IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 28 March 2024, accepted 15 April 2024, date of publication 19 April 2024, date of current version 26 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3391374

RESEARCH ARTICLE

Multidimensional Trajectory Prediction of UAV Swarms Based on Dynamic Graph Neural Network

YU AN[1], AO LIU[1], HAO LIU[2,3], AND LIANG GENG[1]
[1]School of Science, Hubei University of Technology, Wuhan 430068, China
[2]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[3]Wuhan Institute of Digital Engineering, Wuhan 430205, China

Corresponding author: Liang Geng (gengliang@hbut.edu.cn)

This work was supported in part by Shandong Natural Science Foundation under Grant ZR202209130044 and in part by the National Natural Science Foundation of China under Grant 62076249.

ABSTRACT The advent of AI and 5G technologies has markedly enhanced the intelligence and connectivity of UAVs, leading to the development of UAV swarms. These swarms not only exhibit superior efficiency and adaptability in collective tasks but also offer considerable potential in both civilian and military sectors. However, despite the innovative insights provided by UAV swarm networking in trajectory forecasting, current approaches face obstacles due to the inherent dynamic complexity of these swarms, often neglecting the data from inter-swarm interactions. This research begins by defining metrics of link channel capacity to record the informational exchanges within UAV swarms, thus laying the foundation for a network of UAV swarms. It then advances a dynamic graph neural network (DynGN) model that utilizes an encoder-decoder structure combining a graph convolutional network with a gated recurrent unit. This model processes both the evolving network configuration and trajectory data of UAV swarms simultaneously, enabling more precise trajectory predictions. Through experiments focusing on prediction accuracy, node number stability, and noise robustness, the effectiveness of the model is assessed. Results indicate that the DynGN model outperforms conventional trajectory prediction models, achieving notable improvements in accuracy and fit quality. Moreover, its robustness against noise in dynamic trajectory data highlights its extensive utility in practical mission contexts.

INDEX TERMS UAV swarm, trajectory prediction, deep learning, graph neural network.

I. INTRODUCTION

A swarm of unmanned aerial vehicles (UAVs) is composed of multiple individual units endowed with advanced interactive capabilities [1]. Through their synergistic actions, these UAVs enhance the swarm's overall performance in collective tasks, meeting the demands of intricate future scenarios [2], [3]. Nevertheless, as UAV swarm intelligence continues to evolve, the task of analyzing and forecasting situations from discrete data, such as position and direction obtained from radar and laser, presents significant challenges. By deducing the dynamic interactions among the swarm's internal nodes from their trajectories and anticipating future movements, improvements can be made in the positioning and tracking capabilities of the UAV swarm, thereby boosting its monitoring and control efficacy.

Currently, the field of flight trajectory prediction is divided into two principal methodologies: traditional dynamic models and deep learning techniques. Traditional methods rely on fundamental flight dynamics and enhanced algorithms, such as the improved Kalman filter to create and refine 4D flight trajectories [4], [5], [6]. These methods, while based on detailed analytical modeling, are limited by their simplicity and the robustness of their models, leading to prediction inaccuracies. These shortcomings are primarily due to their limited ability to deal with complex dynamic systems and to

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

uncover hidden patterns within historical flight data. On the other hand, deep learning techniques mark a significant shift in methodology, utilizing sophisticated computational tools such as principal component analysis, density-based swarming, and a variety of neural network architectures (including convolutional neural networks (CNN), and long short-term memory (LSTM) models) [7], [8], [9], [10], [11]. These approaches overcome the limitations of traditional models by adeptly identifying complex interaction patterns, resulting in highly accurate predictions. Deep learning methods are particularly effective in analyzing the spatial and temporal aspects of flight trajectories, providing a more nuanced understanding of flight dynamics. However, most research in this area has focused on the trajectory prediction of individual unmanned aerial vehicles (UAVs), often overlooking the complex interactions within UAV swarms.

The UAV swarm system utilizes a distributed architecture that facilitates the exchange of information among individual UAV nodes, which is characterized by dynamic evolution. However, the complexity of environments presents substantial challenges to existing algorithms for UAV swarm trajectory prediction, such as low accuracy and insufficient real-time capabilities. Recently, the Graph Neural Network (GNN) has emerged as a groundbreaking neural network architecture, that conceptualizes real-world problems in terms of node connections and message propagation within a graph [12]. This model effectively captures node dependencies and surpasses traditional methods by adeptly learning dynamic patterns in graph structures, thereby significantly enhancing prediction accuracy [13]. GNN employs a predefined graph structure that more accurately represents node interaction patterns and improves interpretability. The priori graph structure, an essential component in graph neural networks, denotes a graph configuration established before model training, typically based on domain-specific knowledge and assumptions [14]. In this study, the Graph Neural Network model uses a predefined graph structure, devised based on domain knowledge, which accounts for variables such as the physical spatial distance and communication capabilities of UAVs, thus effectively mapping potential interaction patterns among them. The inherent mobility and effective transmission range of UAV nodes result in the frequent emergence and dissolution of inter-node links, rendering the UAV swarm network dynamically time-variant. Consequently, this paper enhances the conventional graph neural network algorithm to accommodate the evolving nature of UAV swarms and trajectory feature information through a dynamic graph neural network. Such an adaptation enables the model to assimilate more comprehensive and precise data regarding node interactions throughout the training phase, thereby augmenting the accuracy of UAV swarm trajectory predictions.

By leveraging the inherent behavioral and dynamic similarities within UAV swarms, this study introduces a multidimensional trajectory prediction method that accurately captures the complexities of information exchange and dynamic interactions among units through graph network mapping. The model is structured around two core modules: the Dynamic Network Generation (DNG) module and the Node Dynamics Prediction (NDP) module. The DNG module constructs a dynamic graph network, where link channel capacities are assigned as weights to the network's edges, derived from discrete data such as UAV swarm node positions. This effectively transforms the UAV swarm's evolving informational interactions into a temporal graph network representation. In the NDP module, utilizing the dynamic graph network developed in the DNG module and integrating it with dynamic trajectory data, we implement a parameterized message-passing mechanism. This mechanism is refined through comparison with actual network dynamics and parameter adjustment via backpropagation algorithms, enabling more precise swarm trajectory predictions.

The purpose of this study is to propose a multidimensional trajectory prediction method for realizing the trajectory prediction of UAV swarms. The main contributions and innovations of our work can be summarized as follows:

(a) This model synergizes the capabilities of a graph convolutional neural network and a gated recurrent unit, enabling simultaneous processing of the UAV swarm network structure and trajectory feature information. This innovation addresses the dynamic complexities of swarms, significantly enhancing UAV swarm trajectory prediction performance.

(b) Utilizing link channel capacity as the network edges' weight, this study introduces a novel approach to depicting the information exchange state among UAV swarm nodes. The adoption of this metric not only augments the network model's descriptive power but also offers a quantifiable technique for analyzing UAV swarm interactions.

(c) This paper unveils a groundbreaking multidimensional trajectory prediction approach for UAV swarms. It incorporates structural considerations of swarm network interactions and the network's temporal evolution. By integrating a graph neural network with dynamic graph capabilities, this method achieves heightened accuracy in forecasting UAV swarm trajectories.

The rest of this paper is structured as follows. Section II presents an overview of the current research status in the field and the methods employed. Section III introduces a multidimensional trajectory prediction method for predicting the future motion trajectory of UAV swarms. In Section IV, the experimental design aimed at showcasing the effectiveness of our method in predicting UAV swarm trajectories. The main conclusions are presented in Section V.

II. RELATED WORKS

Graph Neural Networks (GNNs) constitute an advanced segment of deep learning designed specifically for processing graph-structured data, covering complex systems such as knowledge graphs, social networks, and transportation networks. GNNs adeptly extract features from nodes and edges, supporting tasks like graph analysis and synthesis. The

inception of GNNs was marked by Scarselli's introduction of a recursive model for learning node representations within graphs [12]. Subsequently, Bruna introduced the application of Convolutional Neural Network (CNN) principles to graph data, leading to the creation of Graph Convolutional Networks (GCN) that execute convolutions directly on graphs for efficient data encoding [15]. Defferrard refined convolutional filters using k-localized convolution, avoiding the complex computation of Laplacian eigenvectors [16]. Furthermore, Kipf streamlined the convolution process to a single layer, effectively addressing overfitting issues in graphs with varied degree distributions by curbing hierarchical convolution operations and employing multiple layers to capture k-order neighbor dependencies, thereby augmenting network effectiveness [17].

Time series forecasting is a sequence generation challenge, where future sequences are predicated on historical data. Utilizing GNNs, one can adeptly capture the structure and dynamics of time series for forward-looking predictions. Recent efforts have demonstrated the adaptability of Graph Convolutional Networks (GCNs) to time series forecasting. For instance, Yan et al. developed the Spatial-Temporal Graph Convolutional Network (ST-GCN) for skeleton-based action recognition, utilizing graph convolution to identify local relationships and temporal dynamics among joints, thereby extracting advanced features through hierarchical convolutions [18]. Li et al. created a traffic forecasting model that integrates dynamic changes in road networks with static graphs in graph convolutions, enhancing it further with temporal patterns through recurrent neural networks [19]. Additionally, Geng et al. introduced a Spatio-Temporal Multi-Graph Convolutional Network (ST-MGCN) for demand forecasting in ride-hailing services, which assimilates global contextual data to refine temporal modeling [20]. These initiatives highlight the extensive applicability of GNNs in time series forecasting.

III. METHODS

A. MODELING UAV SWARM DYNAMICS

Models of collective behavior in biological swarms, which simulate the local interactions among individuals, can naturally lead to organized group movements such as bird flocking and fish schooling [21], [22], [23]. The core principle of these models is to unite solitary individuals into cohesive units via local interactions, thereby synchronizing their movement states to manifest organized collective behaviors. Extensive research has confirmed that the distributed, adaptive, and robust traits observed in biological swarms mirror the prerequisites for coordinated autonomous control in UAV swarms [24], [25]. Consequently, by applying the principles of biological swarm intelligence to UAV swarm coordination and control, UAVs can exhibit complex behavioral patterns. These patterns, rooted in local sensory perception, emerge from interactions with fellow UAVs and the surrounding environment.

In typical models of biological swarm motion, taking the Couzin model as an example [26]. This model delineates how simple local interactions among individuals can give rise to complex group dynamics. It is characterized by the division of interpersonal interactions into distinct zones, each governing different behavioral responses based on the proximity of neighboring individuals. These zones are typically defined as the repulsion zone, orientation zone, and attraction zone. Additionally, the concept of a field of perception is introduced, often represented as a sectorial area from which an individual can sense its surroundings. Let us consider the position and velocity of each individual to be represented by vectors r_i and v_i, respectively. The movement of individual i at time t is determined based on the following local interaction rules:

1. Repulsion Rule: To avoid collisions, individuals move away from neighbors located within a very close, repulsion zone. If neighbor j is within this zone of individual i, the individual i moves away from j, which can be represented as:

$$d_{ij} = r_i - r_j \tag{1}$$

Here, d_{ij} is the displacement vector from j to i. If $\|d_{ij}\|$ falls below a repulsion threshold, individual i adjusts its direction away from j.

2. Orientation Rule: Within the orientation zone, individuals tend to align their direction of motion with their neighbors. This can be quantified as:

$$\tilde{v}_i = \sum v_j, \quad j \in \textit{Orientation Zone} \tag{2}$$

3. Attraction Rule: Beyond the repulsion and orientation zones lies the attraction zone, where individuals are drawn towards distant neighbors, aiming to move closer:

$$\tilde{r}_i = \sum r_j, \quad j \in \textit{Arientation Zone} \tag{3}$$

At each time step, the direction and velocity of individual i are updated to reflect the cumulative influence of these rules, often necessitating normalization to maintain a constant speed.

The model iteratively updates the positions and velocities of individuals based on the interaction rules. Specifically, the position of individual i at time $t+1$ is updated as:

$$r_i^{(t+1)} = r_i^{(t)} + v_i^{(t)} \Delta t \tag{4}$$

Δt is the time step, and $v_i^{(t)}$ is the velocity vector considering repulsion, orientation, and attraction influences.

The velocity update incorporates the effects of repulsion, orientation, and attraction, formulated as a function f that integrates these influences based on the relative positions and velocities of neighboring individuals:

$$v_i^{(t+1)} = f\left(d_{ij}, \tilde{v}_i, \tilde{r}_i\right) \tag{5}$$

Biological swarm movement models simulate the motion patterns of natural swarms, thereby encapsulating the autonomy and cooperation of spatial movement nodes. Under this model framework, the movement states of UAV nodes

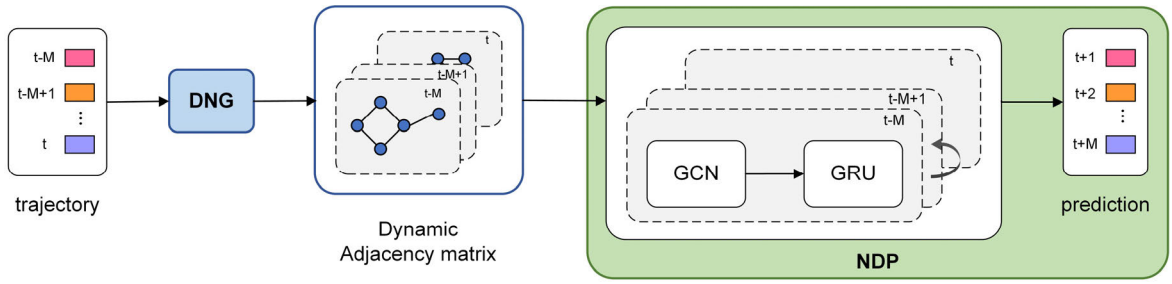


FIGURE 1. A multidimensional trajectory prediction method based on dynamic graph neural networks.

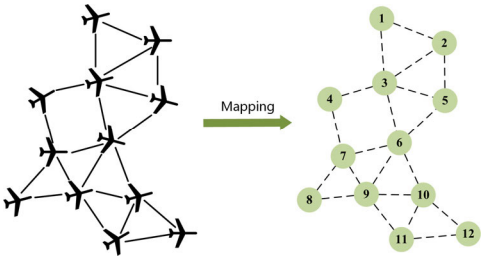


FIGURE 2. Mapping of UAV swarm network.

exhibit high robustness and autonomous control capabilities, enabling the swarm movement model to effectively direct UAV swarms in complex environments. Therefore, we undertake research employing the biological swarm movement model as a basis for developing a UAV swarm movement model.

B. DYNAMIC NETWORK GENERATION FOR UAV SWARM

Utilizing graph theory enables the representation of UAVs as nodes and their communication transmissions as edges, effectively creating a network model of the UAV swarm. This approach allows for the visualization of the swarm's dynamics as alterations in the nodes and edges. The network mapping of the UAV swarm is depicted in Fig. 2.

During the movement of the UAV swarm, the physical positions and distances between different UAVs constantly change, and the capacity of their communication links fluctuates due to path loss, signal fading, and other factors. The communication capacity is always limited by the capacity of the communication channels, regardless of their structure or the nature of information exchange. Channel capacity serves as a key indicator to quantify the communication capabilities between UAV nodes within a swarm. Defined as the maximum quantity of information that can be transmitted through a channel per time unit, it mirrors the efficiency and quality of the UAV communication links. Therefore, the channel capacity can be considered as a physical characteristic of the link, which is used to determine the state of the edges in the network. In this section, we describe the interactive relationships among different nodes in the swarm using the link channel capacity as a feature, construct a dynamic weight matrix to reflect the spatiotemporal dynamics of the UAV swarm nodes, and subsequently build a dynamic graph that captures the temporal changes in the UAV swarm.

Path loss is a significant factor that impacts the quality of wireless signal transmission [27], [28]. For time t, if there is information transmission between UAV i and UAV j, the received signal power is represented by $P_{ij}^{(t)}$. Let P_0 represent the transmitted signal power and u denote the transmission distance between UAV i and UAV j. The received signal power can then be calculated as:

$$P_{ij}^{(t)} = P_0 \times \left(u_{ij}^{(t)}\right)^{-\alpha} \tag{6}$$

α is the path loss exponent, usually ranging from 2 to 4 [29].

Shannon's theorem describes the relationship between the maximum data transmission rate (in bit/s) of a channel and the channel signal-to-noise ratio and bandwidth. According to Shannon's theorem, the data transmission rate of a single channel can be obtained. B represents the channel bandwidth (in HZ), and N_0 represents the noise intensity at the receiver. Then the maximum data transmission rate of the channel C can be expressed as [30]:

$$C = B \times log_2\left(1 + \frac{P_{ij}}{N_0}\right) \tag{7}$$

The maximum data transmission rate of the channel, also known as channel capacity, represents the information transmission capability of that link. By combining Equation (6)-(7), the link capacity between UAV i and UAV j can be obtained as:

$$C_{ij}^{(t)} = B \times log_2\left(1 + \frac{P_0 / \left(u_{ij}^{(t)}\right)^{\alpha}}{N_0}\right) \tag{8}$$

Equation (8) indicates that the distance between nodes is a crucial factor that influences the information transmission capability. The farther the distance between nodes, the greater the path loss, resulting in a decrease in the link capacity between UAV i and UAV j.

The UAV swarm network at time t is denoted by $G^{(t)} = (V^{(t)}, E^{(t)}, W^{(t)})$. $V = \{v_i \,|\, 1 \le i \le |N|\}$ represents the set of UAVs and $|N|$ denotes the total number of UAVs in the network. $E = \{e_{ij} \,|\, 1 \le i, j \le |N|\}$ represents the set of

communication links. W represents the symmetric dynamic adjacency matrix, where w_{ij} is the weighted value on network edges, represents the link channel capacity between UAV i and UAV j. When the link channel capacity between UAV i and UAV j is below a predefined threshold C_0, it indicates that there is no communication link between them, and the value w_{ij} is set to 0. Therefore, the dynamic weight matrix of the UAV swarm network at time t can be expressed as:

$$W_{ij}^{(t)} = \begin{cases} B \times log_2 \left(1 + \dfrac{P_0 / \left(u_{ij}^{(t)}\right)^{\alpha}}{N_0}\right), & C_{ij}^{(t)} > C_0 \\ 0, & C_{ij}^{(t) \leq C_0} \end{cases} \tag{9}$$

C. NODE DYNAMICS PREDICTION FOR UAV SWARM NETWORKS

Assuming that the UAV swarm network evolves in discrete time steps, the predicted dynamic model can be described as:

$$X^{(t+1)} = f(X^{(t)}, W^{(t)}) + \zeta^{(t)} \tag{10}$$

$X^{(t)} = [X_1^{(t)}, X_2^{(t)}, \cdots, X_N^{(t)}] \in \mathbb{R}^{N \times M}$ represents the trajectory information of the swarm at time t, N is the number of nodes, M is the characteristic dimension of each node, and $W^{(t)}$ denotes the adjacency matrix representing the interactive state of the UAV swarm at time t. $\zeta^{(t)} \in \mathbb{R}^{N \times B}$ represents the noise in the data, as trajectory data collected in real environments may contain noise due to factors such as weather and sensor measurement errors.

The model architecture synergistically integrates Graph Convolutional Networks (GCN) and Gated Recurrent Units (GRU) within an encoder-decoder framework [31]. The input of the model is the trajectory feature information of the UAV swarm and the dynamic network adjacency matrix at each time step.

The encoder leverages GCN to process the dynamic adjacency matrix, extracting spatial features that encapsulate the interactions between UAVs. The spatial feature extraction by the GCN can be formalized as:

$$H_{l+1}^{(t)} = \sigma\left(\left(D^{(t)}\right)^{-\frac{1}{2}} W^{(t)} \left(D^{(t)}\right)^{-\frac{1}{2}} H_l^{(t)} \Theta_l\right) \tag{11}$$

Here, $H_l^{(t)}$ denotes the feature matrix at layer l, $W^{(t)} = W^{(t)} + I_N$ is the adjacency matrix with added self-connections represented by I_N. $D^{(t)}$ is the degree matrix of $W^{(t)}$, Θ_l is the weight matrix for layer l, and σ represents ReLU activation function.

To incorporate the graph-convolved features with the pre-existing hidden state H, a weighted average approach is employed to form a dynamic graph convolution layer, as shown in Fig. 3, to enhance the adaptability and capture ability of the model to the dynamic changes of the graph structure:

$$H_{l+1}^{(t)} = \beta H_{l+1}^{(t)} + (1 - \beta) H_l^{(t)} \tag{12}$$

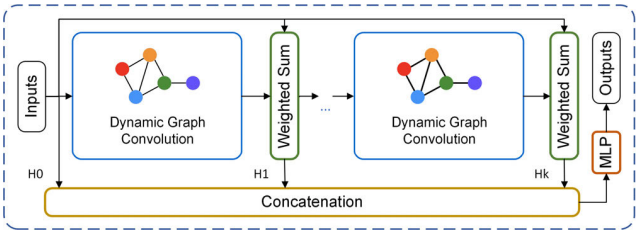


FIGURE 3. The structure of dynamic graph convolution layer module.

β is a weighting factor between 0 and 1, adjusting the importance of the graph-convolved features relative to the original hidden state.

The extracted spatial features from the final GCN layer are then fed into the GRU to model the temporal dynamics, capturing the evolution of UAV interactions over time. The GRU updates for each time step can be summarized as follows:

$$\begin{aligned} z^{(t)} &= \sigma\left(\Theta_z \cdot \left[h^{(t-1)}, H_L^{(t)}\right] + b_z\right) \\ r^{(t)} &= \sigma\left(\Theta_r \cdot \left[h^{(t-1)}, H_L^{(t)}\right] + b_r\right) \\ s^{(t)} &= tanh\left(\Theta_h \cdot \left[r^{(t)} \otimes h^{(t-1)}, H_L^{(t)}\right]\right) + b_n \\ h^{(t)} &= \left(1 - z^{(t)}\right) \otimes h^{(t-1)} + z^{(t)} \otimes s^{(t)} \end{aligned} \tag{13}$$

$H_L^{(t)}$ is the input at time step t, $r^{(t)}$ and $z^{(t)}$ represent the reset and update gate activations respectively. $s^{(t)}$ is the candidate activation, and $h^{(t)}$ denotes the hidden state at time t.

Utilizing the encoder's output, the decoder employs another GRU to predict the future trajectories of the UAV swarm. For prediction step t, it updates as follows:

$$\widehat{h}^{(t)} = \mathrm{GRU}(\widehat{h}^{(t-1)}, h^{(T)}), \quad t = T+1,\ T+2,\ \ldots,\ T+Q \tag{14}$$

Q is the prediction length, and the outputs $\hat{X}^{(t)}$ at future time steps are generated by a fully connected layer:

$$\hat{X}^{(t)} = FC\left(\hat{h}^{(t)}\right) \tag{15}$$

IV. EXPERIMENTS AND RESULTS

In the experiment, UAV swarm trajectory data are generated by mapping based on a typical biological swarm motion model. We selected the search and rescue in the disaster area as the application background and selected the coordinate points (30, 30) as the regional center where the UAVs need to search. The swarm of 5 UAVs is initially randomly distributed in a circular area with a radius of 100 units from the target position until the UAVs hover above the target area (less than 5 units away). By predicting the trajectory of UAVs in disaster areas, rescue teams can deploy drones for search and rescue missions more efficiently. This helps to quickly locate people in need, especially in hard-to-reach areas. The experiment is conducted in the environment of Pytorch 1.10. The entire experimental process is illustrated in Fig. 4.

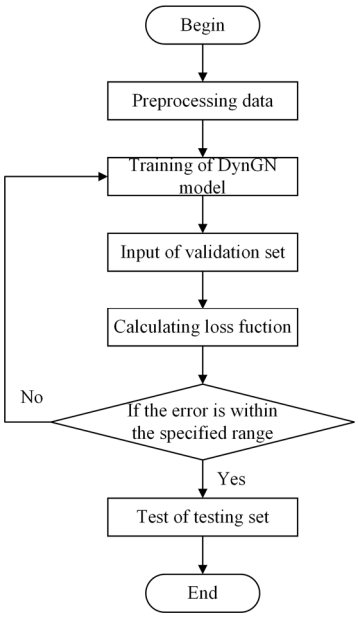


FIGURE 4. Experimental flow chart.

TABLE 1. Datasets of UAV swarm motion trajectory.

Datasets	Number	Length	Nodes	Dimension
Training	7000	100	5	6
Validation	2000	100	5	6
Test	1000	100	5	6

Firstly, the data are normalized and divided into training, validation, and test sets in the ratio of 7:2:1. Then, the data from the training set is input into the model to be used for training, and the data from the validation set is input into the model to calculate the loss function mean square error MSE to iteratively train the model parameters. If the error is within the specified range, the test set data is input into the trained model for prediction, and the model performance is evaluated by comparing the error between the real trajectory and the predicted value, otherwise, the model continues to be trained iteratively until it meets the requirements.

To comprehensively evaluate the predictive performance of our model, we compare the prediction results of our proposed method with commonly used UAV trajectory prediction models currently available. Furthermore, we conduct in-depth examinations of the effectiveness and robustness of our model approach from three aspects: the number of nodes, the prediction time steps, and the degree of noise interference.

A. DATA PREPARATION

Each motion trajectory in the datasets consists of six dimensions: x-coordinate, y-coordinate, z-coordinate, velocity in the x-coordinate, velocity in the y-coordinate, and velocity in the z-coordinate. Table 1 presents a comprehensive description of the UAV swarm trajectory datasets.

1) DATA NORMALIZATION

Due to the significant differences in dimensional scales among various features of the trajectory data, it is necessary to normalize the data before inputting it into the model. We employ the min-max normalization method to eliminate the dimensional differences of variables, defined as follows:

$$X' = \frac{X - min}{max - min} \tag{16}$$

X represents the original sample data, and X' denotes the normalized sample. max and min denote the maximum and the minimum value of the sample.

2) EVALUATION METRICS

To evaluate the predictive performance of our proposed DynGN model, we employ three widely used evaluation metrics for trajectory prediction: 1) Mean Absolute Error (MAE): measures the precision of the predictions by averaging the absolute errors between the predicted values and the observed values. 2) Root Mean Square Error (RMSE): RMSE quantifies the accuracy of the predictions by taking the square root of the expected value of the squared difference between the predicted results and the actual targets. 3) Mean Absolute Percentage Error (MAPE): assesses the relative error of the predictions by comparing the ratio between the error and the actual values. The formulas for calculating these three metrics are as follows:

$$\begin{aligned} \mathrm{MAE}(x, \hat{x}) &= \frac{1}{P} \sum_{t \in P} |x_t - \hat{x}_t| \\ \mathrm{RMSE}(x, \hat{x}) &= \sqrt{\frac{1}{P} \sum_{t \in P} (x_t - \hat{x}_t)^2} \\ \mathrm{MAPE}(x, \hat{x}) &= \frac{1}{P} \sum_{t \in P} \left| \frac{x_t - \hat{x}_t}{x_t} \right| \times 100\% \end{aligned} \tag{17}$$

x_t represents the ground truth values of the UAV swarm trajectory data at time t, $\hat{x}_t$ represents the predicted values of the UAV swarm trajectory data by the model at time t, and P denotes the trajectory time step.

B. EXPERIMENTAL SETTINGS

In this study, we evaluated the predictive performance of the DynGN model and compared it with the BP model, CNN model, LSTM model, and CNN-LSTM model [13]. The structures and parameter settings of each model are as follows:

BP model: This model consists of an input layer, two hidden layers with 64 nodes each, and a fully connected output layer. Dropout layers are inserted after each hidden layer to mitigate overfitting.

CNN model: This model consists of an input layer, two one-dimensional convolutional layers (conv1D) with 64 filters, and a kernel size of 1×3, followed by a fully connected output layer. The ReLU activation function is applied after the convolutional layers.

LSTM model: This model comprises an input layer, two LSTM hidden layers with 64 nodes each, and a fully connected layer. Dropout layers are incorporated after each hidden layer to prevent overfitting.

CNN-LSTM model: This model consists of an input layer, two one-dimensional convolutional layers, two LSTM hidden layers, and a fully connected layer. The structure of the convolutional layers (conv1D) is the same as that in the previous CNN model [13]. After passing through the two convolutional layers, the data flows through two LSTM hidden layers, each with 64 nodes.

DynGN model: This model includes a dynamic graph convolutional recurrent layer, which consists of two GCN layers and two GRU layers. Each GCN layer and GRU layer has 64 hidden nodes. The activation function used is ReLU, and a dropout layer is connected after each hidden layer to prevent overfitting.

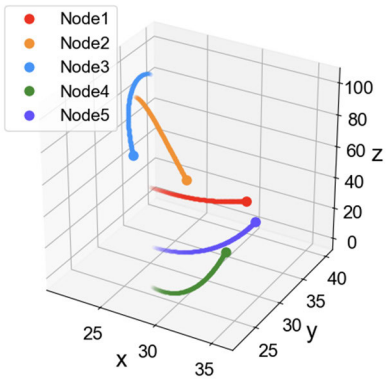
In this experiment, the time step for all models is set to 12. We used the Adam optimizer with an initial learning rate of 0.001. The dropout ratio is set to 0.2. The number of epochs for training is set to 100, and the mean squared error (MSE) is used as the loss function. The batch size for each learning iteration is set to 64.

C. ANALYSIS OF RESULTS

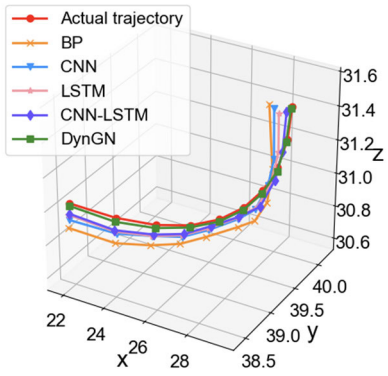
In this experiment, we evaluate the predictive performance of various models by feeding identical UAV swarm trajectory datasets into each. The comparison and analysis are conducted across three dimensions: trajectory prediction accuracy, error metrics utilized for model assessment, and the robustness of the model predictions.

1) COMPARISON BETWEEN REAL TRAJECTORIES AND PREDICTED VALUES

Using a set of UAV swarm trajectories with 5 nodes as an illustration, we examine the single-step prediction outcomes of various models, as demonstrated in Fig. 5. Fig. 5a displays the actual three-dimensional trajectories of the UAV swarm, whereas Fig. 5b contrasts the genuine trajectory of a specific UAV within the swarm against the predictions from different models. From Fig. 5b, it is evident that all models follow the actual trajectory's direction in terms of XYZ coordinates, albeit with varying degrees of accuracy. Notably, the BP and CNN models demonstrate significant deviations from the actual path. Conversely, the LSTM, CNN-LSTM, and our DynGN model present smaller discrepancies in their three-dimensional trajectory predictions. It is significant to mention that prediction inaccuracies for all models are more pronounced on the z-axis when compared to the x and y axes. Specifically, the BP and CNN models show marked variability in their forecasted points at each position. While the LSTM and CNN-LSTM models maintain z-axis prediction errors within a tolerable margin, Fig. 5b highlights that their inaccuracies exceed those of our DynGN model. The DynGN model emerges as the most accurate, closely mirroring the actual trajectories with minimal error, followed



(a) The real trajectory of a group of UAV swarms



(b) Trajectory prediction results for the five models

FIGURE 5. Real and predicted trajectories.

by the CNN-LSTM and LSTM models. The CNN and BP models, however, record larger errors in prediction. Therefore, overall, the ranking of model prediction accuracy is as follows: DynGN > CNN-LSTM > LSTM > CNN > BP.

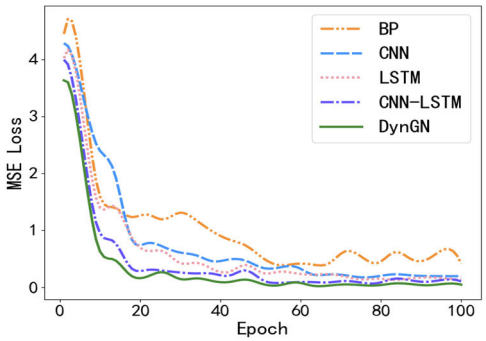
2) COMPARISON OF EVALUATION METRICS

Fig. 6 illustrates the variations in the Mean Squared Error (MSE) loss function for different models on both training and validation datasets. With the exception of the BP model—which exhibits a slower rate of decrease in loss values and minor fluctuations in the later stages of training—the loss function curves of the other four models rapidly decline and reach a state of stability. Notably, the DynGN model demonstrates the lowest loss values on both the training and validation datasets when compared to other models. This superior performance can be attributed to its more intricate network architecture and optimized regularization techniques, which effectively mitigate overfitting and enhance the model's generalization capabilities.

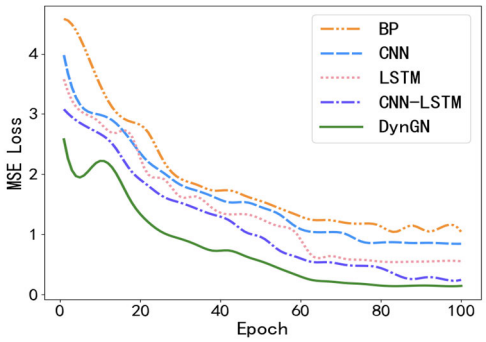
Except for the DynGN model, the other four models (BP model, CNN model, LSTM model, and CNN-LSTM model) can only predict individual UAV trajectories. Therefore, we computed the evaluation metrics for each UAV trajectory within the swarm and took the average as the final evaluation result. Runtime is the time spent per each epoch in seconds.

TABLE 2. Performance of trajectory prediction models.

		BP	CNN	LSTM	CNN-LSTM	DynGN
Dataset1	MAE	0.0728±0.0022	0.0672±0.0015	0.0383±0.0008	0.0109±0.0012	**0.0061±0.0003**
	RMSE	0.1072±0.0034	0.0792±0.0021	0.0474±0.0026	0.0307±0.0019	**0.0091±0.0011**
	MAPE	12.5%±1.08%	10.1%±0.67%	10.9%±0.73%	7.75%±0.86%	**4.65%±0.4%**
Dataset2	MAE	0.0904±0.0085	0.0703±0.0024	0.0142±0.0027	0.0119±0.0016	**0.0079±0.0008**
	RMSE	0.1173±0.0112	0.0864±0.0037	0.0361±0.0048	0.0143±0.0023	**0.0105±0.0013**
	MAPE	15.98%±0.41%	13.46%±0.32%	10.24%±0.41%	10.71%±0.37%	**8.87%±0.28%**
Dataset3	MAE	0.1091±0.0082	0.0726±0.0039	0.0275±0.0052	0.0104±0.0015	**0.0068±0.0007**
	RMSE	0.1203±0.0125	0.0946±0.0059	0.0376±0.0063	0.0152±0.0054	**0.0083±0.0016**
	MAPE	18.95%±0.94%	15.26%±0.43%	12.09%±0.49%	10.08%±0.58%	**7.13%±0.62%**
Average Runtime (s)		**26**	48	62	85	102



(a) Loss in training data



(b) Loss in validation data

FIGURE 6. MSE loss function for different models.

The results in Table 2 indicate that the DynGN model outperforms the other four models in all evaluation metrics, exhibiting significantly lower prediction errors in the task of UAV swarm trajectory prediction. DynGN model, although 20% lower in efficiency than the CNN-LSTM model, is also in the higher category, and 28.01% higher in accuracy than the CNN-LSTM model. Furthermore, the CNN-LSTM and LSTM models exhibit prediction errors markedly lower than those observed in the BP and CNN models. Utilizing the Mean Absolute Percentage Error (MAPE) as a metric, the DynGN model showcases an accuracy (1-MAPE) ranging between 92% and 96% across various prediction tasks, underscoring its consistent predictive prowess. A detailed error analysis across all prediction endeavors reveals that the DynGN model diminishes the average prediction error by 28.01% relative to the CNN-LSTM model, 41.15% in comparison to the LSTM model, 52.65% against the CNN model, and 57.46% when juxtaposed with the BP model. These findings affirm the DynGN model's superior predictive capability across datasets, further validating the dynamic graph network generation and node dynamics modeling's utility in drone swarm trajectory forecasting. While the BP and CNN models are adept at learning data's temporal dependencies, CNN's ability to grasp these dependencies wanes with longer input sequences, affecting prediction accuracy adversely. Though the LSTM and CNN-LSTM models proficiently capture long-term sequence dependencies, the DynGN model excels by dynamically modeling the UAV swarm network and capitalizing on the inter-node information exchange within the swarm. This approach not only boosts prediction precision but also better caters to the requirements of UAV swarm trajectory tracking.

3) MODEL ROBUSTNESS TESTING

Fig. 7a depicts the predictive capabilities of the DynGN model across UAV swarm trajectories featuring varying node counts. The model's performance diminishes with an increase in nodes, achieving optimal results when the node count remains under 50. Conversely, Fig. 6b reveals how enlarging the prediction time step size negatively affects the DynGN model's accuracy in forecasting UAV swarm trajectories, with performance declining as the time step size expands.

In addition, in the real environment where UAV swarms perform missions, swarm trajectory data collected due to various internal and external factors usually contain a lot of noise and uncertainty. In UAV swarm trajectory data, it is usually easy to be disturbed by the coordinate positions and velocity characteristics of nodes. To demonstrate the robustness of our model's prediction in the presence of different degrees of noise in the data, the experiment adds different degrees of noise to the position and velocity features in the simulated generated clean trajectory data. The evaluation results of the model's prediction performance are shown in Fig. 6c. It can be observed that the model performs well when the level of noise interference does not exceed 0.05, but its superiority in predictive performance significantly declines beyond that threshold.

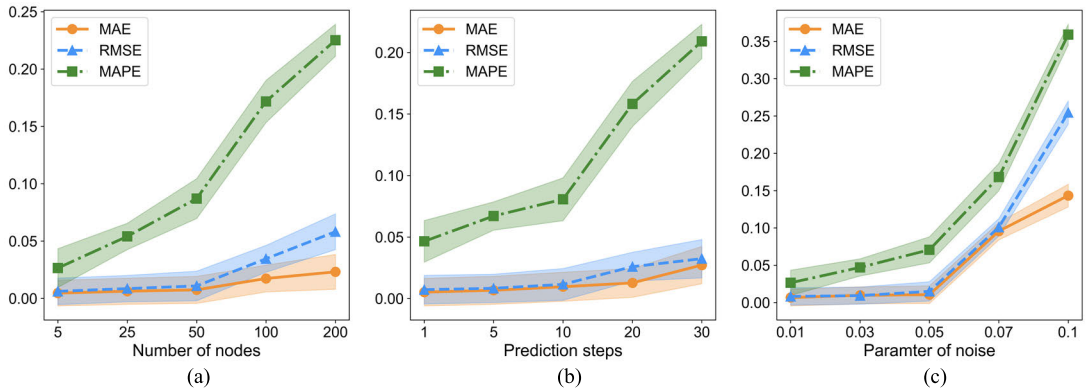


FIGURE 7. Performance of DynGN model prediction.

V. CONCLUSION

We have proposed a multidimensional trajectory prediction method that capitalizes on dynamic flight trajectory data to extract complex pattern information, thus enhancing the prediction of UAV swarm trajectories. The innovation of our approach lies in utilizing link channel capacity attributes derived from this data to construct a temporal graph network representation of the UAV swarm. This allows for the dynamic generation of the UAV swarm network graph. Furthermore, our DynGN model, anchored in the Graph Neural Network (GNN) framework, integrates dynamic trajectory data with the dynamic UAV swarm graph network, enabling effective prediction of UAV swarm trajectories.

The model's performance was evaluated based on accuracy, stability concerning node quantity, and resilience to noise interference. The empirical results highlight our method's proficiency in dynamic network modeling for UAV swarm trajectory prediction. Demonstrating superior predictive accuracy compared to existing models, our method excelled in predicting UAV swarm trajectories in conditions where the node count does not exceed 50 and the noise level is maintained below 0.05. This emphasizes our method's robustness and confirms its suitability for real-world applications where UAV swarms are deployed in mission-critical scenarios.

However, this study highlights several critical areas that warrant further investigation, particularly the need for more comprehensive approaches to obstacle detection and collision avoidance management. Future work will concentrate on exploring these domains, developing UAV swarm control algorithms that integrate advanced obstacle sensing and collision avoidance strategies, and enhancing tracking algorithms to reduce prediction errors. Additionally, we aim to assess the adaptability and robustness of UAV swarms in complex environments, to improve both the overall performance and safety of the system.

REFERENCES

[1] Y. Jia, S. Tian, and Q. Li, "Recent development of unmanned aerial vehicle swarms," *Acta Aeronautica Et Astronautica Sinica*, vol. 41, no. S1, Dec. 2019, Art. no. 723738.

[2] A. Tahir, J. Boling, M.-H. Haghbayan, H. T. Toivonen, and J. Plosila, "Swarms of unmanned aerial vehicles—A survey," *J. Ind. Inf. Integr.*, vol. 16, Dec. 2019, Art. no. 100106.

[3] O. Tutsoy, D. Asadi, K. Ahmadi, and S.-Y. Nabavi-Chasmi, "Robust reduced order Thau observer with the adaptive fault estimator for the unmanned air vehicles," *IEEE Trans. Veh. Technol.*, vol. 72, no. 2, pp. 1601–1610, Feb. 2023.

[4] C. Wang, J. Guo, and Z. Shen, "Prediction of 4D trajectory based on basic flight models," *J. Southwest Jiaotong Univ.*, vol. 44, no. 2, pp. 295–300, 2009.

[5] T. Wang, "4D flight trajectory prediction model based on improved Kalman filter," *J. Comput. Appl.*, vol. 34, no. 6, p. 1812, Jun. 2014.

[6] Y. Lin, J. Zhang, X. Wu, and Y. Liu, "Study on algorithm for flight trajectory prediction based on GMM," *Adv. Eng. Sci.*, vol. 50, pp. 104–109, Jul. 2018.

[7] Z. Wang, M. Liang, and D. Delahaye, "A hybrid machine learning model for short-term estimated time of arrival prediction in terminal manoeuvring area," *Transp. Res. Part C, Emerg. Technol.*, vol. 95, pp. 280–294, Oct. 2018.

[8] H. Zhang, C. Huang, S. Tang, and Y. Xuan, "CNN-based real-time prediction method of flight trajectory of unmanned combat aerial vehicle," *Acta Armamentarii*, vol. 41, pp. 1894–1903, Sep. 2020.

[9] B. Quan and B. Yang, "Prediction model of ship trajectory based on LSTM," *Comput. Sci.*, vol. 45, no. 11A, pp. 126–131, Feb. 2019.

[10] G. Zhong, H. Zhang, J. Zhou, J. Zhou, and H. Liu, "Short-term 4D trajectory prediction for UAV based on spatio-temporal trajectory clustering," *IEEE Access*, vol. 10, pp. 93362–93380, 2022.

[11] L. Ma and S. Tian, "A hybrid CNN-LSTM model for aircraft 4D trajectory prediction," *IEEE Access*, vol. 8, pp. 134668–134680, 2020.

[12] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Jan. 2009, doi: 10.1109/TNN.2008.2005605.

[13] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, Jan. 2020.

[14] A. Sanchez-Gonzalez, N. Heess, J. T. Springenberg, J. Merel, M. Riedmiller, R. Hadsell, and P. Battaglia, "Graph networks as learnable physics engines for inference and control," in *Proc. 35th Int. Conf. Mach. Learn.*, 2018, pp. 4470–4479.

[15] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," 2014, *arXiv:1312.6203*.

[16] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 3844–3852.

[17] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2017, *arXiv:1609.02907*.

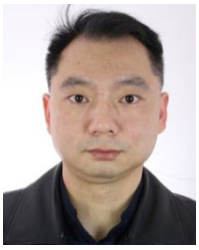
[18] M. Jiang, J. Dong, D. Ma, J. Sun, J. He, and L. Lang, "Inception spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. Int. Symp. Control Eng. Robot. (ISCER)*, Feb. 2022, pp. 208–213.

[19] F. Li, J. Feng, H. Yan, G. Jin, F. Yang, F. Sun, D. Jin, and Y. Li, "Dynamic graph convolutional recurrent network for traffic prediction: Benchmark and solution," *ACM Trans. Knowl. Discovery Data*, vol. 17, no. 1, pp. 1–21, Feb. 2023.
[20] X. Geng, Y. Li, L. Wang, L. Zhang, Q. Yang, J. Ye, and Y. Liu, "Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, 2019, pp. 3656–3663.
[21] J. R. Usherwood, M. Stavrou, J. C. Lowe, K. Roskilly, and A. M. Wilson, "Flying in a flock comes at a cost in pigeons," *Nature*, vol. 474, no. 7352, pp. 494–497, Jun. 2011.
[22] H. Duan and P. Qiao, "Pigeon-inspired optimization: A new swarm intelligence optimizer for air robot path planning," *Int. J. Intell. Comput. Cybern.*, vol. 7, no. 1, pp. 24–37, Mar. 2014.
[23] D. T. Swain, I. D. Couzin, and N. Ehrich Leonard, "Real-time feedback-controlled robotic fish for behavioral experiments with fish schools," *Proc. IEEE*, vol. 100, no. 1, pp. 150–163, Jan. 2012.
[24] S. Misra and P. Agarwal, "Bio-inspired group mobility model for mobile ad hoc networks based on bird-flocking behavior," *Soft Comput.*, vol. 16, no. 3, pp. 437–450, Mar. 2012.
[25] Q. Ouyang, Z. Wu, Y. Cong, and Z. Wang, "Formation control of unmanned aerial vehicle swarms: A comprehensive review," *Asian J. Control*, vol. 25, no. 1, pp. 570–593, Jan. 2023.
[26] I. D. Couzin, J. Krause, R. James, G. D. Ruxton, and N. R. Franks, "Collective memory and spatial sorting in animal groups," *J. Theor. Biol.*, vol. 218, no. 1, pp. 1–11, Sep. 2002.
[27] S. Xiang and J. Yang, "Performance reliability evaluation for mobile ad hoc networks," *Rel. Eng. Syst. Saf.*, vol. 169, pp. 32–39, Jan. 2018.
[28] L. Irio, R. Oliveira, and L. Bernardo, "Aggregate interference in random waypoint mobile networks," *IEEE Commun. Lett.*, vol. 19, no. 6, pp. 1021–1024, Jun. 2015.
[29] A. Jajszczyk, "Introduction to wireless and mobile systems [book review]," *IEEE Commun. Mag.*, vol. 41, no. 11, p. 10, Nov. 2003.
[30] D. J. C. MacKay, *Information Theory, Inference and Learning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
[31] Y. Seo, M. Defferrard, P. Vandergheynst, and X. Bresson, "Structured sequence modeling with graph convolutional recurrent networks," in *Neural Inf. Proc. 25th Int. Conf.*, 2018, pp. 362–373.
[32] W. Chen, L. Chen, Y. Xie, W. Cao, and X. Feng, "Multi-range attentive bicomponent graph convolutional network for traffic forecasting," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 4, 2020, pp. 3529–3536.
[33] Q. Wang, D. Zhuang, and H. Xie, "Identification of influential nodes for drone swarm based on graph neural networks," *Neural Process. Lett.*, vol. 53, no. 6, pp. 4073–4096, Dec. 2021.
[34] Y. Zhang, Y. Guo, Z. Zhang, M. Chen, S. Wang, and J. Zhang, "Universal framework for reconstructing complex networks and node dynamics from discrete or continuous dynamics data," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 106, no. 3, Sep. 2022, Art. no. 034315.
[35] Z. Ghahramani, "Probabilistic machine learning and artificial intelligence," *Nature*, vol. 521, no. 7553, pp. 452–459, May 2015.
[36] J. Li, F. Yang, M. Tomizuka, and C. Choi, "EvolveGraph: Multi-agent trajectory prediction with dynamic relational reasoning," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 19783–19794.
[37] A. Liu, L. Geng, Y. An, H. Wang, and H. Liu, "Unmanned swarm system situation awareness based on dynamic Bayesian network," in *Proc. IEEE Int. Conf. Unmanned Syst. (ICUS)*, Oct. 2022, pp. 448–452.

YU AN received the Ph.D. degree from Harbin Engineering University, China, in 2015. He is currently an Associate Professor with the School of Science, Hubei University of Technology. His research interests include data analysis and decision-making, and information fusion.

AO LIU is currently pursuing the master's degree with the School of Science, Hubei University of Technology, China. Her research interests include data mining and deep learning on trajectory prediction of UAV swarms.

HAO LIU is currently pursuing the Ph.D. degree with Shanghai Jiao Tong University. He is a Professor with Wuhan Digital Engineering Research Institute. His research interests include information fusion and situation cognition.



LIANG GENG received the Ph.D. degree from the Huazhong University of Science and Technology, China, in 2017. He is currently a Professor with the School of Science, Hubei University of Technology. His research interests include data analysis and decision-making, and information fusion.