle and altitude of each UAV will converge when the UAV swarm is close to the destination. When the UAV swarm enters the obstacle zone, the evasion pattern is started. As displayed in the simulation results, the oscillations of the heading angles of UAVs using the proposed algorithm are weaker than those using the basic flocking algorithm when avoiding obstacles, which indicates that the proposed algorithm...
can help to implement a smoother path for obstacle avoidance. Figure 12a,b show history of the order parameter of the proposed algorithm and the basic algorithm, respectively. It can be viewed that the fluctuation of the order parameter curve for the proposed algorithm is less drastic than that for the basic flocking algorithm, which further demonstrates that the proposed algorithm is better at maintaining formation when avoiding obstacles. The position tracking error curve (shown in Figure 13) of using the proposed algorithm is larger than using the basic flocking algorithm due to the fact that a larger distance from obstacles will be kept when using the proposed algorithm.

![Flight paths in obstacle-dense environment.](image)

**Figure 10.** Flight paths in obstacle-dense environment. *(a,c)* are trajectories using the proposed algorithm in 3D space and *x*-y plane. *(b,d)* are trajectories using the basic flocking algorithm in 3D space and *x*-y plane.
Figure 11. The time–response curves of velocity, heading angle, altitude for each UAV and virtual leader in obstacle-dense environment. (a,c,e) are the results using the proposed algorithm; (b,d,f) are the results using the basic flocking algorithm.

Figure 12. The order parameter curves of UAV swarm in obstacle-dense environment. (a) is the result using the proposed algorithm; (b) is the result using the basic flocking algorithm.

Figure 13. Cont.
During the process of obstacle avoidance, the consistency of the UAV swarm using the proposed algorithm is higher, and the curves of velocity and heading angle converge faster in the proposed algorithm and the basic flocking algorithm, respectively. Each dot in Figure 15 represents the positions of the UAVs, and the black dotted lines represent the trajectory of the dynamic threat. It can be seen that the UAVs using the proposed algorithm are less likely to collide with obstacles.

Figure 14a,b are safety parameter curves during obstacle avoidance. Figure 15a,b display the minimum curve distance between obstacles from obstacle one (Obs1) to obstacle twelve (Obs12) and each UAV in the UAV swarm. The UAV swarm only avoids obstacles that have been detected as a collision risk by collision prediction. It can be seen that some UAVs have entered the obstacle zone while using the basic flocking algorithm (as shown in Figure 14b) and the minimum distance of Obs4, Obs9 in Figure 15b is closer to the collision-bound UAVs when compared with Figure 15a, which indicates that UAVs using the proposed algorithm are less likely to collide with obstacles.

Figure 14. The safety parameter curves of the UAV swarm in obstacle-dense environment. (a) is the result using the proposed algorithm; (b) is the result using basic flocking algorithm.

Figure 15. Curves of minimum distances between obstacles and each UAV in obstacle-dense environment. (a) is the result using the proposed algorithm; (b) is the result using the basic flocking algorithm.

4.3. Simulation in Dynamic Threat Environment

In this scenario, a dynamic threat with the velocity of 15 m/s is set. The mission of the UAV swarm and parameters are the same as those in the obstacle-dense environment.
Figure 16 illustrates the formation producing and obstacle avoidance process using the proposed algorithm and the basic flocking algorithm, respectively. Each dot in Figure 16 represents a UAV and the black dotted lines represent the trajectory of the dynamic threat. The colors of the UAV swarm and dynamic threat vary from light to dark with the increase in simulation time. At beginning of the simulation, the UAVs’ positions and heading angles distribute randomly within the given range. In the collective pattern, the formation of the UAV swarm has formed.

![Snapshots of the motion of the UAV swarm and dynamic threat at t = 0 s, t = 12 s, t = 15 s, t = 18 s.](image)

(a) (b) (c) (d)

**Figure 16.** Snapshots of the motion of the UAV swarm and dynamic threat at t = 0 s, t = 12 s, t = 15 s, t = 18 s. (a,c) are the results using the proposed algorithm, while (b,d) are the results using the basic algorithm. (a,b) are in the 3D space while (c,d) are in the horizontal plane.

During the process of obstacle avoidance, the consistency of the UAV swarm using the proposed algorithm is better than using the basic flocking algorithm. In Figure 16d, it is not hard to identify that several UAVs even disperse from the UAV swarm.

In Figure 17, the motion histories of each UAV and the virtual leader, including the velocity, heading angle and altitude, are displayed. It can be seen that after the process of obstacle avoidance, the curves of velocity and heading angle converge faster in the proposed algorithm. However, this advantage is not obvious in the static obstacle environment (Figure 11). Therefore, the proposed algorithm has more advantages in a dynamic threat environment. The oscillation of heading angles of the basic flocking algorithm is still stronger than that of the proposed algorithm, which is further confirmed by the order parameter stated in Figure 18. Tracking error curves shown in Figure 19 illustrate that both the tracking errors of position and altitude of the proposed algorithm are less than that of the basic flocking algorithm. This phenomenon indicates that the proposed algorithm is has a better tracking performance when it is applied to the dynamic threat environment than when it is applied to the static threat environment (Figure 13).
Figure 17. The time–response curves of velocity, heading angle, altitude for each UAV and virtual leader in dynamic threat environment. (a,c,e) are the results using the proposed algorithm; (b,d,f) are the results using the basic flocking algorithm.

Figure 18. The order parameter curves of UAV swarm in dynamic threat environment. (a) is the result using the proposed algorithm; (b) is the result of using the basic flocking algorithm.

Figure 19. Cont.
Figure 19. Tracking errors of position and altitude curves in dynamic threat environment. (a,c) are tracking error curves using the proposed algorithm, while (b,d) are tracking error curves using the basic flocking algorithm.

Figure 20a,b show the safety parameter curves of the UAV swarm in the dynamic threat environment. Curves of minimum distance between the dynamic threat and each UAV in the UAV swarm are shown in Figure 21a,b. It is noticeable that more UAVs in the UAV swarm enter the obstacle zone when using the basic flocking algorithm. Therefore, the security of the UAV swarm is guaranteed when using the proposed algorithm.

Figure 20. The safety parameter curves of the UAV swarm in dynamic threat environment. (a) is the result using the proposed algorithm; (b) is the result using basic flocking algorithm.

Figure 21. Curves of minimum distances between dynamic threat and each UAV in the UAV swarm. (a) is the result using the proposed algorithm; (b) is the result using basic flocking algorithm.

4.4. Simulation in Obstacle Environment with Static and Dynamic Obstacles

In this simulation, the trajectory of the virtual leader is an elliptical shape and the velocity of the virtual leader is \( \mathbf{p}_L(t) = [10 \sin(t/16) \ 5 \cos(t/16) \ -2 \sin(t/5)]^T \). The parameters are the same as those in the dynamic threat environment simulation. The scenario for this experiment is composed of seven static obstacles (Static 1–Static 7) and two dynamic obstacles. Dynamic obstacle one (Dyn1) moves back and forth between point [150, 75, 50] and point [200, 75, 50] along the x-axis, while dynamic obstacle two (Dyn2) moves back and forth between point [295, 0, 70] and point [295, 50, 70] along the y-axis. The basic flocking algorithm and the proposed algorithm are both implemented for com-

\[ \mathbf{p}_L(t) = [10 \sin(t/16) \ 5 \cos(t/16) \ -2 \sin(t/5)]^T \]
parison. The flight paths of each UAV in both the horizontal plane and the 3D plane are displayed in Figure 22, which reflects that the flight paths of UAVs using the proposed algorithm are smoother than those of UAVs using the basic flocking algorithm when the UAV swarm encounters dynamic obstacles. The corresponding state response curves and history of order parameter are shown in Figures 23 and 24, respectively. Tracking errors of position and altitude curves are displayed in Figure 25. From the simulation results, UAVs using both the proposed algorithm and the basic algorithm can track the desired trajectory. However, UAVs using the proposed algorithm are able to accomplish this mission with a more ordered formation and less fluctuation of velocity and altitude. The history of safety parameters and curves of minimum distances between obstacles and each UAV in the UAV swarm are shown in Figures 26 and 27, respectively. By virtue of the local-following pattern, the UAV swarm is able to maintain a safer distance from obstacles when using the proposed algorithm.

Figure 22. Flight paths in obstacle environment with static and dynamic obstacles. (a,c) are trajectories using the proposed algorithm in 3D space and x-y plane. (b,d) are trajectories using the basic flocking algorithm in 3D space and x-y plane.
Figure 23. The time–response curves of velocity, heading angle, and altitude for each UAV and virtual leader in obstacle environment with static and dynamic obstacles. (a,c,e) are the results using the proposed algorithm; (b,d,f) are the results using the basic flocking algorithm.

Figure 24. The order parameter curves of UAV swarm in obstacle environment with static and dynamic obstacles. (a) is the result using the proposed algorithm; (b) is the result using the basic flocking algorithm.

Figure 25. Cont.
Figure 25. Tracking errors of position and altitude curves in obstacle environment with static and dynamic obstacles. (a,c) are tracking error curves using the proposed algorithm, while (b,d) are tracking error curves using the basic flocking algorithm.

Figure 26. The safety parameter curves of the UAV swarm in obstacle environment with static and dynamic obstacles. (a) is the result using the proposed algorithm; (b) is the result using basic flocking algorithm.

Figure 27. Curves of minimum distances between obstacles and each UAV in obstacle environment with static and dynamic obstacles. (a) is the result using the proposed algorithm; (b) is the result using the basic flocking algorithm.

5. Conclusions

This paper investigates the problem of obstacle avoidance of a UAV swarm in an unknown 3D environment by mimicking the behavior patterns of starlings. A flock of starlings exhibits high consistency and flexibility when evading threats. Behavioral mechanisms which emerged in flocks of starlings including collective pattern, evasion pattern and local-following pattern are mapped on the UAV swarm to improve the performance of obstacle avoidance. The collective pattern utilizes neighbor selection and interaction to guarantee the geometric shape of the UAV swarm; the evasion pattern is applied to avoid
collision with obstacles in unknown environments and the local-following pattern selects a
local leader to increase the efficiency of information transmission and enhance safety when the
UAV swarm confronts dynamic threats. The practical dynamic constraints of a fixed-wing UAV, such as velocity limitation, etc., are taken into consideration in this paper. The
proposed algorithm is comprehensively evaluated in three simulation scenarios, including
an obstacle-dense environment, a dynamic threat environment and an unknown environ-
ment with both static and dynamic obstacles. Several metrics are set to the quantify
the performance of both the proposed algorithm and basic flocking algorithm. The simulation
results confirm that the proposed algorithm performs better on formation maintenance,
velocity alignment when avoiding obstacles and the maintenance of safe distances from
obstacles compared with the basic flocking algorithm. In the future, our work will focus on
evaluating the effectiveness of the proposed algorithm on other unmanned systems such
as unmanned ground vehicles (UGVs), autonomous underwater vehicles (AUVs) and so
on. In addition, the proposed algorithm will be applied on a physical platform to solve
the actual obstacle avoidance problem.

Author Contributions: Conceptualization, X.Z. and W.W.; methodology, X.Z. and W.W.; software,
X.Z. and W.W.; validation, X.Z. and W.W.; formal analysis, X.Z. and W.W.; investigation, X.Z. and
W.W.; resources, X.Z. and W.W.; data curation, X.Z. and W.W.; writing—original draft preparation,
W.W.; writing—review and editing, X.Z.; visualization, X.Z. and W.W.; supervision, X.Z. and Y.M.;
project administration, X.Z. and Y.M.; funding acquisition, X.Z. and Y.M. All authors have read and
agreed to the published version of the manuscript.

Funding: This work is supported by National Natural Science Foundation of China (No. 51975011).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Reynolds, A.; Melvor, G. Stochastic modeling of bird flocks: Accounting for the cohesiveness of collective motion. J. R. Soc. Interface 2022, 19, 20210745. [CrossRef]
2. Papadopoulou, M.; Hildenbrandt, H. Emergence of splits and collective turns in pigeon flocks under predation. R. Soc. Open Sci. 2022, 9, 211898. [CrossRef] [PubMed]
3. Susumu, I.; Nariya, U. Emergence of a giant rotating cluster of fish in three dimensions by local interactions. J. Phys. Soc. Jpn. 2022, 91, 064806.
4. Costanzo, A.; Hildenbrandt, H. Causes of variation of darkness in flocks of starlings, a computational model. Swarm Intell. 2022, 16, 91–105. [CrossRef]
5. Hemelrijk, C.; Zuidam, L. What underlies waves of agitation in starling flocks. Behav. Ecol. Sociobiol. 2015, 69, 755–764. [CrossRef] [PubMed]
6. Reynolds, C. Flocks, herds and schools: A distributed behavioral model. Comput. Graph. 1987, 21, 25–34. [CrossRef]
7. Vicsek, T.; Czirok, A. Novel type of phase transition in a system of self-driven particles. Phys. Rev. Lett. 1995, 75, 1226–1229. [CrossRef] [PubMed]
8. Couzin, I.; Krause, J. Collective memory and spatial sorting in animal groups. J. Theor. Biol. 2002, 218, 1–11. [CrossRef]
9. Olfati-Saber, R. Flocking for multi-agent dynamic systems: Algorithms and theory. IEEE Trans. Autom. Control. 2006, 51, 401–420. [CrossRef]
10. Tanner, H.G.; Jadbabaie, F. Flocking in fixed and switch networks. IEEE Trans. Autom. Control. 2007, 52, 863–868. [CrossRef]
11. Duan, H.; Qiao, P. Pigeon-inspired optimization a new swarm intelligence optimizer for air robot path planning. Int. J. Intell. Comput. Cybern. 2014, 7, 24–37. [CrossRef]
12. Zhang, X.; Xia, S. Quantum behavior-based enhanced fruit fly optimization algorithm with application to UAV path planning. Int. J. Comput. Intell. Syst. 2020, 13, 1315–1331. [CrossRef]
13. Zhou, Z.; Duan, H. Unmanned aerial vehicle close formation control based on the behavior mechanism in wild geese. Sci. Sin. Technol. 2017, 47, 230–238. [CrossRef]
14. Xie, R.; Gu, C. A starling swarm coordination algorithm. Wuhan Univ. (Nat. Sci. Ed.) 2019, 65, 229–237.
15. Grammatikis, P.; Sarigiannidis, P. A compilation of UAV applications for precision agriculture. Comput. Netw. 2020, 172, 107148. [CrossRef]
16. Liu, W.; Zhang, T. A hybrid optimization framework for UAV reconnaissance mission planning. *Comput. Ind. Eng.* 2022, 173, 108653. [CrossRef]

17. Pei, J.; Chen, H. UAV-assisted connectivity enhancement algorithms for multiple isolated sensor networks in agricultural Internet of things. *Comput. Netw.* 2022, 207, 108854. [CrossRef]

18. Shi, Y.; Liu, Y. Multi-UAV cooperative reconnaissance mission planning novel method under multi-radar detection. *Sci. Prog.* 2022, 105, 1–20. [CrossRef] [PubMed]

19. He, Z.; Liu, C. Dynamic anti-collision A-star algorithm for multi-ship encounter situations. *Appl. Ocean Res.* 2022, 118, 102995. [CrossRef]

20. Wu, Y.; Gou, J. A new consensus theory-based method for formation and obstacle avoidance of UAVs. *Aerosp. Sci. Technol.* 2020, 107, 106332. [CrossRef]

21. Tian, S.; Li, Y. Multi-robot path planning in wireless sensor networks based on jump mechanism PSO and safety gap obstacle avoidance. *Future Gener. Comput. Syst.* 2021, 118, 37–47. [CrossRef]

22. Vargas, S.; Becerra, H. MPC-based distributed formation control of multiple quadcopters with obstacle avoidance and connectivity maintenance. *Control. Eng. Pract.* 2022, 121, 105054. [CrossRef]

23. Zhang, X.; Duan, H. Altitude consensus-based 3D flocking control for fixed-wing unmanned aerial vehicle swarm trajectory tracking. *Proc. Inst. Mech.Eng. Part G J. Aerosp. Eng.* 2016, 230, 2628–2638. [CrossRef]

24. Qi, J.; Guo, J. Formation tracking and obstacle avoidance for multi quadrotors with static and dynamic obstacles. *IEEE Robot. Autom. Lett.* 2022, 7, 1713–1722. [CrossRef]

25. Wu, J.; Wang, H. Learning-based fixed-wing UAV reactive maneuver control for obstacle avoidance. *Aerosp. Sci. Technol.* 2022, 316, 107623. [CrossRef]

26. Liu, X.; Yan, C. Towards flocking navigation and obstacle avoidance for multi-UAV systems through hierarchical weighting Vicsek model. *Aerospace* 2021, 8, 286. [CrossRef]

27. Fu, X.; Pan, J. A formation maintenance and reconstruction method of UAV swarm based on distributed control. *Aerosp. Sci. Technol.* 2020, 104, 105981. [CrossRef]

28. Lim, J.; Pyo, S.; Kim, N.; Lee, J.; Lee, J. Obstacle magnification for 2-D collision and occlusion avoidance autonomous multirotor aerial vehicles. *IEEE/ASME Trans. Mechatron.* 2020, 25, 2428–2436. [CrossRef]

29. Hemelrijk, C.; Hildenbrandt, H. Diffusion and topological neighbors in flocks of starlings: Relating a model to empirical data. *PLoS ONE* 2015, 10, e0126913. [CrossRef] [PubMed]

30. Ballerini, M.; Cabibbo, N. Empirical investigation of starling flocks: A benchmark study in collective animal behavior. *Anim. Behav.* 2008, 76, 201–215. [CrossRef]

31. Carere, C.; Montanino, S. Aerial flocking patterns of wintering starlings, Sturnus vulgaris, under different predation risk. *Anim. Behav.* 2009, 77, 101–107. [CrossRef]

32. Procaccini, A.; Orlandi, A. Propagating waves in starling, Sturnus vulgaris, flocks under predation. *Anim. Behav.* 2011, 82, 759–765. [CrossRef]

33. Hogan, B.; Hidenbrandt, H. The confusion effect when attacking simulated three-dimensional starling flocks. *R. Soc. Open Sci.* 2017, 4, 160564. [CrossRef] [PubMed]

34. Brown, J.; Bossoainer, T. Information transfer in finite flocks with topological interactions. *J. Comput. Sci.* 2021, 53, 101370. [CrossRef]

35. Zamani, H.; Nadimi-Sharaki, M. Starling murmuration optimizer: A novel bio-inspired algorithm for global and engineering. *Computer Methods in Applied Mechanics and Engineering.* 2022, 392, 114616. [CrossRef]

36. Storms, R.; Carere, C. Complex patterns of collective escape in starling flocks under predation. *Behav. Ecol. Sociobiol.* 2019, 73, 10. [CrossRef]

37. Olcay, E.; Schuhmann, F. Collective navigation of a multi-robot system in an unknown environment. *Robot. Auton. Syst.* 2020, 132, 103604. [CrossRef]

38. Yu, Y.; Duan, H. Turning control multiple UAVs imitating the super-maneuver behavior in massive starling. *Robot. Auton. Syst.* 2020, 132, 103604. [CrossRef]

39. Lei, X.; Liu, M. Fission control algorithm for swarm based on local following interaction. *Control. Decis.* 2013, 28, 741–746.

40. Nagy, M.; Akos, Z.; Biro, D.; Vicsek, T. Hierarchical group dynamic in pigeon flocks. *Nature* 2010, 464, 890–893. [CrossRef]

41. Vásárhelyi, G.; Virág, C. Optimized flocking of autonomous drone in confined environments. *Sci. Robot.* 2018, 3, 3536–3550. [CrossRef] [PubMed]