Article

Imitation Learning with Graph Neural Networks for Improving Swarm Robustness under Restricted Communications

Ce Guo , Pengming Zhu , Zhiqian Zhou , Lin Lang, Zhiwen Zeng * and Huimin Lu

Robotics Research Center, College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China; guoce1997@foxmail.com (C.G.); zhupengming@nudt.edu.cn (P.Z.); zhiqian.zhou13@hotmail.com (Z.Z.); langlin_8502@nudt.edu.cn (L.L.); lhmnew@nudt.edu.cn (H.L.)

* Correspondence: zengzhiwen@nudt.edu.cn; Tel.: +86-731-84576455

Abstract: This paper focuses on generating distributed flocking strategies via imitation learning. The primary motivation is to improve the swarm robustness and achieve better consistency while respecting the communication constraints. This paper first proposes a quantitative metric of swarm robustness based on entropy evaluation. Then, the graph importance consistency is also proposed, which is one of the critical goals of the flocking task. Moreover, the importance-correlated directed graph convolutional networks (IDGCNs) are constructed for multidimensional feature extraction and structure-related aggregation of graph data. Next, by employing IDGCNs-based imitation learning, a distributed and scalable flocking strategy is obtained, and its performance is very close to the centralized strategy template while considering communication constraints. To speed up and simplify the training process, we train the flocking strategy with a small number of agents and set restrictions on communication. Finally, various simulation experiments are executed to verify the advantages of the obtained strategy in terms of realizing the swarm consistency and improving the swarm robustness. The results also show that the performance is well maintained while the scale of agents expands (tested with 20, 30, 40 robots).

Keywords: swarm robustness; graph importance entropy; graph convolutional networks; imitation learning

Citation: Guo, C.; Zhu, P.; Zhou, Z.; Lang, L.; Zeng, Z.; Lu, H. Imitation Learning with Graph Neural Networks for Improving Swarm Robustness under Restricted Communications. *Appl. Sci.* **2021**, *11*, 9055. <https://doi.org/10.3390/app11199055>

Academic Editor: Yara Khaluf

Received: 20 August 2021

Accepted: 24 September 2021

Published: 28 September 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

1.1. Motivation

Multi-robot systems have a wide range of applications in everyday life, such as target tracking [1,2], search and rescue [3], task sequencing and assignment [4], and collaborative monitoring [5]. One of the most prominent research topics on multi-robot systems is the flocking control [6,7], and its goal is to achieve velocity consistency and avoid collisions using only local information. Inspired by flocking behaviors such as bird migration [8,9] and fish predation [10,11], researchers have realized coherent and efficient movement of robot swarms, which is crucial for improving the survival rates, reducing energy consumption, and improving the quality and efficiency of completing tasks.

For sparse swarms with a dozen or fewer robots, a new problem has arisen. Sparse robot swarms, especially swarms with sparsely available communications, usually have poor robustness, as evidenced by the frequent splitting of individuals during flocking. Sparsely available communications are specified by limited communication distance and limited communication bandwidth. Due to the lack of redundancy of nodes and links in the swarms, the destruction of a single node or link can easily lead to swarm splitting. Meanwhile, the random initial states coupled with the locally available information makes the swarm converge to at best a locally optimal stable form. In summary, the poor swarm robustness and restricted communications limit further improvements in consistency performance and the application of distributed control strategies to physical robots.

We believe that the poor swarm robustness is a result of the unbalanced distribution of node and link importance. The local information structure can easily lead to the existence of a few critical robot nodes or communication links. These nodes or links clearly occupy more critical positions in the communication topology, and their destruction will cause more damage to the structure of the swarms and even lead to the swarm splitting due to the disrupted communication path. From an internal perspective, the formation of the key nodes or links will increase the overall risk of swarm splitting due to the failure of an individual node or link, and from an external perspective, these critical nodes or links are more likely to be targeted. In conclusion, the formation of the key nodes or links, i.e., the unbalanced distribution of node and link importance, will have a significant impact on causing the swarm to exhibit poor robustness, especially in sparse communication environments.

Therefore, this paper is devoted to generating a distributed flocking control strategy, of which the goal is to achieve better swarm consistency and improve the swarm robustness with sparsely available communications. Meanwhile, we believe that achieving a balanced distribution of node importance and link importance is a key way to improve the swarm robustness.

1.2. Related Work

Formation control, which is one of the most actively studied topics in the field of multi-agent systems, generally aims to drive multiple agents to achieve prescribed geometric constraints on their states [12]. Oh et al. categorized the existing results on formation control into position-, displacement-, and distance-based according to types of sensed and controlled variables [12]. Under position-based control schemes, the agents sense and control their own positions concerning a global coordinate system to achieve the desired formation. Under displacement-based control schemes, the agents sense and control the relative positions of neighbors. As for the distance-based control scheme, the inter-agent distances are actively controlled, and the individual agent is assumed to be able to sense the relative positions of its neighbor. Ren et al. proposed a position-based formation control scheme and applied it to coordinate the movements of multiple mobile robots [13]. Kang et al. proposed the distance-based adaptive formation control laws for the leader-follower system. In [14], the developed controller enables all the agents to maintain the formation group and move with a constant reference velocity in a plane.

Note that displacement-based formation control usually requires a balanced capability of sensing and communication for practical application. Ji and Egerstedt studied connectivity preservation in displacement-based formation control. By adding appropriate weights to the edges in the graphs, they guarantee the connectivity of the network [15]. Tanner et al. studied the input-to-state stability of displacement-based formation control systems with directed and acyclic interaction topology [16]. Based on the Jacobi over-relaxation algorithm, Cortés has proposed a displacement-based formation control algorithm for single-integrator modeled agents in discrete-time domain [17].

Flocking control is a special category of formation control inspired by biological behavior and is also defined as behavior-based formation control [9]. Like some animal swarms, each individual in a multi-robot system will independently analyze the information and make actions based on local observations. Inspired by bird flocks, Reynolds [9] first proposed three basic rules for flocking: collision avoidance, velocity matching, and flock centering. Based on these three basic rules, many valuable research works have been conducted. Vicsek et al. investigated the orientation consistency of discrete-time model agents based on velocity matching rules. In Vicsek's model, the agents move at roughly the same velocity under noise interference, and the motion state of individual updates according to the average value of the agent and its neighbors' motion directions [18]. By introducing the vector field, Muslimov et al. implemented a circular formation of fixed-wing unmanned aerial vehicles [19]. Based on noncooperative (defective or malicious) robots, Saulniek et al. presented a resilient flocking algorithm for mobile robot swarms [20]. Some other works concentrate on the interactions and topologies in robot swarms. Based on the

locust-inspired case study of coherently marching agents, Rausch et al. have investigated the balance between social feedback and noise as well as the relationship with collective coherence [21]. Moreover, they have demonstrated that the motif type constituting the networks has a paramount influence on group decision-making that cannot be explained solely in terms of degree distribution [22]. Besides, they have also suggested that the scale-free networks could accelerate building up a rapid collective response to cope with the environmental changes [23].

Although there has been a lot of work on flocking control, few researchers have focused on the swarm robustness in the procedure of flocking. Many researchers have devoted their efforts to implementing control strategies for large-scale swarms, where robots are often assumed to be cheap and fragile [24,25]. Thus, the destruction of robots is often ignored. However, when the control strategy is migrated to physical robots, the swarm robustness must be considered, since the splitting of any individual is dangerous and can easily cause the collapse of the whole swarm. Recently, some researchers raised a topic from an adversarial perspective named private flocking [26–28], which is a similar topic to the swarm robustness. Private flocking emphasizes how individuals in a swarm can control themselves to hide the critical nodes and links, which can help avoid swarm splits due to the destruction of core individuals in adversarial environments. However, the lack of an effective quantitative evaluation metric for the performance of different strategies causes its conclusions to be less convincing and the optimization process to be slower. Besides, some of the few existing swarm robustness evaluation metrics require complex calculations [29] or excessive information [30] and are therefore less practical.

Current flocking control strategies can be divided into two categories: centralized control and distributed control. Centralized control strategies plan and control robots' behaviors based on global observations. Taking all robot states into account, they can usually get a globally optimal and complete solution. Typical work of centralized control strategies includes a swarm intelligence (SI)-based centralized flocking solution proposed in [31] and a collision avoidance framework for robot swarms presented in [32]. The former is applied to optimize the wireless sensor network localization and network coverage [31], and the latter can navigate a swarm of autonomous nano-drones to complete the planting task [32]. However, the centralized control strategy always relies too much on the central node and result in a fragile swarm. The larger the swarm scale is, the severer the problem becomes. In contrast, distributed control strategies are built based on local observations and communications. All robots are of equal importance and make their decisions independently, resulting in a uniform computational burden across the robot swarm. Wang et al. proposed a dynamic group learning distributed swarm optimization method, which could be extended for the large-scale cloud workflow scheduling [33]. In [34], Liang et al. presented a distributed swarm control framework for path following, where the dynamic models and velocity measures of underactuated unmanned marine vehicles are inaccurate and uncertain. However, because of the limited observation range and communication range, distributed control strategies easily fall into a local optimum. Therefore, there is a large gap of performance between distributed control strategies and centralized control strategies, which encourages researchers to improve distributed control strategies.

In order to combine the advantages of centralized control strategies and distributed control strategies, some researchers have implemented imitation learning for flocking control in recent years [35,36]. Based on the distributed structure and local observations, behavior cloning of centralized expert strategies is performed, aiming to obtain distributed control strategies of which control effects are close to the global optimums. As a learning-based method, imitation learning also faces the problem of state representation. Currently, the most popular solution is to combine imitation learning with graph neural networks (GNNs) [37–41]. GNNs can effectively extract and uniformly describe the feature of non-Euclidean data and can further help generate scalable distributed strategies. Until now, the GNNs-based imitation learning methods have been successfully applied in multi-robot path planning [37], multi-robot formation control [38], and multi-robot task assignment [39].

Various graph-based networks such as graph recurrent neural networks [41] and graph attention networks [39] are proposed and utilized in recent years.

1.3. Contributions and Paper Organization

In this work, we aim to improve the swarm robustness and consistency under sparsely available communications. The scale of robot swarms is only a dozen or less. Both the communication bandwidth and the communication distance of robots are limited. Besides, since the paper focuses on the flocking control mechanism, the swarm will be represented in an agent-level form. The main contributions of this paper include:

1. The graph importance entropy is defined to quantitatively evaluate the swarm robustness. A quantitative evaluation protocol, where the graph importance entropy is one of the core metrics, is established to determine the best learning-based strategy and rank different distributed flocking control strategies;
2. The importance-correlated directed graph convolutional networks (IDGCNs) are constructed for multidimensional feature extraction and aggregation of graph data under restricted communications. Furthermore, the weighted link contraction degree matrix is designed to portray the distribution of graph importance in real-time while extracting graph structure information for backbone network maintenance. In addition, based on IDGCNs, time-varying topology relations and swarm scale can be described uniformly, which ensures the scalability of the distributed flocking control strategies;
3. The centralized expert strategy is modified to realize the balanced distribution of graph importance entropy. A distributed strategy with performance close to the expert template is generated based on imitation learning, enabling scalable graph importance consistency and scalable swarm robustness.

The rest of the paper is structured as follows: Section 2 formulates the description of robot swarm with restricted communications and the flocking task. Section 3 introduces the quantitative evaluation metric of swarm robustness and the imitative learning approach based on IDGCNs. Section 4 designs several simulation experimental scenarios to compare the distributed strategy obtained in this paper with other distributed flocking control strategies. Section 5 concludes this paper.

2. Problem Formulation

2.1. Description of the Robot Swarm

We consider a distributed swarm consisting of N omnidirectional robots in the two-dimensional space. The sampling time interval is denoted as T_s , and n denotes the sampling sequence number. At time nT_s , the positions, velocities, action policies and observations of robots are denoted by $\mathbf{R}_n = [\mathbf{r}_{1,n}, \mathbf{r}_{2,n}, \dots, \mathbf{r}_{N,n}]^T \in \mathbb{R}^{N \times 2}$, $\mathbf{V}_n = [\mathbf{v}_{1,n}, \mathbf{v}_{2,n}, \dots, \mathbf{v}_{N,n}]^T \in \mathbb{R}^{N \times 2}$, $\mathbf{U}_n = [\mathbf{u}_{1,n}, \mathbf{u}_{2,n}, \dots, \mathbf{u}_{N,n}]^T \in \mathbb{R}^{N \times 2}$ and $\mathbf{X}_n = [\mathbf{x}_{1,n}, \mathbf{x}_{2,n}, \dots, \mathbf{x}_{N,n}]^T \in \mathbb{R}^{N \times F}$, respectively, where F represents the dimension of observation vector. In the following, if there is no superscript, the default is to represent the information at time nT_s . Besides, since omnidirectional robots are modeled as discrete systems, $\mathbf{U}_n$ will last a time interval, from the moment nT_s to the moment $(n+1)T_s$. The discrete dynamic model of the robot is described as:

$$\begin{cases} \mathbf{R}_{n+1} - \mathbf{R}_n = \mathbf{V}_n T_s + \frac{\mathbf{U}_n}{2} T_s^2, \\ \mathbf{V}_{n+1} - \mathbf{V}_n = \mathbf{U}_n T_s. \end{cases} \quad (1)$$

We assume the existence of time-varying local communication constraints, and further assume that each individual can only obtain information from at most the c nearest individuals in the communication area, where the number of c is related to the ability of the communication equipment. At this point, even if the distance between robots is less than the maximum communication distance, it is uncertain that the bidirectional interaction can be achieved, so the communication relationship between nodes can only be represented by a directed graph.

In this work, the robot swarm is described by a directed graph $\mathcal{G}^d = (\mathcal{V}, \mathcal{E})$, of which the vertex set $\mathcal{V} = (v_1, v_2, \dots, v_N)$ and the edge set $\mathcal{E}$ represent the robot nodes and the communication links, respectively. The bidirectional neighbors of v_i are defined as $\mathcal{N}(v_i) = \{v_j | \|\mathbf{r}_i - \mathbf{r}_j\| < d_{max}, i \neq j\}$, where d_{max} is the maximum communication distance. However, since the communication bandwidth of v_i is restricted, the communication links between v_i and its bidirectional neighbors may be blocked. Therefore, we further define directed neighbors of v_i as $\mathcal{N}^d(v_i) = \{v_j | v_j \in \mathcal{N}(v_i) \text{ and } l_{ij} = 1\}$, where $l_{ij} = 1$ indicates that v_j is one of the c nearest individuals in the communication area. We assume that v_i can only receive data from $\mathcal{N}^d(v_i)$, reference information from other individuals in the communication area will be ignored to ensure the communication quality and reduce the burden of memory space. The directed edge which starts from v_j and ends at v_i is denoted as $e_{ji}^d = \langle v_j, v_i \rangle$. Finally, the edge set is rewritten to $\mathcal{E} = \{e_{ji}^d | \forall v_i \in \mathcal{V}, v_j \in \mathcal{N}^d(v_i)\}$.

For the directed graphs, the number of edges pointing to v_i is called the degree of v_i and is denoted as $\deg(v_i)$. A vertex whose observations can be received by the vertex v_i after at least k times data transmissions is called a k -hop neighbor of vertex v_i , and the 0-hop neighbor is the node itself. The subgraph containing all the $\tilde{k}$ -hop ($\tilde{k} = 0, 1, 2, \dots, k$) neighbors and the edges between them are called the k -subgraph of vertex v_i .

2.2. Description of the Flocking Task

As shown in Figure 1, the flocking task is divided into five parts: avoiding collisions, maintaining connectivity, achieving consistency of velocities, one-hop distance and graph importance:

1. Avoiding collisions: we assume that there are no obstacles in the environment, and the collisions may occur only between robots. The safe distance between robots is denoted as d_{min} , then $\forall v_i, v_j \in \mathcal{V}, \|\mathbf{r}_i - \mathbf{r}_j\| > d_{min}$ should be satisfied;
2. Maintaining connectivity: the swarm connectivity is determined by whether there are isolated vertices in the graph, i.e., whether the graph is a connected graph. Meanwhile, as for the each robot, $\forall v_i \in \mathcal{V}, \exists v_j \in \mathcal{V}, \|\mathbf{r}_i - \mathbf{r}_j\| < d_{max}$ should be satisfied;
3. Achieving velocity consistency: the average velocity of the robot swarm is expressed as μ , and the target is to minimize $\sum_{v_i \in \mathcal{V}} (\|\mathbf{v}_i\| - \mu)^2$;
4. Achieving one-hop distance consistency: the one-hop distance is defined as the distance between the robot and its one-hop neighbor. The average of the one-hop distances is expressed as δ , and the target is to minimize $\sum_{e_{ji}^d \in \mathcal{E}} (\|e_{ji}^d\| - \delta)^2$;
5. Achieving graph importance consistency: the graph importance consistency is achieved when the graph importance is balanced distributed, which is evaluated by the graph importance entropy proposed in Section 3.1.

Figure 1. The flocking task. (a) Avoiding collisions and maintaining connectivity. (b) Achieving velocity consistency. (c) Achieving one-hop distance consistency. (d) Achieving graph importance consistency.

3. Methods

This paper proposes an imitation learning method based on importance-correlated directed graph convolutional networks (IDGCNs), of which the framework is illustrated in Figure 2.

Figure 2. The framework of the proposed imitation learning method based on IDGCNs. The part in the pink box will become the distributed flocking control strategy we need by learning from the template.

The swarm states are randomly generated and consist of the initial states of robots and the topology relations of the swarm. All the initial topology graphs are Connected graphs. Next, the expert template is generated by employing the centralized Artificial Potential Field (APF) strategy. Meanwhile, the swarm states are encoded and input to the IDGCNs. The IDGCNs are retrofitted based on graph convolutional networks and can realize the graph data's multidimensional feature extraction and structure-related aggregation. Through L times of tandem graph convolution and graph activation operations, the encoded graph data is mapped to the action policies of the swarms. Later on, the MSE loss function is utilized to evaluate the difference between the expert template and the action policies of IDGCNs. Finally, to minimize the MSE of the sampled batches from the expert template and the IDGCNs' action policies, the IDGCNs are optimized and ultimately become the distributed flocking control strategy we need.

3.1. Quantitative Evaluation Metric for Swarm Robustness

In a sparse robot swarm, robots transfer information via the local and bandwidth-limited communication network. Generally, the more reliable the communication network is, the more robust the swarm could be. Therefore, we define the swarm robustness based on the reliability of the communication network.

In the field of communication, researchers have proposed entropy-evaluation-based methods [42,43] to quantitatively evaluate the reliability of communication networks. Without any prior knowledge about nodes and links, the reliability can be directly defined based on the topological relations. The swarm robustness depends on the distribution of the node and link importance. In other words, the more balanced the distribution is, the less the impact of a single node or link is, and the more robust the robot swarm will be.

In informatics, Shannon's information entropy is used to evaluate the uncertainty of a signal source, and its value indicates the amount of information needed to eliminate the uncertainty of the source. For a random variable X , its information entropy $H(X)$ is defined as:

$$\begin{cases} H(X) = -\sum_{m=1}^M p_m \log_2(p_m), \\ \sum_{m=1}^M p_m = 1, \end{cases} \quad (2)$$

where p_m represents the probability of $X = X_m$ and M is the number of all possible values of X .

When all possible values of X appear with equal probability, $H(X)$ reaches the maximum value $\log M$. In that case, the value of X achieves the maximum uncertainty, and the maximum amount of information is needed to eliminate the uncertainty.

The importance of nodes and links in a swarm can also be considered as a kind of information. Suppose some nodes and links are of far greater importance than others. In that case, the information entropy of the importance distribution will be small, and these critical nodes and links could be identified easily. On the contrast, if all nodes and links are of similar or equal importance, the information entropy of the swarm will be larger, and it is hard to identify critical nodes or links. From the perspective of private flocking, such a swarm will be more robust in adversarial environments. Therefore, we introduce the information entropy into the definition of node importance and link importance.

Note that the graph importance is evaluated from the perspective outside the swarm, so the direction of the data flow is usually difficult to obtain. Therefore, we consider all links to be bidirectional when evaluating the graph importance.

When a node in the swarm breaks down, it also leads to the failure of the communication links directly connected to it. Obviously, the greater the degree of the node, the greater the impact of its destruction on the network. Therefore, the degree of a node can be used to measure its importance, which can be expressed by:

$$p_i = \frac{\deg(v_i)}{\sum_{v_i \in \mathcal{V}} \deg(v_i)}. \quad (3)$$

Notice that $\sum_{v_i \in \mathcal{V}} p_i = 1$. Then, we can further define the node importance entropy based on the node importance:

$$E_N = - \sum_{i=1}^N p_i \log p_i. \quad (4)$$

To define the link importance entropy, we first clarify the generalization of link contraction degree [42]. Link contraction is defined as the contraction of two connected nodes to a new node along the link between them, and the number of nodes directly connected to the new node is the link contraction degree. Analogously to the definition of node importance, the link importance p_{ij} of link e_{ij} can be expressed as:

$$p_{ij} = \begin{cases} \frac{\deg(v_{ij})}{\sum \deg(v_{ij})}, & \text{if } \sum \deg(v_{ij}) \neq 0 \text{ and } e_{ij} \in \mathcal{E}, \\ 0, & \text{otherwise,} \end{cases} \quad (5)$$

where v_{ij} denotes the new node after the contraction along e_{ij} .

We believe that the importance of links is not only reflected in the topological relationship of links but also closely related to the link length. Achieving a consistent link length helps achieve a uniform distribution of nodes, which in turn contributes to a balanced distribution of node and link importance. For a link, the closer the length is to the preset optimal distance, the more critical it is and the more it should be maintained and protected. Based on the above analysis, the link importance decay coefficient ω_{ij} is defined as:

$$\omega_{ij} = 1 - \frac{||\mathbf{r}_{ij}|| - d_{best}}{d_{max}}, \quad (6)$$

where d_{best} represents the preset optimal link length according to the requirements of avoiding collisions and maintaining connectivity, and $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ denotes the relative position.

The new link importance is defined as the product of the initial importance and the link importance decay coefficient. In order to satisfy Equation (2), the decayed link importance is normalized. Finally, the link importance entropy is defined as:

$$E_L = - \sum_{e_{ij} \in \mathcal{E}} \frac{\omega_{ij} p_{ij}}{\sum_{e_{ij} \in \mathcal{E}} \omega_{ij} p_{ij}} \log \frac{\omega_{ij} p_{ij}}{\sum_{e_{ij} \in \mathcal{E}} \omega_{ij} p_{ij}}. \quad (7)$$

When the importance of all links reaches the same and all link lengths converge to the preset link length, the link importance reaches the maximum value $\log M$, where M is the total number of links.

Based on the node importance entropy E_N and link importance entropy E_G , the graph importance entropy E_G can be defined as:

$$E_G = E_N + E_L. \quad (8)$$

In fact, the graph importance entropy is a direct measure of the swarm robustness. The larger the graph importance entropy, the more balanced the distribution of node importance and link importance, and the higher the swarm robustness.

3.2. Importance-Correlated Directed Graph Convolutional Networks

In order to better realize multidimensional feature extraction and topology-related weighted aggregation of non-Euclidean data in restricted communication environments, the IDGCNs are proposed, which can also help to solve the problem of changing observation dimensionality caused by the varying swarm scale and realize more efficient end-to-end mapping. Based on the IDGCNs, the control effect of the learning-based strategy can be optimized and the swarm robustness can be facilitated.

3.2.1. Directed Graph Convolution Layer

Adjacency matrices and Laplace matrices are often used to formally describe the communication relationships between robot nodes, and such matrices are also known as graph shift operators. Considering the time-varying local communication, the graph shift operator at n th sampling moment is uniformly denoted by $\mathbf{S}_n$. The graph shift operation $\mathbf{S}_n(\cdot)$ realizes the linear transfer and recombination of the graph signal along the edge at n th sampling moment.

In this paper, the adjacency matrix based on directed graphs is expressed as:

$$[\mathbf{S}_n^d]_{ij} = \begin{cases} 1, & \text{if } e_{ji}^d \in \mathcal{E}, \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

For the directed graph, the graph shift operator $\mathbf{S}_n^d$ is not guaranteed to be symmetric, which also leads to the fact that the Graph Fourier transformation cannot be executed on the graph signal based on the directed graph, and the graph convolution operation cannot be explained from the perspective of frequency filtering. Based on the assumption of directed graphs, this paper explains the directed graph convolution operation only from the spatial convolution perspective.

We assume that local communication between the node and its neighbors can be achieved only once per sampling period, i.e., the data of k th neighbors acquired by the node in real-time come from the $(n - k)$ th sampling moment. The graph convolution operation based on the assumption of time-varying graph structure and communication delays can be defined as a weighted linear combination of graph shift operation in multidimensional subgraphs:

$$\mathcal{A}_n(\mathbf{X}_{n-k}, \mathbf{S}_{n-k}^d | k = 0, 1, \dots, K-1) = \sum_{k=0}^{K-1} (\mathbf{S}_n^d \mathbf{S}_{n-1}^d \cdots \mathbf{S}_{n-k}^d) \mathbf{X}_{n-k} \mathbf{A}_k, \quad (10)$$

where $\mathbf{S}_{n-k}^d \in \mathbb{R}^{N \times N}$ describes the directed graph structure at the $(n-k)$ th sampling moment, $\mathbf{X}_{n-k} \in \mathbb{R}^{N \times F}$ represents the observation matrix of the swarm at the $(n-k)$ th sampling moment, $\mathbf{A}_k \in \mathbb{R}^{F \times D}$ is the matrix of filter coefficients to be optimized, and the $\mathbf{A}_k$ are shared by each node. K is the max order of the subgraph, which is the maximum range of behavior reference for that node. The larger the value of K , the larger the range for individual behavior reference, which theoretically has a positive impact on the flocking control effect. However, due to the influence of communication delay, the higher the orders of the neighboring nodes, the less reliable their data are, so the value of K needs to be chosen reasonably. The optimal value of K will be discussed in Section 4.2.4.

Obviously, the size of the filter coefficient matrix does not change as long as the dimensionality of the observation and action policy remains unchanged. Therefore, the control strategy has the scalability, which means that the strategy obtained by training on a small-scale swarm can be directly applied to a larger-scale swarm.

3.2.2. Importance-Correlated Directed Graph Convolutional Networks

The whole graph convolutional neural networks consist of L layers of tandem graph neural networks, each layer containing a graph convolutional layer and a graph activation layer. The graph activation layer can be expressed by:

$$\mathbf{X}_l = \sigma[\mathcal{A}_l(\mathbf{X}_{l-1}, \mathbf{S}^d)], \quad (11)$$

where $\mathbf{X}_{l-1} \in \mathbb{R}^{N \times F_{l-1}}$ is the input of the l th layer, $\mathbf{X}_l \in \mathbb{R}^{N \times F_l}$ is the output of the l th layer, $\mathcal{A}_l(\mathbf{X}_{l-1}, \mathbf{S}^d)$ is the output of the graph convolution layer in the l th layer, and σ denotes the graph activation function. The output of the last layer $\mathbf{X}_L = \mathbf{U}_n$ is the action policy of each robot. In this paper, we choose the Tanh activation function as the graph activation function.

Laplace matrix and Adjacency matrix are familiar graph shift operators that assign the same weight to each node in the subgraph. When they are employed as the graph shift operators, as long as communication relationships are established between nodes, the neighbors' information will be aggregated by equal-weight transformation. In this paper, we assume that robot nodes and links have different importance and their weights in the process of data aggregation should be distinguished, so we propose to use a weighted link contraction degree matrix to replace the traditional graph shift operator.

It is worth noting that the calculation of link contraction degree only requires obtaining local information of 1-hop neighbors, in contrast to the calculation of link importance, which requires obtaining contraction degrees of all links in the swarm. Therefore, link contraction degree is selected for the real-time construction of graph shift operators.

The concepts of link contraction $\deg(e_{ij})$ and link importance decay coefficient ω_{ij} are clarified in Section 3.1. In order to evaluate the weight of information from different nodes in a constrained communication environment, this paper defines the directed contraction degree and contraction decay coefficients related to the link length. For the directed graph, this paper specifies that the link can only be contracted along the direction of the directed edges, and the link contraction degree is calculated by counting only the in-degree of the new node after contraction, the directed link contraction is denoted as $\deg(e_{ji}^d)$. The directed link contraction degree decay coefficient ξ_{ij}^d is defined as:

$$\begin{cases} d_i = \frac{\sum \|\mathbf{r}_{ij}\|}{|\mathcal{N}^d(v_i)|}, \{j|v_j \in \mathcal{N}^d(v_i)\}, \\ \xi_{ij}^d = 1 - \frac{\|\mathbf{r}_{ij}\| - d_i}{d_{max}}, \{j|v_j \in \mathcal{N}^d(v_i)\}, \end{cases} \quad (12)$$

where d_i is the average length of all directed edges ending at node v_i , $|\mathcal{N}^d(v_i)|$ is the total number of 1-hop neighbors of node v_i in the directed graph, and d_{max} is the maximum communication distance.

In addition, we use $\omega_{ij}^d \deg(e_{ji}^d)$ to evaluate the weights of information from different links. To ensure the general performance of the control strategy, the weights of all links of the same node are normalized so that the weighted link contraction matrix can be expressed as:

$$[\mathbf{S}^{d*}]_{ij} = \begin{cases} -\frac{\omega_{ij}^d \deg(e_{ji}^d)}{\sum_{v_j \in \mathcal{N}^d(v_i)} \omega_{ij}^d \deg(e_{ji}^d)}, & \text{if } e_{ji}^d \in \mathcal{E}, \\ 1, & \text{if } i = j, \\ 0, & \text{otherwise,} \end{cases} \quad (13)$$

where the value on the diagonal is set to be 1 and the value of the node's neighbors are set to be the opposite of its weighted link contraction degree, since this paper is more concerned with the observation error of the neighboring nodes versus the node itself.

The significance of using the weighted link contraction matrix instead of traditional graph shift operators is to make robot nodes more likely to maintain the stability of links with high contraction degrees and lengths closer to the local average. In other words, using the weighted link contraction matrix can help to improve the attention to the information from the backbone subgraph and important neighbor nodes and reduce the impact of anomalies of non-critical nodes on the whole swarm.

From another perspective, the traditional graph shift operators can only reflect the topological relationship between nodes and ignore the further information of relative positions. Thus, the relative position of nodes corresponding to the same matrix may be completely different. The weighted link contraction matrix takes the link length and the contraction degree of nodes into account, which means that it provides a better portrayal and extraction for the graph structure and can help to achieve better swarm consistency. In Section 4.2.3, we compare the weighted link contraction matrix and other traditional graph shift operators.

3.3. Learning from the Expert Template

In this section, a distributed flocking control strategy is generated by imitative learning from the centralized expert template. In order to improve the swarm robustness, the centralized APF strategy is utilized based on fixed and fully connected topology to achieve the balanced distribution of graph importance.

To give full play to the performance of the centralized APF strategy, this paper chooses to execute the strategy in a small-scale swarm and assumes that each individual can access to global information in real-time, which means that the communication topology is fixed and fully connected. According to the demonstration of [44], in this case, the centralized APF strategy can realize velocity consistency and collision avoidance while ensuring the global lowest potential energy.

First, the relative potential energy function between the robot v_i and v_j is defined as:

$$U(\mathbf{r}_i, \mathbf{r}_j) = \frac{d_{best}}{\|\mathbf{r}_{ij}\|} + \log \|\mathbf{r}_{ij}\|, \quad (14)$$

where the position vector of v_i and v_j is denoted as $\mathbf{r}_i$ and $\mathbf{r}_j$, the relative position vector is denoted as $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$. Obviously, when $\|\mathbf{r}_{ij}\| \rightarrow 0$ or $\|\mathbf{r}_{ij}\| \rightarrow \infty$, and the potential energy $U(\mathbf{r}_i, \mathbf{r}_j) \rightarrow \infty$. When $\|\mathbf{r}_{ij}\| = d_{best}$, the relative potential energy between the robot v_i and v_j reaches the minimum value. Therefore, the value of d_{best} represents the ideal distance between robots, and by adjusting the value of d_{best} , the shrinkage or expansion of the swarm can be realized.

The gradient of the relative potential energy is the interaction force between robots. Thus, the centralized APF strategy is defined as:

$$\mathbf{u}_i^* = -\sum_{j=1}^N (\mathbf{v}_i - \mathbf{v}_j) - \sum_{j=1}^N \nabla_{\mathbf{r}_i} U(\mathbf{r}_i, \mathbf{r}_j). \quad (15)$$

The control law consists of two parts, a sum of relative velocity vectors to achieve velocity consistency and a gradient of relative potential energy defining the interaction forces between robots to maintain swarm connectivity and avoid collisions. We design comparative experiments to verify the effectiveness of this strategy for improving swarm robustness, and the related experimental results are shown in Section 4.2.1.

Meanwhile, the distributed APF strategy is defined as:

$$\mathbf{u}_i^d = - \sum_{v_j \in \mathcal{N}^d(v_i)} (\mathbf{v}_i - \mathbf{v}_j) - \sum_{v_j \in \mathcal{N}^d(v_i)} \nabla_{\mathbf{r}_i} U(\mathbf{r}_i, \mathbf{r}_j). \quad (16)$$

Unlike the centralized APF strategy (15), the reference data in Equation (16) only come from neighboring nodes that can achieve direct communication, so Equation (16) is a typical distributed strategy. The distributed APF strategy is used as a benchmark for evaluating distributed learning strategies in Section 4.

According to the goal to realize swarm consistency, the robot observations are designed as follows:

$$\mathbf{x}_i = \left[\sum_{v_j \in \mathcal{N}^d(v_i)} \mathbf{r}_{ij}, \sum_{v_j \in \mathcal{N}^d(v_i)} (\mathbf{v}_i - \mathbf{v}_j), \sum_{v_j \in \mathcal{N}^d(v_i)} \frac{\mathbf{r}_{ij}}{\|\mathbf{r}_{ij}\|^2}, \sum_{v_j \in \mathcal{N}^d(v_i)} \frac{\mathbf{r}_{ij}}{\|\mathbf{r}_{ij}\|^3} \right], \quad (17)$$

where the first two items respectively represent the vector sum of the relative position and relative velocity of robot v_i and its neighbors, and the latter two items are related to the gradient of the potential energy function defined by Equation (14).

The expert template is denoted as $\mathcal{T}$. The element of $\mathcal{T}$ is denoted as $\mathcal{T}_n = \{(\mathbf{X}_n, \mathbf{S}_n^d), \mathbf{u}_n^*, (\mathbf{X}_{n+1}, \mathbf{S}_{n+1}^d), \mathcal{R}_n\}$, where $(\mathbf{X}_{n+1}, \mathbf{S}_{n+1}^d)$ represents the current swarm state, $\mathbf{u}_n^*$ represents the expert action policy of current state, $(\mathbf{X}_n, \mathbf{S}_n^d)$ represents the next swarm state, and $\mathcal{R}_n$ represents the single-step reward. Among them, the transition of the swarm state is determined by the expert action policy and $\mathcal{R}_n$ includes the standard deviation of velocity and one-hop distance, and graph importance entropy. The template contains the solution of multiple episodes, of which the swarm's initial states are randomly generated.

To avoid overfitting and improve the strategy's generalization ability, the expert action policies are directly replaced by the strategy outputs with a certain probability P_u when generating the template [45]. We refer to this operation as the anti-overfitting mechanism, which means that the template will contain not only continuous action policy of the centralized APF strategy, but also the expert solutions for certain stochastic states generated by the unoptimized network. The effect of introducing the anti-overfitting mechanism is further discussed in Section 4.2.2.

The value of P_u has an impact on the generalization ability of the strategy. If P_u is too large, the strategy's generalization ability may be insufficient, and if P_u is too small, the expert template will not provide sufficient continuous guidance. The optimal value of P_u will be discussed in Section 4.2.4.

The objective of imitation learning can be expressed as follows:

$$\mathbf{A}^* = \arg \min_{\mathbf{A}} \sum_{\mathcal{T}_n \in \mathcal{T}} \mathcal{L}(\mathbf{U}_n, \mathbf{u}_n^*), \quad (18)$$

where $\mathbf{A}$ represents the matrix of filter coefficients to be trained and $\mathcal{L}(\mathbf{U}_n, \mathbf{u}_n^*)$ represents the loss function between the expert action policies and the strategy outputs. In this paper, we utilized the MSE loss function. The minimization of the cumulative sum of loss functions is realized by adjusting the values of the coefficients in $\mathbf{A}$.

We set test episodes during the learning process after a specific number of episodes and the new parameters are saved when the strategy achieves better velocity consistency, one-hop distance consistency, and graph importance consistency at the same time.

4. Experiments and Results

In this section, simulation experiments are executed to compare the different distributed strategies. We first propose the evaluation protocol and then thoroughly compare the performance of the strategies in realizing the swarm consistency and avoiding swarm splitting. Validation experiments are executed to verify the effectiveness of the optimization proposed before. The scalability performance, the tracking performance for a given motion trajectory, and the ability to deal with the sudden change of communication topology are also evaluated.

4.1. Evaluation Protocol for the Flocking Task

Most of the existing studies on flocking [38,39,41] have only regarded velocity consistency as the ultimate goal, and the velocity standard deviation is often used as the only evaluation metric for flocking control strategies. This paper has redefined the consistency requirements of the flocking in terms of three aspects: velocity, one-hop distance, and graph importance. In addition, for a flocking task, we have to examine the speed of achieving consistency and the ability to maintain the graph's connectivity and avoid swarm splitting in the flocking process.

Based on the above analysis, this paper proposes a evaluation protocol for the flocking task, which specifically contains the following items:

1. Velocity Reward (VR): the mean of the cumulative single-step velocity standard deviation values for all test episodes, which is used to evaluate the velocity consistency of the swarm. To ensure a positive correlation between the velocity reward and the velocity consistency, the opposite number of the mean value is chosen as the VR, which means that the maximum value of VR is 0;
2. Distance Reward (DR): the mean of the cumulative single-step one-hop distance standard deviation values for all test episodes, which is used to evaluate the one-hop distance consistency of the swarm. Similar to the velocity reward, the DR is also changed to the opposite number;
3. Graph Importance Entropy (GIE): the average of the single-step graph importance entropy for all the test episodes, which is used to evaluate the graph importance consistency and is also the metric to evaluate the swarm robustness;
4. Steady-state time (TS): the average of the time consumed for all episodes in which the variation ranges of single-step VR, DR and GIE all decay and remain within $\pm 2\%$. The TS is to evaluate the speed of achieving consistency in a robot swarm;
5. Success Rate (SR): the success of the flocking task is defined as the situation where the connectivity is well maintained from the beginning to the end of the episode, i.e., none of the robots splits from the swarm in the flocking task. The success rate is defined as the number of successful test episodes as a percentage of the total number of test episodes. This metric is used to evaluate the essential ability of the flocking control strategy to avoid swarm splitting, which is also a direct reflection of the swarm robustness.

Based on the above evaluation protocol, a comprehensive evaluation can be made on the consistency performance and the robustness of the robot swarms in the flocking task. This protocol can be used to evaluate and compare the advantages and disadvantages of different flocking control strategies and optimize the parameters.

It is essential to note that the calculation of the metrics requires the acquisition of the positions and velocities of all robots, so it is an objective evaluation protocol of different strategies but cannot be used to generate the action policy in real-time.

4.2. Validation Experiments

In this subsection, comparative validation experiments are performed, including comparisons of different expert strategies, whether to introduce the anti-overfitting mechanism, different graph shift operators, and different network hyperparameter settings.

4.2.1. Comparison of Expert Strategies

In order to verify the effectiveness of the centralized APF strategy proposed in this paper for improving the swarm robustness, another APF strategy [46] based on dynamic topology relations is selected for comparative experiments. By defining potential truncation surfaces where the potential gradient becomes zero, the computational load of this strategy is significantly reduced, which makes it directly applicable to large-scale flocking control. However, based on the dynamic topology relations, this strategy can only achieve minimal potential energy of individuals in the local range [46], which indicates the unbalanced distribution of graph importance.

In the following, we design a flocking control experiment of seven robots to compare the performance of the two expert strategies. We consider a swarm of $N = 7$ robots with a maximum communication radius of $d_{max} = 1.2$ m, a bandwidth-constraints of $c = 4$, and the sampling time of $T_s = 0.01$ s. The maximum velocity of the robot is set to be 3 m/s. The experiment is executed based on the same random initial state of the swarm, where the initial topology is a connected graph. Figure 3 shows the morphological changes of the swarm under the control of the two strategies.

Figure 3. Variation of swarm morphology under the control of two expert strategies. (a) Three random initial forms of swarms. (b) The stable forms under the control of the APF strategy proposed in this paper. (c) The stable forms under the control of the APF strategy proposed in [46].

Figure 4 shows the variation of the single-step consistency evaluation metrics under the control of the two expert strategies in a specific episode.

Figure 4. Variation of the single-step consistency evaluation metrics under the control of the two expert strategies in a specific episode. (a) Variation of the single-step VR. (b) Variation of the single-step DR. (c) Variation of the single-step GIE.

It is worth noting that during the execution of the centralized expert strategy, information interaction exists between all nodes and the graph is fully connected. In order to ensure the objectivity and consistency of the GIE evaluation, the maximum communication distance is still assumed to exist. As long as the relative distance exceeds the maximum communication distance, it is considered that no edge exists between the two nodes when performing the entropy evaluation.

Through the comparison, it can be found that although the APF strategy proposed achieves the velocity and one-hop distance consistency slightly slower than the APF strategy in [46], the stable values are almost the same. In terms of graph importance entropy, the APF strategy proposed in this paper significantly achieves a better performance, which proves that it can effectively achieve a balanced distribution of graph importance. The balanced distribution can also be intuitively reflected by the final stable form of the swarm. Since the APF strategy proposed in [46] is locally optimal, the final swarm morphology is influenced more by the initial morphology of the swarm, which is contrary to the goal of this paper to achieve a balanced distribution of graph importance and further improve the swarm robustness.

Table 1 lists the values of the consistency evaluation metrics obtained by executing the two expert strategies separately on the test set. We can find that although the strategy proposed in this paper has smaller VR and DR than the strategy proposed in [46], it performs significantly better than the latter in terms of GIE, which validates its better performance in achieving the balanced distribution of graph importance. Furthermore, in terms of the SR, the expert strategy proposed in this paper is also more consistent with the goal of this paper to improve the swarm robustness. In a word, the expert APF strategy proposed in this paper realizes a balanced distribution of graph importance through more individual movements to improve the swarm robustness of the steady-state, and the velocity consistency and one-hop distance consistency before reaching the steady-state are partially sacrificed.

Table 1. Evaluation results under the control of the two expert strategies. The bold data represent the optimal values of different strategies on the same evaluation metric. Same as the tables below.

Strategy	VR	DR	GIE	TS	SR
APF strategy proposed in this paper	−52.878	−72.027	1.891	7.681	1.0
APF strategy proposed in [46]	−29.041	−35.881	1.713	7.881	0.975

4.2.2. Validation of the Anti-Overfitting Mechanism

In order to demonstrate the effectiveness of the anti-overfitting mechanism on improving the strategy's generalization ability, the performance of the strategies learned from the templates with and without the anti-overfitting mechanism is compared on the test set. The evaluation results are shown in Table 2.

Table 2. Evaluation results under the control of the strategies learned from the templates with and without the anti-overfitting mechanism. P_u is set to be 0.9. The bold data represent the optimal values of different strategies on the same evaluation metric.

Template	VR	DR	GIE	TS	SR
With the anti-overfitting mechanism	−209.553	−169.048	1.863	15.925	0.85
Without the anti-overfitting mechanism	−321.548	−209.437	1.851	19.183	0.79

It can be seen from Table 2 that the introduction of the anti-overfitting mechanism results in better performance of both the effectiveness and speed of achieving consistency, especially in terms of the SR. This indicates that the introduction of the anti-overfitting mechanism has a positive impact on improving the strategy's generalization ability.

4.2.3. Comparison of Graph Shift Operators

To verify the advantages of the graph shift operator designed in this paper, the weighted link contraction matrix, Laplace matrix, and Adjacency matrix are respectively used as the graph shift operator to train and test the strategy. The other parameters are kept the same in this process. The experimental results are shown in Table 3.

Table 3. Evaluation results of different graph shift operators.

Graph Shift Operator	VR	DR	GIE	TS	SR
Weighted link contraction matrix	−112.724	−139.456	1.881	11.507	0.945
Laplace matrix	−185.034	−143.733	1.871	13.289	0.885
Adjacency matrix	−209.553	−169.048	1.863	15.925	0.85

From Table 3, it can be seen that after using the weighted link contraction matrix as the graph shift operator, the values of the evaluation metrics are significantly improved. The increased values of velocity reward and distance reward and the less steady-state time represent the better and faster consistency on the velocity and one-hop distance. Furthermore, the improvement of graph importance entropy and success rate especially indicates that the swarm robustness is increased. In summary, the weighted link contraction matrix can significantly help to achieve better swarm consistency and improve the swarm robustness.

4.2.4. Validation of the Hyperparameters

To obtain better hyperparameters, we generated an independent validation set for parameter optimization. This paper mainly discusses three parameters, namely, the number of graph convolution layers L , the max order K of subgraph whose information is aggregated in the graph convolution layer, and the probability P_u of expert policy replacement. Figure 5 shows the evaluation results with different hyperparameters, and the actual values of different evaluation metrics are mapped to the same interval for better visualization.

Figure 5. Evaluation results with different hyperparameters. (a) Results with different L . (b) Results with different K . (c) Results with different P_u .

Considering the performance under different parameters and the complexity of the networks, the final values of the three parameters are chosen to be $L = 2$, $K = 3$, and $P_u = 0.8$. Optimal values of some other hyperparameters are also obtained by validation experiments. In summary, we build a fully connected neural network with hidden layers $L = 2$ and 64 neurons of each layer. The network is implemented in PyTorch, using the Adam optimizer with learning rate 5×10^{-5} and forgetting factors $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The size of the training set is 600 episodes, with 3000 steps per episode. The validation and test set consist of 200 episodes, respectively, with 5000 steps per episode.

4.3. Distributed Control Strategy Comparison Experiments

The distributed flocking control strategy based on IDGCNs is obtained by learning from the expert template. To verify the effectiveness and scalability of the strategy, the fully distributed APF strategy (16) and a distributed learning strategy based on Delayed Aggregation Graph Neural Networks (DAGNNs) [38] are selected for comparison.

4.3.1. Test on Small-Scale Swarms

Table 4 presents the evaluation results for the three strategies mentioned above on the test set. To provide a benchmark for comparison, the test results of the centralized APF strategy proposed in this paper is also listed in the table.

Table 4. Evaluation results under the control of the centralized expert strategy and the three distributed strategies.

Strategy	VR	DR	GIE	TS	SR
Centralized APF strategy	−52.878	−72.027	1.891	7.681	1.0
IDGCNs-based strategy	−112.724	−139.456	1.881	11.507	0.945
DAGNNs-based strategy	−225.831	−49.448	1.669	16.223	0.755
Distributed APF strategy	−1199.25	−21.055	1.215	4.293	0.08

The test results show that, the IDGCNs-based strategy does achieve the performance close to the expert template while respecting the communication constraints.

Table 4 also shows that the control results of the two learning-based strategies are much better than the traditional distributed APF strategy in terms of SR. Based on the same initial states, the distributed APF strategy nearly completely fails to achieve the essential task of maintaining connectivity and avoiding splitting. Comparing the consistency evaluation metrics of the two learning-based control strategies, we can find that the metrics implemented based on IDGCNs are mostly better than those of the learning strategy based on DAGNNs, especially in the metric GIE and SR. The DAGNNs-based strategy is also generated by imitation learning with graph convolutional networks. However, it aims to achieve only velocity consistency, so the method performs poorly in terms of the swarm robustness.

Figure 6 illustrates the final stable form reached by three different distributed control strategies for the same initial states in a specific episode. In addition, Figure 7 records the variation of the single-step consistency evaluation metrics of the two learning-based strategies during the test episode.

Figure 6. Variation of swarm morphology under the control of different control strategies in a specific episode. (a) The initial form of the swarm. (b) The stable form under the control of the centralized APF strategy. (c) The stable form under the control of the IDGCNs-based strategy. (d) The stable form under the control of the DAGNNs-based strategy. (e) The stable form under the control of the distributed APF strategy.

Figure 7. Variation of the single-step consistency evaluation metrics under the control of the two distributed learning-based strategies in a specific episode, in which the red line represents the control result of the IDGCNs-based strategy and the blue dotted line represents the control result of the DAGNNs-based strategy. (a) Variation of the VR. (b) Variation of the DR. (c) Variation of the GIE.

From the changes of swarm forms in Figures 3 and 6, it can be found that the learning strategy based on IDGCNs has learned from the expert strategy to achieve the balanced distribution of graph importance. At the same time, the stable form under the control of the DAGNNs-based strategy is fragile, and the swarm is completely split under the control of distributed APF strategy.

From the variation of the single-step consistency evaluation metrics of the selected episode, it can be found that the time consumed to achieve consistency under the control of the two strategies is almost the same, and the IDGCNs-based strategy eventually achieves better velocity consistency and the DAGNNs-based strategy achieves better one-hop distance consistency. However, the difference of the final steady-state values is minimal. From the variation of the single-step GIE and the average GIE of all the test episodes, it can be found that the IDGCNs-based strategy is much better than the DAGNNs-based strategy. Thus, we believe that achieving the form with a balanced distribution of graph importance is the determining factor for the higher SR of IDGCNs-based control strategy to avoid swarm splitting.

The experimental results show that, on the one hand, the idea of improving swarm robustness by achieving the consistent importance of nodes and links is effective; on the other hand, it also shows that GIE is highly correlated with the swarm robustness and can be used as a quantitative evaluation metric of swarm robustness.

From the evaluation results above, we can conclude that the IDGCNs-based strategy realizes better performance in swarm consistency and swarm robustness than the other two distributed strategies, and achieves the control results close to the centralized template while respecting the realistic communication constraints.

4.3.2. Test of the Scalability

To verify the scalability of the learning-based strategies, we apply the strategies learned in the small-scale swarm directly to the larger-scale swarm ($N = 20, 30, 40$) while keeping other parameters related to the network structure unchanged. The quantitative evaluation results are shown in Table 5.

Table 5. Evaluation results of different swarm scales under the control of the three distributed strategies.

Scale	Strategy	VR	DR	GIE	TS	SR
20	IDGCNs-based strategy	−289.071	−416.025	2.919	169.834	0.955
	DAGNNs-based strategy	−1790.77	−3524.47	2.911	289.037	0.75
	Distributed APF strategy	−9853.81	−41.796	2.374	7.491	0.0
30	IDGCNs-based strategy	−380.834	−2858.52	3.302	154.626	0.97
	DAGNNs-based strategy	−1411.9	−3248.99	3.299	290.176	0.83
	Distributed APF strategy	−11,386.1	−40.631	2.777	7.499	0.0
40	IDGCNs-based strategy	−535.422	−2902.76	3.571	144.961	0.955
	DAGNNs-based strategy	−750.481	−3224.67	3.573	287.891	0.885
	Distributed APF strategy	−12,382.8	−43.578	3.043	8.971	0.0

Table 5 shows that the distributed APF strategy is entirely unable to maintain connectivity as the swarm scale increases. In all episodes, individuals split out of the swarm uncontrollably, so its other consistency metrics are of little significance.

A longitudinal comparison of different swarm scales under the control of IDGCNs-based strategy shows that the strategy trained on small-scale swarms can be directly migrated to larger-scale flocking control. The SR and other consistency evaluation metrics are maintained at a high level, indicating that the learning strategy based on IDGCNs has high scalability.

A horizontal comparison of the control strategies based on IDGCNs and DAGNNs shows that although the DAGNNs-based strategy is also scalable, it is significantly weaker than the IDGCNs-based strategy in terms of the consistency evaluation metrics and the success rate. As the swarm scale increases, the SR based on DAGNNs strategy increases significantly since the redundancy of communication links increases, and therefore the individuals are less likely to split from the swarm. The smaller the size of the swarm and the more restrictive the communication, the more harmful the destruction of a single node and link is to the whole swarm. This is the reason why this paper focuses on improving the robustness of small-scale swarms under restricted communication environments.

4.4. Following Experiment

In the process of flocking control, individuals refer to the behaviors of neighboring nodes to adjust their actions. As a result, in a swarm composed of homogeneous robots, without the introduction of external guidance, the final stable form and velocity depend primarily on the initial state of the swarm, which has a significant degree of uncertainty. In the following, we introduce the leader nodes to the swarm to guide the behaviors of neighboring robots, forming a hierarchical control structure of leader–follower swarm. The following task requires the flocking control strategy to achieve the following motion

and trajectory imitation of the very few leaders with only local information. Since the leader's identity cannot be recognized by other individuals, its behavior has the same reference value as that of other neighboring nodes, and this paper refers such leader nodes as implicit leaders.

We assume that the swarm consists of one leader and six followers. The initial form is a regular hexagonal where the leader is in the center, and the initial position and velocity of each individual are appended with Gaussian noise. The behavior of the implicit leader is preset to a sinusoidal motion with a maximum velocity of 0.8 m/s and a maximum acceleration of 0.5 m/s². We verify whether the three distributed control strategies can effectively realize the following motion while avoiding swarm splitting and maintaining consistency in velocity, one-hop distance, and graph importance. The evaluation results are shown in Table 6.

Table 6. Evaluation results of the three distributed strategies in the following experiment.

Strategy	VR	DR	GIE	SR
IDGCNs-based strategy	−229.239	−351.562	1.868	0.995
DAGNNs-based strategy	−243.937	−372.027	1.862	0.99
Distributed APF strategy	−235.586	−178.002	1.729	0.845

Since the initial form of the swarm exhibits symmetric and tight aggregation, the values of the consistency evaluation metrics are almost the same under the control of the IDGCNs-based and DAGNNs-based learning strategies. In terms of the success rate of avoiding swarm splitting, both of the learning strategies are significantly better than the distributed expert strategy.

Figure 8 plots the sinusoidal motion trajectories and some intermediate morphology of the swarm under the control of the three strategies, where the implicit leader is marked by a larger red dot.

It can be found from the trajectories that by executing the IDGCNs-based strategy, all the individuals achieve the imitation of the sinusoidal motion of the implicit leader, and the motion trajectory exhibits as an approximate standard sinusoidal curve. In the trajectories realized by the distributed APF strategy and the DAGNNs-based strategy, the nodes located along the direction vertical to the motion direction generate trajectories significantly deformed compared with the standard sinusoidal curve. In terms of the intermediate morphology of the swarm, the following motion implemented based on IDGCNs can maintain a more stable swarm morphology during the following process. Furthermore, under the control of the IDGCNs-based strategy, the links are more abundant, and the swarm is more resistant to interference.

4.5. Communication Topology Mutation Experiment

In order to further evaluate the swarm robustness under the control of the IDGCNs-based strategy, we design a communication topology mutation experiment, where we assume that some robots are destructed due to the attacks or their own reasons and new robots merge into the swarm during the flocking process. The strategies are verified whether they can avoid swarm splitting while maintaining the swarm consistency during the dynamic change of the swarm scale and the sudden change of the communication topology.

The experiment also assumes that the swarm is composed of seven robots and the initial form of the swarm is a regular hexagon. The positions and velocities of individuals are attached with Gaussian noise and the maximum velocity of the robot is set to 2 m/s. The test results on the test set are shown in Table 7.

Figure 8. Motion trajectories and intermediate forms of the swarm in the following experiment. (a) Trajectories and intermediate forms of the IDGCNs-based strategy. (b) Trajectories and intermediate forms of the DAGNNs-based strategy. (c) Trajectories and intermediate forms of the distributed APF strategy.

Table 7. Evaluation results of the three distributed strategies in the communication topology mutation experiment.

Strategy	VR	DR	GIE	SR
IDGCNs-based strategy	−1178.36	−552.709	1.802	0.815
DAGNNs-based strategy	−1169.61	−695.476	1.763	0.56
Distributed APF strategy	−1211.969	−659.868	1.614	0.725

The quantitative results show that the velocity and one-hop distance consistency obtained under the control of the three strategies are at the same level. However, the control strategy based on IDGCNs has better performance in the GIE and SR, which means that the swarms controlled by the IDGCNs-based strategy achieve higher robustness.

Figure 9 plots the motion trajectories and some intermediate forms under the control of the three strategies in a specific episode. The breaking point of each trajectory indicating the failure position of the corresponding robot and the start point of each trajectory indicating the departure position of a new robot which is ready to merge into the swarm.

Figure 9. Motion trajectories and intermediate forms of the swarm in the communication topology mutation experiment. (a) Trajectories and intermediate forms of the IDGCNs-based strategy. (b) Trajectories and intermediate forms of the DAGNNs-based strategy. (c) Trajectories and intermediate forms of the distributed expert strategy.

From the motion trajectory of the swarm, it can be seen that the IDGCNs-based strategy can maintain the consistency during the dynamic change of the swarm scale and can constantly adjust the swarm morphology according to scale to achieve a balanced distribution of graph importance, which can help to improve the swarm robustness. Considering the trajectories and intermediate forms under the control of the distributed APF strategy and the DAGNNs-based strategy, it can be found that both of them achieve various morphology because they do not consider the consistency of graph importance, which directly leads to lower swarm robustness and is reflected in the lower success rate of avoiding swarm splitting.

5. Conclusions

This paper focuses on achieving better swarm consistency and improving the swarm robustness with restricted communications. Based on theoretical analysis and experimental verification, the following conclusions can be drawn: First, graph importance entropy can effectively portray the swarm robustness and can be used as a quantitative evaluation metric for swarm robustness. Second, achieving a balanced distribution of graph

importance, i.e., graph importance consistency, effectively improves swarm robustness. Third, compared with the distributed APF strategy and the similar distributed learning strategy based on graph neural networks, the distributed IDGCNs-based strategy has better performance on the effectiveness and speed of achieving swarm consistency. At the same time, the IDGCNs-based strategy has better performance in avoiding swarm splitting, scaling performance, tracking performance, and coping with communication topology mutations. In summary, we propose the experimental results demonstrate that the distributed IDGCNs-based strategy does help improve the swarm robustness and achieve better swarm consistency in restricted communication environments.

In future work, firstly, we would like to apply the proposed approach to physical robots. For practical use, we hope the flocking strategy can be directly applied to large-scale physical robot swarms by training with only a small number of agents. Secondly, we will consider heterogeneous node types and hierarchical communication network structures. Thirdly, we will focus on the flocking control in the presence of static and dynamic obstacles, addressing the rapid propagation of environmental threats in swarms. Fourthly, we will also consider various observations and their limits in reality, such as raw vision information input with limited field-of-view.

Author Contributions: Conceptualization, C.G. and Z.Z. (Zhiwen Zeng); data curation, P.Z. and H.L.; formal analysis, C.G.; investigation, P.Z.; methodology, C.G., P.Z. and Z.Z. (Zhiwen Zeng); project administration, L.L.; resources, C.G. and H.L.; software, C.G.; supervision, Z.Z. (Zhiqian Zhou), L.L., Z.Z. (Zhiwen Zeng) and H.L.; validation, Z.Z. (Zhiqian Zhou), Z.Z. (Zhiwen Zeng) and H.L.; writing—original draft, C.G.; writing—review and editing, P.Z., Z.Z. (Zhiqian Zhou), Z.Z. (Zhiwen Zeng) and H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (61773393, U1913202).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data generated or analyzed during this study are included in this paper or are available from the corresponding authors on reasonable request.

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

References

1. Elfakharany, A.; Ismail, Z.H. End-to-End Deep Reinforcement Learning for Decentralized Task Allocation and Navigation for a Multi-Robot System. *Appl. Sci.* **2021**, *11*, 2895. [\[CrossRef\]](#)
2. Zhu, P.; Dai, W.; Yao, W.; Ma, J.; Zeng, Z.; Lu, H. Multi-robot flocking control based on deep reinforcement learning. *IEEE Access* **2020**, *8*, 150397–150406. [\[CrossRef\]](#)
3. Cardona, G.A.; Calderon, J.M. Robot swarm navigation and victim detection using rendezvous consensus in search and rescue operations. *Appl. Sci.* **2019**, *9*, 1702. [\[CrossRef\]](#)
4. Garattoni, L.; Birattari, M. Autonomous task sequencing in a robot swarm. *Sci. Robot.* **2018**, *3*. [\[CrossRef\]](#)
5. Shah, S.H.H.; Steinnes, O.M.H.; Gustafsson, E.G.; Hameed, I.A. Multi-Agent Robot System to Monitor and Enforce Physical Distancing Constraints in Large Areas to Combat COVID-19 and Future Pandemics. *Appl. Sci.* **2021**, *11*, 7200. [\[CrossRef\]](#)
6. Vászrhelyi, G.; Virágh, C.; Somorjai, G.; Nepusz, T.; Eiben, A.E.; Vicsek, T. Optimized flocking of autonomous drones in confined environments. *Sci. Robot.* **2018**, *3*, eaat3536. [\[CrossRef\]](#)
7. Ibuki, T.; Wilson, S.; Yamauchi, J.; Fujita, M.; Egerstedt, M. Optimization-based distributed flocking control for multiple rigid bodies. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1891–1898. [\[CrossRef\]](#)
8. Lwowski, J.; Majumdar, A.; Benavidez, P.; Prevost, J.J.; Jamshidi, M. Bird flocking inspired formation control for unmanned aerial vehicles using stereo camera. *IEEE Syst. J.* **2021**, *13*, 3580–3589. [\[CrossRef\]](#)
9. Reynolds, C.W. Flocks, herds and schools: A distributed behavioral model. In Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques, New York, NY, USA, 27–31 July 1987; pp. 25–34.
10. Berlinger, F.; Gauci, M.; Nagpal, R. Implicit coordination for 3D underwater collective behaviors in a fish-inspired robot swarm. *Sci. Robot.* **2021**, *6*, eabd8668. [\[CrossRef\]](#)
11. Huth, A.; Wissel, C. The simulation of the movement of fish schools. *J. Theor. Biol.* **1992**, *156*, 365–385. [\[CrossRef\]](#)
12. Oh, K.K.; Park, M.C.; Ahn, H.S. A survey of multi-agent formation control. *Automatica* **2015**, *53*, 424–440. [\[CrossRef\]](#)

13. Ren, W.; Atkins, E. Distributed multi-vehicle coordinated control via local information exchange. *Int. J. Robust Nonlinear Control IFAC-Aff. J.* **2007**, *17*, 1002–1033. [\[CrossRef\]](#)
14. Kang, S.M.; Park, M.C.; Lee, B.H.; Ahn, H.S. Distance-based formation control with a single moving leader. In Proceedings of the 2014 American Control Conference, Portland, OR, USA, 4–6 June 2014; pp. 305–310.
15. Ji, M.; Egerstedt, M. Distributed coordination control of multiagent systems while preserving connectedness. *IEEE Trans. Robot.* **2007**, *23*, 693–703. [\[CrossRef\]](#)
16. Tanner, H.G.; Pappas, G.J.; Kumar, V. Input-to-state stability on formation graphs. In Proceedings of the 41st IEEE Conference on Decision and Control, Las Vegas, NV, USA, 10–13 December 2002; pp. 2439–2444.
17. Cortés, J. Global and robust formation-shape stabilization of relative sensing networks. *Automatica* **2009**, *45*, 2754–2762. [\[CrossRef\]](#)
18. Vicsek, T.; Czirók, A.; Ben-Jacob, E.; Cohen, I.; Shochet, O. Novel type of phase transition in a system of self-driven particles. *Phys. Rev. Lett.* **1995**, *75*, 1226. [\[CrossRef\]](#)
19. Muslimov, T.Z.; Munasypov, R.A. Adaptive decentralized flocking control of multi-UAV circular formations based on vector fields and backstepping. *ISA Trans.* **2020**, *107*, 143–159. [\[CrossRef\]](#)
20. Saulnier, K.; Saldana, D.; Prorok, A.; Pappas, G.J.; Kumar, V. Resilient flocking for mobile robot teams. *IEEE Robot. Autom. Lett.* **2017**, *2*, 1039–1046. [\[CrossRef\]](#)
21. Rausch, I.; Reina, A.; Simoens, P.; Khaluf, Y. Coherent collective behaviour emerging from decentralised balancing of social feedback and noise. *Swarm Intell.* **2019**, *13*, 321–345. [\[CrossRef\]](#)
22. Rausch, I.; Khaluf, Y.; Simoens, P. Collective decision-making on triadic graphs. In *Complex Networks XI, Proceedings of the 11th Conference on Complex Networks, Exeter, UK, 31 March–3 April 2020*; Springer: Berlin, Germany, 2020; pp. 119–130.
23. Rausch, I.; Simoens, P.; Khaluf, Y. Adaptive Foraging in Dynamic Environments Using Scale-Free Interaction Networks. *Front. Robot. AI* **2020**, *7*, 86. [\[CrossRef\]](#) [\[PubMed\]](#)
24. Lyu, D.; Wang, B.; Zhang, W. Large-Scale Complex Network Community Detection Combined with Local Search and Genetic Algorithm. *Appl. Sci.* **2020**, *10*, 3126. [\[CrossRef\]](#)
25. Wei, D.; Wang, F.; Ma, H. Autonomous path planning of AUV in large-scale complex marine environment based on swarm hyper-heuristic algorithm. *Appl. Sci.* **2019**, *9*, 2654. [\[CrossRef\]](#)
26. Zheng, H.; Panerati, J.; Beltrame, G.; Prorok, A. An adversarial approach to private flocking in mobile robot teams. *IEEE Robot. Autom. Lett.* **2020**, *5*, 1009–1016. [\[CrossRef\]](#)
27. Prorok, A.; Kumar, V. Privacy-preserving vehicle assignment for mobility-on-demand systems. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 1869–1876.
28. Li, L.; Bayuelo, A.; Bobadilla, L.; Alam, T.; Shell, D.A. Coordinated multi-robot planning while preserving individual privacy. In Proceedings of the International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 2188–2194.
29. Xiaohong, W.; Zhang, Y.; Lizhi, W.; Dawei, L.; Guoqi, Z. Robustness evaluation method for unmanned aerial vehicle swarms based on complex network theory. *Chin. J. Aeronaut.* **2020**, *33*, 352–364.
30. Wang, X.; Zhang, Y.; Wang, L.; Lu, D.; Sun, Y.; Yao, J.; Wang, X. Task decision-making for UAV swarms based on robustness evaluation. In Proceedings of the IEEE 19th International Conference on Software Quality, Reliability and Security Companion (QRS-C), Sofia, Bulgaria, 22–26 July 2019; pp. 242–248.
31. Mahapatra, C.; Payal, A.; Chopra, M. Swarm intelligence based centralized clustering: a novel solution. *J. Intell. Manuf.* **2020**, *31*, 1877–1888. [\[CrossRef\]](#)
32. Loayza, K.; Lucas, P.; Peláez, E. A centralized control of movements using a collision avoidance algorithm for a swarm of autonomous agents. In Proceedings of the IEEE Second Ecuador Technical Chapters Meeting (ETCM), Salinas, Ecuador, 16–20 October 2017; pp. 1–6.
33. Wang, Z.J.; Zhan, Z.H.; Yu, W.J.; Lin, Y.; Zhang, J.; Gu, T.L.; Zhang, J. Dynamic group learning distributed particle swarm optimization for large-scale optimization and its application in cloud workflow scheduling. *IEEE Trans. Cybern.* **2019**, *50*, 2715–2729. [\[CrossRef\]](#)
34. Liang, X.; Qu, X.; Wang, N.; Li, Y.; Zhang, R. A novel distributed and self-organized swarm control framework for underactuated unmanned marine vehicles. *IEEE Access* **2019**, *7*, 112703–112712. [\[CrossRef\]](#)
35. Riviere, B.; Hönig, W.; Yue, Y.; Chung, S.J. Glas: Global-to-local safe autonomy synthesis for multi-robot motion planning with end-to-end learning. *IEEE Robot. Autom. Lett.* **2020**, *5*, 4249–4256. [\[CrossRef\]](#)
36. Le, H.M.; Yue, Y.; Carr, P.; Lucey, P. Coordinated multi-agent imitation learning. In Proceedings of the International Conference on Machine Learning, PMLR, Volterra, Tuscany, Italy, 14–17 September 2017; pp. 1995–2003.
37. Li, Q.; Gama, F.; Ribeiro, A.; Prorok, A. Graph neural networks for decentralized multi-robot path planning. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 24 October–24 January 2021; pp. 11785–11792.
38. Tolstaya, E.; Gama, F.; Paulos, J.; Pappas, G.; Kumar, V.; Ribeiro, A. Learning decentralized controllers for robot swarms with graph neural networks. In Proceedings of the Conference on Robot Learning, PMLR, Vienna, Austria, 12–18 July 2020; pp. 671–682.
39. Wang, Z.; Gombolay, M. Learning scheduling policies for multi-robot coordination with graph attention networks. *IEEE Robot. Autom. Lett.* **2020**, *5*, 4509–4516. [\[CrossRef\]](#)

40. Hu, T.K.; Gama, F.; Wang, Z.; Ribeiro, A.; Sadler, B.M. Vgai: A vision-based decentralized controller learning framework for robot swarms. *arXiv* **2020**, arXiv:2002.02308.
41. Gama, F.; Tolstaya, E.; Ribeiro, A. Graph neural networks for decentralized controllers. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 5260–5264.
42. Jiang, Y.; Hu, A.; He, M. Evaluation method for the network reliability based on the entropy measures. In Proceedings of the International Conference on Networks Security, Wireless Communications and Trusted Computing, Wuhan, China, 25–26 April 2009; Volume 2, pp. 423–426.
43. Zan, B.; Gruteser, M.; Hu, F. Improving robustness of key extraction from wireless channels with differential techniques. In Proceedings of the International Conference on Computing, Networking and Communications(ICNC), Maui, HI, USA, 30 January–2 February 2012; pp. 980–984.
44. Tanner, H.G.; Jadbabaie, A.; Pappas, G.J. Stable flocking of mobile agents, Part I: Fixed topology. In Proceedings of the 42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475), Maui, HI, USA, 9–12 December 2003; Volume 2, pp. 2010–2015.
45. Ross, S.; Gordon, G.; Bagnell, D. A reduction of imitation learning and structured prediction to no-regret online learning. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, Fort Lauderdale, FL, USA, 11–13 April 2011; pp. 627–635.
46. Tanner, H.G.; Jadbabaie, A.; Pappas, G.J. Stable flocking of mobile agents, Part II: Dynamic topology. In Proceedings of the 42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475), Maui, HI, USA, 9–12 December 2003; Volume 2, pp. 2016–2021.