



Xingyu Liu, Xiaojia Xiang, Yuan Chang 🔎, Chao Yan, Han Zhou * and Dengqing Tang

College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China; liuxingyu15@nudt.edu.cn (X.L.); xiangxiaojia@nudt.edu.cn (X.X.); changyuan10@nudt.edu.cn (Y.C.); yanchao17@nudt.edu.cn (C.Y.); tangdengqing09@nudt.edu.cn (D.T.)

* Correspondence: zhouhan@nudt.edu.cn

Abstract: Flocking navigation, involving alignment-guaranteed path following and collision avoidance against obstacles, remains to be a challenging task for drones. In this paper, we investigate how to implement flocking navigation when only one drone in the swarm masters the predetermined path, instead of all drones mastering their routes. Specifically, this paper proposes a hierarchical weighting Vicsek model (WVEM), which consists of a hierarchical weighting mechanism and a layer regulation mechanism. Based on the hierarchical mechanism, all drones are divided into three layers and the drones at different layers are assigned with different weights to guarantee the convergence speed of alignment. The layer regulation mechanism is developed to realize a more flexible obstacle avoidance. We analyze the influence of the WVEM parameters such as weighting value and interaction radius, and demonstrate the flocking navigation performance through a series of simulation experiments.

Keywords: drone swarm; flocking navigation; Vicsek model; hierarchical weighting mechanism



Citation: Liu, X.; Xiang, X.; Chang, Y.; Yan, C.; Zhou, H.; Tang, D. Hierarchical Weighting Vicsek Model for Flocking Navigation of Drones. *Drones* **2021**, *5*, 74. https://doi.org/ 10.3390/drones5030074

Academic Editor: Oleg Yakimenko

Received: 13 June 2021 Accepted: 29 July 2021 Published: 3 August 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/).

1. Introduction

In recent years, researchers are showing growing interests in drone swarms, among which flocking navigation is particularly eye-catching. Flocking behaviors refer to the emergence of group-level phenomena through simple individual interactions and probably arise from local rules without the central coordination [1–3]; navigation refers to the swarm tracking a predetermined path [4].

There are a great deal of studies that can achieve flocking navigation successfully by assuming that all drones have access to the predetermined path information [5–7]. If all drones need to master the path information, the data transmitting load is heavy, or it is simply difficult to achieve. On the one hand, the drones receive the predetermined path from the ground station in remote control mode. Then, all drones should be equipped with communicators, which in turn, causes a heavy data transmitting load. On the other hand, the drones are designed to plan their path based on the perceived goal position in autonomous control mode, but the capability of path planning varies a lot; thus, maybe only few of them can accomplish the planning task by obtaining the goal information and selecting the path accordingly and correctly. Furthermore, the noncooperation goal position may not be accurately expressed and the path information may not be transmitted for the interruption of communication. Therefore, it is a meaningful task to navigate the flocking with only one drone mastering the path information. At the same time, it is also a challenging one due to the requirements of rapid convergence and obstacle avoidance under the limited interaction radius.

To achieve flocking navigation, we need to ensure that all drones align to the path of the information drone who means the drone that knows the predetermined path. As a classic flocking model, the Vicsek model (VEM) describes the system where individuals align their movement directions to those individuals within a local neighborhood, and all individuals in the flock eventually align [8]. Taking the advantage of the alignment



properties of VEM, as long as all the drones successfully make themselves align with the information drone within a short enough time, they can follow with the information drone closely; thus, the swarm can move along the desired path.

To achieve alignment as soon as possible, the topology relationship is established according to the information held by the drone and the interaction radius. Related studies have suggested that if the hierarchy exists and if several drones are not easily influenced by other drones, they can induce of the movement of the Vicsek-model swarm [9–11]. A hierarchical structure—dividing all drones into α drones, β drones, and γ drones—is established in this paper to guarantee enough influence of the crucial information, as shown in Figure 1. Besides, the information is no longer limited to the path information and can also include information about the perceived obstacles. A layer-regulation mechanism is also designed to ensure rapid alignment with the drones that detect the obstacle, which leads to the more flexible obstacle avoidance.

Accordingly, this paper proposes the hierarchical weighting Vicsek model (WVEM), including the hierarchical weighting mechanism and the layer regulation mechanism. The hierarchical weighting mechanism is designed to improve the convergence speed of alignment with the information drone, and the layer regulation mechanism is to improve the performance of obstacle avoidance.



Figure 1. The scenario of flocking navigation. The drones' movements are chaotic and disordered at the beginning. Only the α drone masters the path information and it adjusts its heading angle accordingly. The drone whose perception range covers the α drone (β drone with blue color) can perceive the information of the α drone, while the drone whose range does not cover the α drone (γ drone with purple color) cannot perceive it directly.

The rest of this paper is organized as follows. In Section 2, we introduce some background work including VEM and several matrixes. Section 3 proposes the WVEM and the WVEM-based flocking algorithm. Numerous simulations are carried out in Section 4 to verify and analyze the model performance. The last section summarizes all the points of this paper.

2. Related Works

This section firstly introduces the classic Vicsek Model and a detailed description of the involved parameters. Then, we introduce several coefficient matrixes to describe the relationship among drones in the flocking.

3 of 16

2.1. Vicsek Model

VEM describes *N* drones moving on the $L \times L$ plane with periodic boundaries at a constant speed v_0 in discrete time, and the drone density is denoted as $\rho = N/L^2$. At time step *k*, the position of the drone *i* is noted as $\vec{x}_i(k) \in \mathbb{R}^2$ and the velocity is $\vec{v}_i(k) \in \mathbb{R}^2$, $i \in 1, ..., N$. At the initial moment, both the horizontal coordinate and vertical coordinate of each drone obey [0, L] uniform distribution, the direction obeys $[0, 2\pi]$ uniform distribution. The heading angle of each drone at time step k + 1 is the average of the directions of its neighbors at time step *k*, which is updated by

$$\theta_i(k+1) = \tan^{-1} \left(\frac{\sum_{j \in \mathcal{N}_i(k)} \sin[\theta_j(k)]}{\sum_{j \in \mathcal{N}_i(k)} \cos[\theta_j(k)]} \right) + \Delta \theta_i(k), \tag{1}$$

where $\mathcal{N}_i(k)$ represents the index set of the neighbors of drone i, $\Delta \theta_i(k)$ denotes the noise and is evenly distributed in $[-\eta/2, \eta/2]$, and η is the noise intensity. Drone j is the neighbor of drone i if it requires $0 < \|\vec{x}_j(k) - \vec{x}_i(k)\| \le r$, and r is the interaction radius.

The velocity of the drone *i* is updated by

$$\vec{v}_i(k+1) = v_0 \cos[\theta_i(k+1)]\vec{e}_1 + v_0 \sin[\theta_i(k+1)]\vec{e}_2,$$
(2)

where \vec{e}_1 and \vec{e}_2 are mutually perpendicular unit vectors, \vec{e}_1 is parallel to the x-axis, and \vec{e}_2 is parallel to the y-axis. The position of drone *i* is updated by

$$\vec{x}_i(k+1) = \vec{x}_i(k) + \vec{v}_i(k+1).$$
 (3)

2.2. Coefficient Matrixes

In this paper, some matrixes will be adjusted to describe our proposed model and characterize the relationship among drones clearly, including the contribution matrix, dominance matrix, and adjacency matrix [12].

The dominance matrix $B_N = [b_{ij}]_{N \times N}$ is defined to describe the information flow direction between each pair of drones. The information flow specifies the set of drones whose behavior influences the decision of the given drone. For drone *i* and drone *j*, only if the information of *j* flows to *i* can its behavior influence drone *i* and $b_{ij} = 1$; otherwise, $b_{ij} = 0$. For VEM, each pair of drones have the same status and their behaviors can be influenced by each other—that is, $b_{ij} = 1, \forall i, j = 1, \dots, N$.

The contribution matrix $C_N = [c_{ij}]_{N \times N}$ is defined to describe the contribution intensity of every drone during the decision-making process regarding the new preferred directions of the drones. The element value in the matrix corresponds to the weight value assigned to every drone, respectively, and c_{ij} represents the contribution of drone *j* to drone *i*. For VEM, every drone has the same contribution intensity, and it is reasonable to assume that $c_{ij} = 1, \forall i, j = 1, ..., N$.

The adjacency matrix $A_N = [a_{ij}]_{N \times N}$ is defined to describe the drones within the perception range of the given drone. Only the drones within the perception range of the given drone can influence the given drone. The element of A_N is expressed as

$$a_{ij}(k) = \begin{cases} 1, i = 1, \dots, N, j \in \mathcal{N}_i(k) \\ 0, \text{ otherwise} \end{cases},$$
(4)

where $N_i(k) = \{j \mid 0 < ||\vec{x}_i(k) - \vec{x}_j(k)|| \le r\}$ and *r* denotes the interaction radius. Therefore, the movement direction of the drone in VEM is given by

$$\theta_i(k+1) = \tan^{-1} \left(\frac{\sum_{j=1}^N l_{ij}(k) \sin\left[\theta_j(k)\right]}{\sum_{j=1}^N l_{ij}(k) \cos\left[\theta_j(k)\right]} \right) + \Delta \theta_i(k),$$
(5)

where $l_{ij}(k) = c_{ij} \cdot b_{ij} \cdot a_{ij}(k)$, $\forall i, j = 1, ..., N$. These matrixes mentioned above will be applied later when describing our model.

3. Hierarchical Weighting Vicsek Model

This section introduces our hierarchical weighting Vicsek model. Firstly, the properties of the α drone are presented. Then, all drones are divided into three layers based on the interaction topology and assigned different weights. Next, the layer regulation mechanism is developed for the flock to adapt to a variety of scenarios. Finally, a WVEM-based flocking algorithm is proposed to achieve flocking navigation for the drone swarm.

3.1. Hierarchical Weighting Mechanism

As long as each drone follows the information drone within a permitted time, flocking navigation can be achieved. The goal of the hierarchical weighting mechanism is to align drones with the information drone as soon as possible.

A so-called α drone is proposed to represent the drone that knows the path information or detects the obstacle. The heading angle $\theta_i(k)$ of the α drone is only determined by the scenario-specific actual demand and is not affected by other drones, which is the reference state of all the other drones. To describe the ability of the α drone to access the reference state, this paper proposes the enable matrix $E_N = [e_i]_{N \times 1}$, in which the elements follow the rule,

$$e_i(k) = \begin{cases} 0, \text{ if } i \notin \Lambda_{\alpha} \\ 1, \text{ if } i \in \Lambda_{\alpha} \end{cases}$$
(6)

where $i \in \Lambda_{\alpha}$ means that drone *i* is the α drone. The approach to define the α drone is introduced in the next section.

Based on the interaction topology relationship with the α drone, other drones are divided into two layers: β drones and γ drones. The drone that can directly interact with the α drone is called a β drone, namely, the drone whose perception range covers the α drone. Correspondingly, the drones that do not directly interact with the α drone are called γ drones. The definition of drones is shown in Figure 2. The definition process of drones is dynamic during the flocking motion. As the system evolves, some drones enter or leave the interaction range of the α drone. The β drone changes to a γ drone once failing to perceive the α drone. Instead, the γ drone becomes a β drone. The drones in the entire system are divided into three layers: α drone, β drone, and γ drone, which follows the rule,

$$i \in \begin{cases} \Lambda_{\beta}, \text{ if } i \notin \Lambda_{\alpha}, \exists j \in \Lambda_{\alpha}, 0 < \|\vec{x}_{i}(k) - \vec{x}_{j}(k)\| \le r\\ \Lambda_{\gamma}, \text{ if } i \notin \Lambda_{\alpha}, \forall j \in \Lambda_{\alpha}, \|\vec{x}_{i}(k) - \vec{x}_{j}(k)\| > r \end{cases}$$
(7)

The definition matrix $D_N = [d_i]_{N \times 1}$ is proposed to denote the layer of every drone, and the element d_i follows the rule,

$$d_i(k) = \begin{cases} 1, & \text{if } i \in \Lambda_{\alpha} \\ 2, & \text{if } i \in \Lambda_{\beta} \\ 3, & \text{if } i \in \Lambda_{\gamma} \end{cases}$$
(8)

where $i \in \Lambda_{\beta}$ means that drone *i* is a β drone and $i \in \Lambda_{\gamma}$ means that drone *i* is a γ drone.

In the model, the movement direction of the α drone will not be affected by any other drone. For drones of other layers, all members excluding themselves have the ability to influence their movement directions. The flows of direction information among drones are shown in the Figure 2a. Only when the information of a certain drone flows to a given drone, can it contribute to the given drone. WVEM has its dominance matrix B_N , in which the element b_{ij} follows the rule,

$$b_{ij}(k) = \begin{cases} 0, & \text{if } i \in \Lambda_{\alpha} \\ 1, & \text{if } i \in \Lambda_{\beta} \text{ or } \Lambda_{\gamma} \end{cases}$$
(9)

Besides, $b_{ii} = 0$, thereby B_N is also called a strictly dominance matrix.

Furthermore, we hope to increase the system convergence speed by weighting drones. The same weight is assigned to the drones of the same layer, and the drones of different layers are assigned with different weights. The α drone is assigned with weight k_{α} , β drones are assigned with k_{β} , and γ drones are assigned with k_{γ} . WVEM has its contribution matrix C_N , in which the element c_{ii} follows the rule,

$$c_{ij}(k) = \begin{cases} k_{\alpha}, & \text{if } j \in \Lambda_{\alpha} \\ k_{\beta}, & \text{if } j \in \Lambda_{\beta} \\ k_{\gamma}, & \text{if } j \in \Lambda_{\gamma} \end{cases}$$
(10)

The value of weighting plays a significant role in system performance. In the system evolution, the heading angle of the α drone is exactly the reference state. A β drone is directly influenced by the α drone, so that it aligns with the α drone more quickly and its direction is closer to the reference state than the γ drone. The closer the drone movement direction is to the reference state, the higher the assigned weight value to encourage all drones to align with the reference state faster. Therefore, there should be $k_{\alpha} > k_{\beta} > k_{\gamma} > 0$. Its rationality is verified, and more beneficial weighting guidelines have been researched by experiments since. To simplify the model, without loss of generality, let $k_{\gamma} = 1$ in this paper, which means that the contribution intensity of the γ drone is regarded as a benchmark for other drones.

The drones in WVEM have the same kinematic model as VEM. The updated model of drones heading angles is as follows and is shown in Figure 2b:

$$\theta_i(k+1) = \tan^{-1} \left(\frac{\sum_{j=1}^N l_{ij}(k) \sin\left[\theta_j(k)\right]}{\sum_{j=1}^N l_{ij}(k) \cos\left[\theta_j(k)\right]} \right) + e_i(k) \cdot \theta_i^d(k) + \Delta \theta_i(k)$$
(11)

where

$$l_{ij}(k) = c_{ij}(k) \cdot b_{ij}(k) \cdot a_{ij}(k), \forall i, j = 1, \dots, N,$$
(12)

and $\theta_i^d(k)$ refers to the angle obtained by the α drone after adjusting its angle according to the path information that it masters.



Figure 2. Various relationships between drones. (**a**) Definition of different layers of drones and the interaction topology diagram between drones. (**b**) Vector synthesis of drone velocity. In (**a**), the direction of the red dashed arrow represents the information flow direction. In (**b**), the length of the dotted line represents the contribution intensity. The drone does not contribute to its new preferred movement direction.

3.2. Layer Regulation Mechanism

As long as the α drone is identified, the layers of other drones can also be determined, which denotes that the identification of the α drone is the most important. Whoever masters

the information conducive to the flock's survival becomes the α drone. The setup of the α drone is related to the specific scenario encountered by the flocking. This paper mainly focuses on two actual scenarios:

- Scenario (I), Flocking Navigation. This refers to the process where the flocking movement is prompted by the drone that masters the predetermined trajectory.
- Scenario (II), Obstacle Avoidance. This refers to the process where the flocking movement is prompted by the drones that detect the obstacle.

The scenario matrix $S_N = [s_i]_{N \times 1}$ represents the scenario information mastered by each drone, and the matrix element s_i follows the rule,

$$s_{i}(k) = \begin{cases} 0, \text{ if } i \notin \Gamma_{\text{navigation}} \& i \notin \Gamma_{\text{avoidance}} \\ 1, \text{ if } i \in \Gamma_{\text{navigation}} \& i \notin \Gamma_{\text{avoidance}} \\ 2, \text{ if } i \notin \Gamma_{\text{navigation}} \& i \in \Gamma_{\text{avoidance}} \\ 3, \text{ if } i \in \Gamma_{\text{navigation}} \& i \in \Gamma_{\text{avoidance}} \end{cases}$$
(13)

where $i \in \Gamma_{\text{navigation}}$ means that drone *i* knows the trajectory information, and $i \notin \Gamma_{\text{navigation}}$ means that *i* does not know. Thereby, the definition of the α drone is as follows:

$$i \in \begin{cases} \alpha, & \text{if } s_i(k) \neq 0\\ \text{non} - \alpha, & \text{if } s_i(k) = 0 \end{cases}.$$
(14)

The behavior of the α drone is to adjust its heading angle $\theta_i(k)$, which is exactly the reference state of the flocking. For specific scenarios, the α drone has different behavior strategies:

- In scenario (I), the α drone acts to move along the trajectory $\theta_i(k) = \theta_{i,s=1}^d$.
- In scenario (II), the α drone acts to stay away from the obstacle $\theta_i(k) = \theta_{i,s}^d = \theta_{i,s}^d$.

Different scenarios possess different degrees of importance. When the α drone is in comprehensive scenarios, it takes the action that should be taken assuming the most important scenario. For the flocking survival, obstacle avoidance is more important, and the α drone takes action to stay away from the obstacle if finding the objective and obstacle at the same time,

$$\theta_{i,s_i=3}^d = \theta_{i,s_i=2}^d. \tag{15}$$

During the flocking movement, the abrupt scenario appears randomly, while an immutable layer of the drone cannot address it appropriately. Hence, the layer regulation mechanism for the drone is proposed in response to the abruptness and randomness of the scenario. The layer regulation mechanism only refers to the mutual transformation of non- α drones and the α drone, and the layers of other drones are transformed accordingly. For a navigating flock who suddenly detects obstacle, the regulation mechanism includes two rules:

- If a non- α drone detects the objective or obstacle, it immediately transforms into an α drone and takes corresponding actions required by the specific scenario. For the α drone, only if it detects the obstacle during the process of moving to the objective does it change its action to stay away from the obstacle; otherwise, it maintains the previous behavior.
- For the α drone that previously detected the obstacle and is no longer detecting the obstacle, if it does not master the objective location, it is transformed into a non- α drone and its identity is determined with (7) and (8), otherwise it is also an α drone and only changes behavior.

3.3. WVEM-Based Flocking Algorithm

VEM is the solution to the rule of velocity matching and provides the alignment term. To achieve complete flocking navigation, it is still necessary to add attraction term and repulsion term for the drone swarm [13,14]. The attraction force is used to maintain the swarm compactness and the repulsion force is to avoid the collision. The form of the attraction term is as follows:

$$\vec{v}_{i}^{att}(k+1) = \sum_{j \in \mathcal{N}_{i}^{1}(k)} \left(\frac{r_{rep} - \|\vec{x}_{ij}(k)\|}{r_{att} - r_{rep}} \cdot \frac{\vec{x}_{ij}(k)}{\|\vec{x}_{ij}(k)\|} \right),$$
(16)

where r_{att} represents the range of the attraction force, r_{rep} represents the range of the repulsion force, and $j \in \mathcal{N}_i^1(k)$ satisfies $r_{rep} < \|\vec{x}_{ij}\| \le r_{att}$. The form of the repulsion term is as follows:

$$\vec{v}_{i}^{rep}(k+1) = \sum_{j \in \mathcal{N}_{i}^{2}(k)} \left(\frac{r_{rep} - \|\vec{x}_{ij}(k)\|}{r_{rep}} \cdot \frac{\vec{x}_{ij}(k)}{\|\vec{x}_{ij}(k)\|} \right),$$
(17)

where $j \in N_i^2(k)$ satisfies $0 < ||\vec{x}_{ij}|| \le r_{rep}$. The intensities of both terms of the drones of all layers to other drones are the same and the α drone is still not influenced by any other drone. The velocity of the drone is updated according to the following equations:

$$\vec{v}_i^{\text{des}}(k+1) = c^{align} \cdot \vec{v}_i^{align}(k+1) + c^{rep} \cdot \vec{v}_i^{rep}(k+1) + c^{att} \cdot \vec{v}_i^{att}(k+1),$$
(18)

$$\vec{v}_i(k+1) = \frac{\vec{v}_i^{\text{des}}(k+1)}{\|\vec{v}_i^{\text{des}}(k+1)\|} \cdot \min\Big\{ \|\vec{v}_i^{\text{des}}(k+1)\|, v^{\max}\Big\},\tag{19}$$

where c^{align} , c^{rep} , c^{att} represent the coefficients of the alignment term, the repulsion term, and the attraction term, respectively, and v^{max} represents the max velocity magnitude of the drone, $v^{max} = 0.1$.

Accordingly, the process of the WVEM-based flocking algorithm is summarized as Algorithm 1.

Algorithm 1 WVEM-based flocking algorithm

```
Require:
     Initialize every drone's position \vec{x}_i^0 and velocity \vec{v}_i^0 randomly;
     Initialize maximum time step k_{max};
 1: for the time step k := 1, \ldots, k_{\max} \mathbf{do}
 2:
        for the drone number i := 1, \ldots, N do
 3:
            Update s_i with (13);
 4:
            if s_i \neq 0 then
 5:
                Layer regulate: d_i(k) = 1;
 6:
                Behavior: e_i(k) = 1, \theta_i(k) = \theta_{i,s_i}^d;
 7:
            else
 8:
                Obtain the neighbor index \mathcal{N}_i(k) = \{j_1, j_2, \ldots\};
 9:
                Obtain each neighborhood identity d_i(k);
10:
                Determine the identity d_i with (8) and (9);
                Obtain each neighborhood position \vec{x}_i(k);
11:
12:
                Obtain each neighborhood velocity \vec{v}_i(k);
13:
             end if
             Update a_{ij}(k), b_{ij}(k), c_{ij}(k) with (4), (9), (10), respectively;
14:
            Update l_{ij}(k) with (12);
15:
16:
            Update the heading angle \theta_i(k+1) with (11);
            Update the alignment term \vec{v}_i^{align}(k+1) with (2);
Update the attraction term \vec{v}_i^{att}(k+1) with (16);
17:
18:
            Update the repulsion term \vec{v}_i^{rep}(k+1) with (17);
Update the velocity \vec{v}_i(k+1) with (18) and (19);
19:
20:
21.
            Update the position \vec{x}_i(k+1) with (3);
22:
         end for
23: end for
```

4. Numerical Simulation and Analysis

This section firstly verifies whether all drones converge to the reference state with the α drone in a fixed movement direction. Next, the factors affecting the model alignment performance are analyzed in detail. Then, the flocking navigation and better avoidance of abrupt are realized in the presence of noise, which verifies the robust performance and practical value of the model.

4.1. The Verification of Alignment with the α Drone

This section mainly observes whether all drones can converge to the fixed direction, which is the foundation of achieving flocking navigation. The attraction term and repulsion term are ignored and the experiments are conducted in the periodic boundary environment. We use convergence time k_{con} as a performance metric to evaluate system performance. k_{con} denotes the minimum time steps required for the system to reach the desired consistency degree Deg_{θ}^d . Deg_{θ} represents the consistency degree between all drones' heading angles and the reference state,

$$\vec{\theta}(k) = \left[\theta_1(k) - \theta^d, \theta_2(k) - \theta^d, \cdots, \theta_N(k) - \theta^d\right],$$
(20)

$$\theta_{con}(k) = \left\| \vec{\theta}(k) \right\|,\tag{21}$$

$$\operatorname{Deg}_{\theta}(k) = \left[\pi - \frac{\theta_{con}(k)}{N - N_{\alpha}}\right] / \pi,$$
(22)

where θ^d represents the reference state and N_{α} represents the quantity of the α drone. Accordingly, k_{con} is expressed as follows;

$$k_{\text{con}} = \min\left\{k_0 \mid \forall k \ge k_0, \operatorname{Deg}_{\theta}(k) \ge \operatorname{Deg}_{\theta}^d\right\},\tag{23}$$

where $0 < \text{Deg}_{\theta}^{d} < 1$; the higher Deg_{θ}^{d} , the more severe consistency degree required by the system. The shorter k_{con} , the faster convergence speed and the better performance.

In these experiments, the reference state is set to $\pi/4$. There is one α drone whose heading angle is always $\pi/4$, with an initial position that is in the lower-left corner of the area. To stress the feasibility of the alignment to the reference state as the focus of our study, this section dose not consider any noise temporarily. The parameter settings are as follows: $v_0 = 0.03$, $\rho = 4.102$, r = 1, $\text{Deg}_{\theta}^d = 0.9984$, $\eta = 0$, $k_{\alpha} = 10$, $k_{\beta} = 5$, N = 201, L = 7. The process of system evolution is shown in Figure 3.



Figure 3. System circumstance at different time steps under different groups. The red drone represents the α drone, blue represents β drones, and purple represents γ drones. The β drones are almost within a certain range around the α drone.

Figure 3 shows the drones transforming from the initial chaos and disorder to the final alignment with the α drone, vividly illustrating the feasibility. As shown in Figure 4, the movement directions of all drones converge to the reference state and the consistency degree gradually tends to 1, which also demonstrates the reliability of WVEM.



Figure 4. The difference between the movement direction of each drone and the reference state at each time step. The curves of different color display the trend of different drone. The inset shows the consistency degree of the swarm at each time step.

Besides, to verify the significance of WVEM, we compare it with two baseline models, i.e., the classic Vicsek model (VEM) and the classic Vicsek model with one α drone (VEM- α). The initial position and velocity of drone also influence the convergence performance, thereby the 100 simulations with different initial states are carried out for each model to avoid the randomness and obtain the general results. Three sets of groups are discussed: (a)N = 101, L = 4.962; (b)N = 201, L = 7; (c)N = 501, L = 11.051. The results are shown in Table 1.

As shown in Table 1, VEM- α can induce the swarm to converge to the reference state $\pi/4$ while the swarm inspired by VEM cannot. The only difference between VEM- α and VEM is α drone, which suggests that the α drone is the foundation for the swarm to converge to the reference state. The direction of the α drone is not affected by other drones. Other drones are constantly directly or indirectly affected by the α drone, leading to the tendency that they align with the α drone and eventually align to the reference state.

Compared with VEM- α , the convergence time of WVEM with $k_{\alpha} = 10$, $k_{\beta} = 5$ is significantly shorter. VEM- α is a special case of WVEM, specifically referring to the case of $k_{\alpha} = k_{\beta} = 1$, which means that in the swarm, all drones posses the same contribution intensities. The superior system performance of WVEM demonstrates that the weighting mechanism can improve the speed of the alignment of all drones to the reference state.

Test Case Setup		Mathad	Convergence Time				Converge to
Number of Drones	Size	Method	Avg	25th pctl	50th pctl	75th pctl	Reference State
100	4.96 × 4.96	VEM	84.13	55.00	81.00	105.50	No
		VEM-α	568.39	362.50	542.50	741.00	Yes
		WVEM	113.70	83.50	97.00	122.00	Yes
200	7 × 7	VEM	153.74	104.00	135.50	193.50	No
		VEM-α	1236.60	701.50	969.00	1489.50	Yes
		WVEM	238.57	152.50	185.50	301.50	Yes
500	11.2 × 11.2	VEM	296.30	204.50	282.50	357.00	No
		VEM-α	4097.30	1550.50	3157.00	4333.00	Yes
		WVEM	662.36	370.50	656.00	757.00	Yes

Table 1. The	alignment	performance	of eac	h model
--------------	-----------	-------------	--------	---------

4.2. Analysis of Factors Affecting Alignment Performance

This section analyses the factors affecting the alignment performance. There are many factors that affect the system convergence time due to numerous parameters involved in

WVEM—these include, but not limited to, weighting values, quantity and location of the α drone, drone interaction radius, drone density, noise intensity and type, consistency degree requirement, limited field of view, drone velocity, and other factors. Among these factors, weighting is regarded as the highlight of this paper. We firstly explore the influence of weighting values on the model performance. Then, we discuss how the drone interaction radius influences the performance. The attraction term and repulsion term are also ignored and the experiments are conducted in the periodic boundary environment. There is one α drone whose heading angle is always $\pi/4$ and initial position is the lower-left corner of the area in the system.

4.2.1. Impacts of Weighting Values

To initially analyze the impact of weighting values on the performance with k_{α} changing, let $k_{\beta} = 1$. The other parameter settings are as follows: $v_0 = 0.03$, $\rho = 4.102$, r = 1, $\text{Deg}_{\theta}^d = 0.9984$, $\eta = 0$, N = 201, L = 7. Experiments of different initial states are conducted 100 times for different k_{α} to obtain the general conclusions. The results are shown in Figure 5.

As shown in Figure 5, the general trend is that the convergence time decreases with k_{α} increasing. The α drone holds larger contribution intensity to induce other drones to align with it more quickly, which allows better system performance. Before $k_{\alpha} = 16$, the system convergence time decreases strictly. From $k_{\alpha} = 16$, the system convergence time begins to fluctuate. When $k_{\alpha} = 66$, the system has the shortest convergence time and optimal performance.



Figure 5. The average convergence time of 100 trials. The blue point indicates that the system performance has reached saturation and the red indicates that the system performance has reached optimal. The upper, blue dashed line ordinate is 352.72, on the lower side of which the blue point has the smallest abscissa. The lower line has ordinate 335.93 and passes through the red point. All of the points behind the blue point are between the two dashed lines.

From $k_{\alpha} = 41$ to $k_{\alpha} = 100$, none of the convergence time exceeds the 5% of optimal convergence and the performance does not change significantly, which means the influence of the α drone on surrounding drones has reached saturation at $k_{\alpha} = 41$. When the surrounding drones receive the information of the α drone, although there is interference from γ drones and other β drones, they can take the fastest action to align with the α drone. Even if the weight of the α drone is further increased, the surrounding drones cannot take faster action. The existence of fluctuation is because the flocking is probabilistic but not too large—as mentioned, it does not exceed 5%. The more numerical simulations, the smaller the fluctuation. On the other hand, the number of drones within the interaction range of the α drone is limited due to the limitation of drone density, which constrains the α drone from influencing more drones in a given time. Consequently, although k_{α} continues to increase,

the model's performance cannot be further improved. At this moment, it is essential to

adjust the value of k_{β} to improve model performance. To further study the influence of weighting values on model performance, we simultaneously adjust the values of k_{α} and k_{β} to observe system performance. The results are as shown in Figure 6 and Table 2.



Figure 6. The average convergence time of 100 simulations under different weighting values. (a) Smaller weighting values. (b) Larger weighting values. The value of k_{β} in (a) is absolute, while in (b) it is relative to the value of k_{α} .

Table 2. System performance.

	Convergence Time	k_{lpha}	k_{eta}
Saturation Performance	236.89	30	18 ($0.6k_{\alpha}$)
Optimal Performance	227.95	90	63 (0.7 k_{α})

Figure 6a shows that appropriate parameter settings reduce the convergence time greatly compared with the case of $k_{\alpha} = 1$, $k_{\beta} = 1$. Weighting distinctly improves the system performance, which demonstrates its feasibility and significance. Besides, for any fixed k_{α} , the system performance with $k_{\beta} \ge 1$ is superior to $k_{\beta} < 1$, which verifies the rationality of $k_{\beta} > k_{\gamma}$.

For any fixed k_{β} , the general trend is that the convergence time decreases as k_{α} increases, which is consistent with the result shown in Figure 5. The movement direction of the α drone is the reference state, and the increase of its contribution intensity does not produce any interference information to the system or increase the weight of interference. The continuing increase of k_{α} only becomes meaningless due to the existence of saturated performance, but does not negatively affect system performance.

For fixed k_{α} , when k_{β} is relatively small, the convergence time decreases as k_{β} increases. Compared with γ drones, the proximity of the heading angle of β drones to the reference state is higher. When a β drone owns higher contribution intensity, it induces other drones to converge to the reference state more quickly. When *b* continues to increase, the system performance begins to fluctuate and there is no significant increase. Similar to the α drone, performance fluctuation and saturated performance are also adopted by the β drone. As shown in Figure 6b, when $k_{\alpha} = 90, k_{\beta} = 63$, the system possesses global optimal performance, while when $k_{\alpha} = 30, k_{\beta} = 18$, the system has achieved global saturation performance. For any fixed value of k_{α} , the system has obtained local saturation performance when $k_{\beta} = k_{\alpha}$. Therefore, it is not necessary for the α drone and β drones to be assigned with extremely large values because of the existence of the saturation performance, and it is sufficient that $k_{\alpha} \ge k_{\beta}$.

Furthermore, when $k_{\beta} \gg k_{\alpha}$ —specifically, $k_{\beta} \ge 4k_{\alpha}$ —the system convergence time increases sharply with k_{β} increasing. For a given β drone, besides the α drone, other β drones in the interaction range also affect its movement direction. The movement direction of a β drone may be far from the reference state, and the increase of k_{β} implies the improvement of the contribution intensity of possible interference information. The influence of the movement information of the α drone on the system becomes relatively insignificant due to the larger intensity of other interference information, and given that β drones and α drones align more slowly. Although the β drone induces the γ drone to align with it faster, the current movement direction of the β drone is not necessarily the exact reference state, which is not only meaningless but also may seriously reduce the system performance. Therefore, it is inappropriate that $k_{\beta} \gg k_{\alpha}$.

Accordingly, to a certain extent, the rationality of $k_{\alpha} > k_{\beta} > k_{\gamma}$ has been successfully proved.

4.2.2. Impacts of Interaction Radius

This section mainly discusses the influence of interaction radius on the model. Let $k_{\alpha} = 10, k_{\beta} = 5, v_0 = 0.03, \rho = 4.102, \text{Deg}_{\theta}^d = 0.9984, N = 201, L = 7$. The results are shown in Figures 7 and 8.



Figure 7. The system evolution after 1000 time-steps under different, too short interaction radii. All drones are divided into many clusters. The α drone induces the cluster in the red circle to converge to the reference state successfully.



Figure 8. Average convergence time of 100 simulations with different initial states. To reduce computational burden, the upper limit of time step is 1000.

Figure 7 presents many clusters instead of a whole flocking under r = 0.1, r = 0.2, and r = 0.3, although after a long enough operation time. The α drone induces only a few drones to converge to the reference state, because too small an interaction radius limits the influence range of the α drone and the system is not globally connected. Most drones do not establish direct or indirect interaction with the α drone from start to finish, thereby they do not align with the α drone. The system only spontaneously forms several small clusters with their own unpredictable direction. Therefore, the whole system is unable to converge to the reference state.

As shown in Figure 8, when $r \le 2.3$, the convergence time decreases as the interaction radius increases. The drones that align with the α drone equivalently master the reference state. With the interaction radius increasing, more drones are capable of interacting with the α drone or the drones that master the reference state. This means that information of the reference state holds a larger influence range and a rapider transmission efficiency, and all drones are able to interact with the reference state in a shorter time. Therefore, the drones converge to the reference state more quickly.

Furthermore, when r > 2.3, the larger the interaction radius, the longer the convergence time, which is contrary to our ordinary perception. This is because the correction efficiency of the drone angle is reduced. Correction efficiency refers to the time taken for a drone to align to the reference state after interacting with drones that have mastered the reference state and depends on the intensity of the reference state relative to other interference information. The movement directions of β drones and γ drones are determined by the movement direction of the surrounding drones. It is impossible for all the surrounding drones to align with the α drone; the larger the drone interaction radius, the more interference information the drone may receive. If the contribution intensity of the reference state is not large enough, the drones are more likely to be induced by interference information, thus, increasing the convergence time. Considering this, the system performance can be further improved by adjusting k_{α} and k_{β} .

Therefore, an optimal interaction radius exists when other settings are fixed. r = 2.3 is the optimal radius under the current settings.

4.3. Flocking Navigation and Obstacle Avoidance

This section mainly verifies whether the swarm can achieve flocking navigation according to the WVEM-based flocking algorithm and if better obstacle avoidance can be realized.

4.3.1. The Performance of Flocking Navigation

As long as the movement direction of each drone aligns with the α drone within a short enough time, the drone swarm can move along the predetermined trajectory. This paper considers the flock composed of 15 drones, among which there is one α drone. The attraction term and repulsion term are added and the periodic boundaries removed. In this section, a certain amount of noise is added; let $v_0 = 0.1$, $\rho = 4.102$, r = 1, N = 15, $c^{align} = 1$, $c^{rep} = 0.05$, $r^{rep} = 0.7$, $c^{att} = 0.001$, $r^{att} = 1$, $\eta = 0.1$. At the initial moment, the coordinates of the α drone are (0.9585,0.9585); both abscissa and ordinate of each drone obey a [0,1.9170] uniform distribution, and the heading angle obeys [0, 2π] uniform distribution. The α drone moves along a rectangular trajectory. The results are shown in Figure 9.

In Figure 9a, $k_{\alpha} = 50$, $k_{\beta} = 1$; in Figure 9b, $k_{\alpha} = 50$, $k_{\beta} = 16$. Figure 9a shows that some drones detach from the flocking and do not move along the predetermined trajectory, exposing the failure of flocking navigation. Figure 9b shows that in the presence of noise, all drones always align with and follow the α drone closely and move along the predetermined trajectory, which means the system achieves flocking navigation successfully and acquires a certain degree of robustness. To eliminate the influence of noise on system performance and obtain the general conclusions, ten trials of experiments are carried out for each set of parameter settings. For $k_{\alpha} = 50$, $k_{\beta} = 1$, the average number of drones that can move

along the predetermined trajectory is 10; for $k_{\alpha} = 50$, $k_{\beta} = 16$, the average number is 15; all drones can achieve flocking navigation. It is obvious that a suitable weighting value plays a significant role for WVEM. The flocking navigation can be achieved based on WVEM if the parameter settings are appropriate.



Figure 9. The trajectories of all drones. The red line shows the trajectory of the α drone. The blue lines represent the drones that move along the predetermined trajectory. The black lines indicate drones that have deviated from the predetermined trajectory.

4.3.2. The Performance of Obstacle Avoidance

In this paper, the layer regulation mechanism is an approach to improve the survival of the swarm when detecting the obstacle, rather than a specific avoidance mechanism. Therefore, the simple obstacle and corresponding avoidance mechanism are set, which are enough to verify the effectiveness of the layer regulation mechanism.

In this section, there is an infinitely long obstacle parallel to the y-axis at x = 13. The detection radius of the drone is equal to the interaction radius, r = 1. For drone *i*, when $x_i \ge 12$, it can detect the obstacle and adjust its movement direction to $\theta_i = \pi/2$. When the layer regulation mechanism is applicable, if β drones or γ drones detect the obstacle, they become α drones and own the reference state $\theta_{i,s_i=2}^{de} = \pi/2$; if α drones detect the obstacle, they adjust reference state $\theta_{i,s_i=3}^{de} = \pi/2$. When the layer regulation mechanism is not applicable, if β drones or γ drones detect the obstacle, their layers remain the same and they are still influenced by repulsion terms and attraction terms of other drones. Let $v_0 = 0.1$, $\rho = 4.102$, r = 1, N = 15, $c^{align} = 1$, $r^{align} = 1$, $c^{rep} = 0.05$, $r^{rep} = 0.7$, $c^{att} = 0.001$, $r^{att} = 1$, $\eta = 0.1$, $k_{\alpha} = 50$, $k_{\beta} = 16$. At the initial moment, the coordinates of the α drone are (0.0959,0.0959), both abscissa and ordinate of other drones obey a [0,1.9170] uniform distribution, and the heading angles of all drones are 0. The results are shown in Figure 10.

Figure 10 presents, in the presence of noise, the success of obstacle avoidance, during which no drone appears $x_i \ge 13$. When drones detect the obstacle, the faster the movement direction of the flocking converges to the reference state, the further the drones are away from the obstacle, and the safer the drones are. To evaluate the system performance in detail, the average distance between all drones and the obstacle within a certain period of time is proposed, which is calculated as follows:

$$Dis = \frac{1}{N} \frac{1}{T_{dis}} \sum_{T_{dis}} \sum_{i=1}^{N} (x_{obs} - x_i),$$
(24)

where $x_{obs} = 13$, $T_{dis} = 20$. The last 20 time-steps are selected for calculations. In Figure 10a, the layer regulation mechanism is applied, Dis = 1.562; in Figure 10b, the layer regulation is not applied, Dis = 1.314. It is obvious that the safe distance with the layer regulation mechanism is larger than without the mechanism. More drones becoming α drones can

possess greater contribution intensity to induce the whole flocking to move in the direction conducive to survival. Therefore, the appropriate application of the layer regulation mechanism can improve system performance, and its robustness and practical value are verified.



Figure 10. The trajectories of all drones under different conditions. (**a**) Applying layer regulation mechanism. (**b**) Not applying layer regulation mechanism.

5. Conclusions

This paper proposes WVEM based on VEM to achieve flocking navigation with only one drone mastering the path information. First, WVEM introduces the α drone, which occupies two characteristics: the specific movement direction and immunity to any other drone. It is verified that the existence of the α drone is the fundamental condition to achieve flocking navigation. Then, all drones are divided into three layers dynamically based on the interaction topology and the drones at different layers are assigned with different weights. The weighting rule is put forward to improve the alignment performance and has been proved through experiments. In addition, a detailed analysis about the influence of interaction radius on model performance is conducted. Then, the layer regulation mechanism is proposed to achieve the flexible obstacle avoidance. Thereby, the flocking navigation and obstacle avoidance composed of 15 drones are achieved, which confirm the reliability of the proposed model and the effectiveness of proposed mechanisms. In the future, we will map the settings to the real world, design a better approach to define the layers and assign weights to improve the model performance, and propose better order parameters to analyze flocking performance.

Author Contributions: Conceptualization, X.X.; Data curation, D.T.; Formal analysis, C.Y.; Funding acquisition, H.Z.; Investigation, X.L.; Methodology, Y.C.; Project administration, Y.C.; Resources, X.X.; Software, X.L.; Supervision, H.Z.; Validation, X.L.; Visualization, X.L. and C.Y.; Writing—original draft, X.L.; Writing—review & editing, Y.C., C.Y. and H.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by National Nature Science Foundation under grant 61803377.

Data Availability Statement: Not appliable.

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

References

- 1. Parrish, J.; Edelstein-Keshet, L. Complexity, pattern, and evolutionary trade-offs in animal aggregation. *Science* **1999**, 284, 99–101 . [CrossRef] [PubMed]
- 2. Couzin, I. Collective minds. Nature 2007, 445, 715. [CrossRef] [PubMed]
- 3. Vicsek, T.; Zafeiris, A. Collective motion. Phys. Rep. 2012, 517, 71–140. [CrossRef]

- 4. Beaver, L.E.; Malikopoulos, A.A. An Overview on Optimal Flocking. *arXiv* **2020**, arXiv:abs/2009.14279.
- Hayes, A.T.; Dormiani-Tabatabaei, P. Self-organized flocking with agent failure: Off-line optimization and demonstration with real robots. In Proceedings of the IEEE International Conference on Robotics & Automation, Washington, DC, USA, 11–15 May 2002.
- Zhou, H.; Zhao, H.; Han, T.; Huang, H.Q. Cooperative flight and evasion control of uav swarm based on rules. *Syst. Eng. Electron.* 2016, *38*, 1374–1382.
- Hoang, D.; Tran, D.M.; Tran, T.S.; Pham, H.A. An adaptive weighting mechanism for reynolds rules-based flocking control scheme. *Peerj Comput. Sci.* 2021, 7, e388. [CrossRef] [PubMed]
- 8. Vicsek, T.; Czirók, A.; Ben-Jacob, E.; Cohen, I.; Shochet, O. Novel type of phase transition in a system of self-driven drones. *Phys. Rev. Lett.* **1995**, *75*, 1226–1229. [CrossRef] [PubMed]
- 9. Baglietto, G.; Albano, E.V.; Candia, J. Gregarious versus individualistic behavior in vicsek swarms and the onset of first-order phase transitions. *Phys. A Stat. Mech. Appl.* **2013**, *392*, 3240–3247. [CrossRef]
- 10. Netzer, G.; Yarom, Y.; Ariel, G. Heterogeneous populations in a network model of collective motion—Sciencedirect. *Phys. A Stat. Mech. Appl.* **2019**, *530*, 121550. [CrossRef]
- 11. Xue, T.; Li, X.; Grassberger, P.; Chen, L. Swarming transitions in hierarchical societies. Phys. Rev. Res. 2020, 2, 042017. [CrossRef]
- 12. Jia, Y.; Vicsek, T. Modelling Hierarchical Flocking. New Phys. Soc. 2019 21, 093048. [CrossRef]
- Vásárhelyi, G.; Virágh, C.; Somorjai, G.; Tarcai, N.; Szörényi, T.; Nepusz, T.; Vicsek, T. Outdoor flocking and formation flight with autonomous aerial robots. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 September 2014; pp. 3866–3873
- 14. Reynolds, C.W. Flocks, Herds, and Schools: A Distributed Behavioral Model. In Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, Anaheim, CA, USA, 27–31 July 1987; pp. 25–34. [CrossRef]