drones

MDPI

Article

BiDGCNLLM: A Graph–Language Model for Drone State Forecasting and Separation in Urban Air Mobility Using Digital Twin-Augmented Remote ID Data

Zhang Wen [1], Junjie Zhao [2,*], An Zhang [1,3,*], Wenhao Bi [1,3], Boyu Kuang [2], Yu Su [4] and Ruixin Wang [5]

[1] School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, China; zhang.wen.998@mail.nwpu.edu.cn (Z.W.); biwenhao@nwpu.edu.cn (W.B.)
[2] Faculty of Engineering and Applied Sciences (FEAS), Cranfield University, Cranfield MK43 0AL, UK; neil.kuang@cranfield.ac.uk
[3] National Key Laboratory of Aircraft Configuration Design, Xi'an 710072, China
[4] Lincoln Institute for Agri-Food Technology, University of Lincoln, Lincoln LN6 7TS, UK; ysu@lincoln.ac.uk
[5] College of Information Science and Engineering, Northeastern University, Shenyang 110819, China; 2202001@stu.neu.edu.cn
* Correspondence: junjie.zhao@cranfield.ac.uk (J.Z.); zhangan@nwpu.edu.cn (A.Z.)

Abstract

Accurate prediction of drone motion within structured urban air corridors is essential for ensuring safe and efficient operations in Urban Air Mobility (UAM) systems. Although real-world Remote Identification (Remote ID) regulations require drones to broadcast critical flight information such as velocity, access to large-scale, high-quality broadcast data remains limited. To address this, this study leverages a Digital Twin (DT) framework to augment Remote ID spatio-temporal broadcasts, emulating the sensing environment of dense urban airspace. Using Remote ID data, we propose BiDGCNLLM, a hybrid prediction framework that integrates a Bidirectional Graph Convolutional Network (BiGCN) with Dynamic Edge Weighting and a reprogrammed Large Language Model (LLM, Qwen2.5–0.5B) to capture spatial dependencies and temporal patterns in drone speed trajectories. The model forecasts near-future speed variations in surrounding drones, supporting proactive conflict avoidance in constrained air corridors. Results from the AirSUMO co-simulation platform and a DT replica of the Cranfield University campus show that BiDGCNLLM outperforms state-of-the-art time series models in short-term velocity prediction. Compared to Transformer-LSTM, BiDGCNLLM marginally improves the R^2 by 11.59%. This study introduces the integration of LLMs into dynamic graph-based drone prediction. It shows the potential of Remote ID broadcasts to enable scalable, real-time airspace safety solutions in UAM.

Keywords: Urban Air Mobility; remote ID; air corridor; large language model; time series prediction; UAS traffic management; digital twin; drones

check for updates

Academic Editor: Pablo Rodríguez-Gonzálvez

Received: 29 May 2025
Revised: 15 July 2025
Accepted: 17 July 2025
Published: 19 July 2025

Citation: Wen, Z.; Zhao, J.; Zhang, A.; Bi, W.; Kuang, B.; Su, Y.; Wang, R. BiDGCNLLM: A Graph–Language Model for Drone State Forecasting and Separation in Urban Air Mobility Using Digital Twin-Augmented Remote ID Data. *Drones* **2025**, *9*, 508. https://doi.org/10.3390/drones9070508

Copyright: © 2025 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. Background

Urban Air Mobility (UAM) represents a revolutionary shift in air transportation, aiming to ease urban traffic congestion by deploying Electric Vertical Take-off and Landing (eVTOL) aircraft and unmanned aerial vehicles (UAVs) [1]. According to market research data commissioned by NASA, by 2030, there could be up to 500 million package deliveries and 750 million UAM operations in the United States each year, demonstrating the

enormous commercial potential and development prospects of this field [2]. With the anticipated proliferation of autonomous drones operating in structured low-altitude air corridors, ensuring real-time situational awareness and conflict avoidance has become a critical concern. In response, regulatory frameworks such as the U.S. Federal Aviation Administration (FAA) Remote Identification (Remote ID or RID) mandate now require UAVs to broadcast essential flight parameters, including position, velocity, and identification, forming a novel class of spatio-temporal broadcast data [3]. While not derived from traditional satellite or aerial platforms, these broadcasts represent a low-level, distributed sensing layer, offering new opportunities for data-driven urban airspace monitoring, trajectory inference, and conflict risk assessment.

In this context, achieving efficient and reliable conflict detection and resolution within structured air corridors has emerged as a central research priority in UAM. The air corridor used in most studies is shown in Figure 1 [4–7], and the Remote ID data transfer flow is shown in Figure 2. Prior studies have explored speed regulation and general trajectory prediction to mitigate conflict risks [8–11]; however, relatively few have focused on proactively forecasting the short-term speed changes of nearby UAVs, especially those ahead or behind in constrained corridors, based on historical flight dynamics. Such predictive capabilities enable real-time, distributed decision-making in dense airspace without centralised control. Furthermore, despite the growing volume of Remote ID data, access to real-world datasets remains limited due to operational and legal constraints. Furthermore, Bagnall investigated the impact of electromagnetic interference (EMI) and RF desensitisation on the reception of Remote ID signals [12]. To overcome this challenge, researchers have turned to Digital Twin (DT)-augmented platforms that emulate Remote ID broadcasts and UAV interactions, providing a scalable and controllable environment for algorithm development and evaluation.

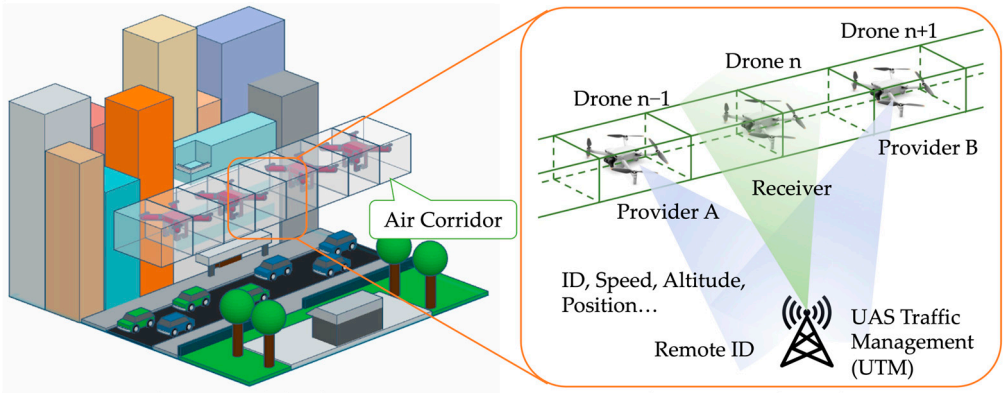


Figure 1. Urban air mobility and urban air corridor.

Against this backdrop, the present study introduces BiDGCNLLM. This hybrid spatio-temporal prediction framework integrates Bidirectional Graph Convolutional Networks (BiGCNs), Dynamic Edge Weight, and a reprogrammed Large Language Model (LLM). By training on Remote ID data streams generated within the AirSUMO, the proposed method aims to forecast imminent speed changes of neighbouring UAVs with high fidelity, thereby facilitating proactive and decentralised conflict avoidance in urban air corridors. Experimental results demonstrate that BiDGCNLLM significantly outperforms several advanced time-series baselines, highlighting its potential as a robust solution for predictive airspace management grounded in remote sensing-inspired data environments.

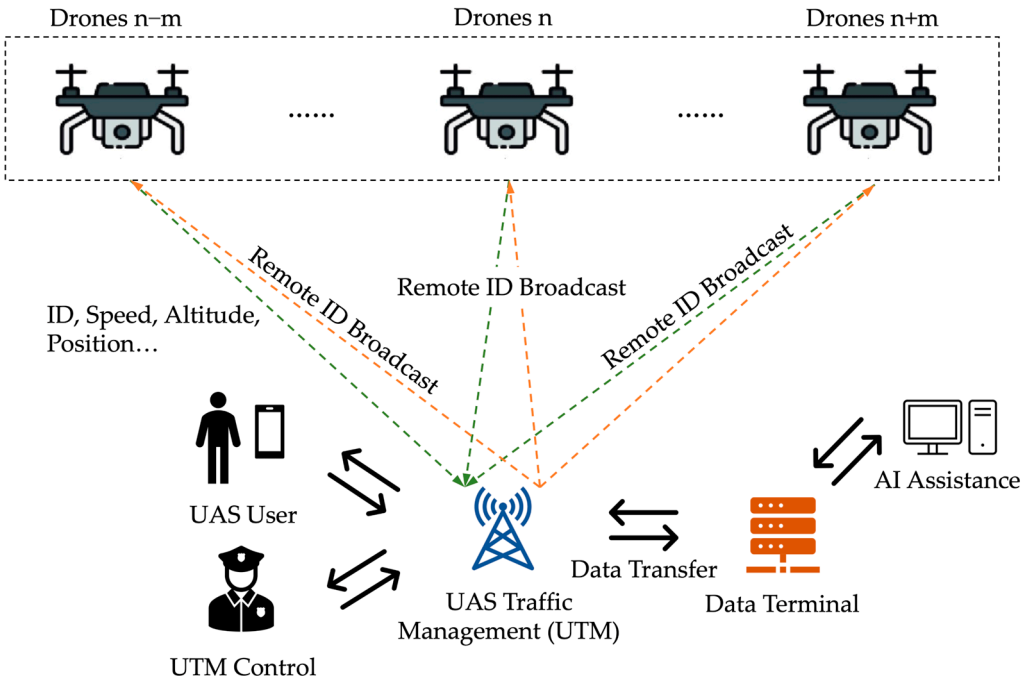


Figure 2. Remote ID and UAM data transfer route.

1.2. Related Work

In recent years, the rapid development of UAM has brought increasing attention to the safe operation within air corridors. Recent studies have proposed a variety of corridor management strategies to mitigate external conflict risks. One approach introduces ground-aligned air routes with one-way flow and spacing constraints [13], while another explores digital traffic light control to manage drone interactions in dense urban environments [14]. A block-based multi-layer corridor design, which limits each cell to a single drone, has also been investigated [15]. Additionally, structured fixed-path logistics systems have been developed to support high-volume urban drone deliveries [16,17]. However, most existing studies emphasise macro-level traffic design, with limited focus on internal corridor operations.

The emergence of UAM has introduced new challenges for low-altitude traffic management, particularly within structured air corridors where autonomous drones must operate safely. In response, a growing body of research has explored conflict detection and resolution in such environments. One study proposed a Remote ID-based UTM framework that applies a reverse-teardrop detection pattern and random forest classifiers to assess the impact of communication latency on conflict alerting [18]. Another approach developed a fuzzy logic-based UTM model leveraging real-time state information to enable timely trajectory adjustments and enhance conflict avoidance performance [19]. A voxelised hexagonal grid indexing method has been introduced to support four-dimensional trajectory prediction in integrated UTM/ATM systems, significantly improving operational safety [20]. Risk assessment frameworks based on volume-based collision models have also been explored, offering practical standards for strategic conflict detection and resolution [21]. Additionally, decentralised receding-horizon NMPC strategies have been applied to manage airspace merging conflicts, demonstrating scalable and robust trajectory deconfliction under UAM conditions [22].

On the modelling front, Graph Neural Networks (GNNs) and deep temporal learning methods have been widely adopted for traffic forecasting. Early efforts introduced Graph Convolutional Networks (GCNs) to capture spatial dependencies through neighbourhood aggregation techniques [23]. Subsequent models integrated graph convolution with tempo-

ral convolution to learn complex spatio-temporal patterns [24]. Recurrent frameworks such as ConvLSTM were developed to represent spatio-temporal dynamics over time [25]. To support dynamic node representations, adaptive graph structures combined with gated recurrent units have also been employed [26]. Transformer-based architectures have recently gained increasing attention in this domain. Some models utilise patching and attention mechanisms to extract multiscale temporal dependencies, while others combine graph convolution with global self-attention to enhance spatio-temporal representation [27,28]. More recent approaches based on LLM further extend temporal reasoning capabilities. For instance, time series have been modelled as token sequences using patch embedding and reprogramming strategies, and temporal attention mechanisms have been applied to predict future trends in traffic speed data [29,30].

DT technologies have also emerged as a valuable tool for simulating and validating UAM operations. Recent studies have leveraged DT frameworks to replicate urban environments and test UAS operations under realistic conditions. For instance, virtual 3D urban airspace models have been used to analyse network structure, air route allocation, and conflict resolution strategies within complex cityscapes [31,32]. DT-based simulation platforms have also supported performance evaluation of traffic management systems, enabling the integration of environmental variables, communication constraints, and flight behaviour modelling [33–35]. These efforts demonstrate the potential of DT to serve as safe, cost-effective, and scalable testbeds for UAM design and validation.

Despite this progress, little attention has been given to modelling Remote ID-like broadcasts in structured UAM corridors, particularly in the context of real-time UAV speed forecasting and proactive conflict mitigation. Most existing models are not explicitly designed for fixed air corridor UAM environments, nor do they consider the unique spatio-temporal dependencies between leading and following UAVs in congested air corridors. Furthermore, integrating LLM with BiGCN and dynamic graph construction remains unexplored mainly in this domain.

1.3. Contributions

The development of UAM has promoted the structured management of low-altitude airspace, where air corridors are becoming a key enabler for large-scale UAV operations. While previous studies have explored route planning and conflict avoidance strategies, research on reducing collision risks through short-term speed prediction remains limited [9,36], particularly under dynamic conditions such as meteorological disturbances and system state fluctuations, which may compromise safe separation intervals. To address this gap, this study proposes a DT–based framework that augments Remote ID broadcasts in structured air corridors and introduces BiDGCNLLM, a hybrid model that integrates BiGCN, Dynamic Edge Weight, and the LLM for short-term UAV speed forecasting and operational safety enhancement. The study is grounded in the hypothesis that combining spatial graph modelling with semantic-aware temporal reasoning can significantly improve prediction accuracy and formation stability in dense UAM scenarios. The proposed method is validated in the AirSUMO. The innovations of this paper are as follows:

- This study is the first to explore the use of LLM for predicting UAV speed within air corridors, aiming to improve the applicability of autonomous UAS operations.
- Augmented Remote ID broadcasts using AirSUMO to generate high-fidelity UAV telemetry data, enabling reliable speed prediction, conflict risk assessment, and DT-based evaluation under UAM scenarios.
- Designed the BiDGCNLLM, a novel model that combines BiGCN with Dynamic Edge Weight and integrates the Qwen2.5–0.5B LLM as its backbone. This architecture leverages the knowledge richness and adaptability of LLM to handle time-series pre-

diction tasks efficiently, achieving high performance while maintaining computational efficiency.
- The proposed model is evaluated through short-term prediction tasks, ablation studies, and comparisons with state-of-the-art time series forecasting baselines. Results show that BiDGCNLLM outperforms most existing methods in prediction accuracy.
- The model is deployed in the AirSUMO and tested in a DT of the Cranfield campus. Speed curves of three UAVs before and after LLM optimisation demonstrate the effectiveness of BiDGCNLLM in improving stability and predictability.

1.4. Organisation

The remainder of this paper is organised as follows: Section 2 presents the proposed BiDGCNLLM methodology, including the hybrid spatio-temporal prediction framework integrating BiGCN, Dynamic Edge Weight, and a reprogrammed LLM. Section 3 details the algorithmic implementation and training process, encompassing data preprocessing, graph construction, GCN Encoder, LLM integration, and comprehensive comparative and ablation experiments. Section 4 describes the deployment of BiDGCNLLM within a high-fidelity DT environment based on the Cranfield University campus, showcasing simulation results and visualisations. It also includes a safety analysis to evaluate the performance of inter-drone conflict mitigation under various configurations. Section 5 provides an in-depth discussion of the results, implications for predictive UAM traffic coordination, and the advantages of integrating GNN and LLM in UAV speed prediction. Section 6 concludes the paper by summarising key findings and outlining future research directions, including large-scale deployment, adaptive traffic regulation, and broader applications of DT-enhanced predictive frameworks in UAM.

2. Methodology

2.1. Overview of Methodology

To enhance coordination and reduce conflict risk in fixed-route UAM scenarios, we propose BiDGCNLLM, a hybrid graph LLM tailored for short-term UAV state forecasting. Central to this framework is the utilisation of DT-augmented remote ID data, which provides structured, high-fidelity spatio-temporal information derived from virtual reconstructions of urban airspace. BiDGCNLLM integrates a BiGCN with Dynamic Edge Weight and an LLM (Qwen2.5–0.5B), enabling the extraction of temporal and relational dependencies across UAV agents. The model encodes relative velocity through graph-based structures and learns temporal dynamics that are critical for short-horizon forecasting. This integration ultimately enables precise speed prediction, supporting conflict mitigation and rhythm stabilisation within structured UAM corridors.

The overall system consists of three core components: (1) a collaborative simulation environment using AirSim and SUMO to simulate low-altitude urban airspace; (2) a 3D DT module for reconstructing structured air corridor operations; and (3) an intelligent decision module powered by BiDGCNLLM. This module builds UAV interaction graphs with dynamic topologies, performs spatial encoding via GCN, and uses LLM-based temporal modelling to forecast speed changes in structured UAM routes.

2.2. BiDGCNLLM

2.2.1. Algorithm Overview

To address the challenge of short-term UAV speed prediction, we propose BiDGCNLLM (Bidirectional Dynamic Edge Weight Graph Convolutional Network with Large Language Model). This cooperative framework combines the dynamic GNN with the LLM. It dynamically constructs drone interaction graphs based on speed similarity, extracts spa-

tial features through bidirectional graph convolutions, and leverages temporal–semantic reasoning via an LLM aligned through a modality-bridging reprogramming layer.

As shown in Figure 3, the data pipeline begins with historical UAV speed inputs. These are used to dynamically generate graph structures and Dynamic Edge Weight, which the GCN then processes to encode spatial relationships. The resulting embeddings are passed through a reprogramming layer into the LLM, which captures deep temporal semantics. The final output is the predicted future speed. The operation framework and data flow are as shown in Figure 4.

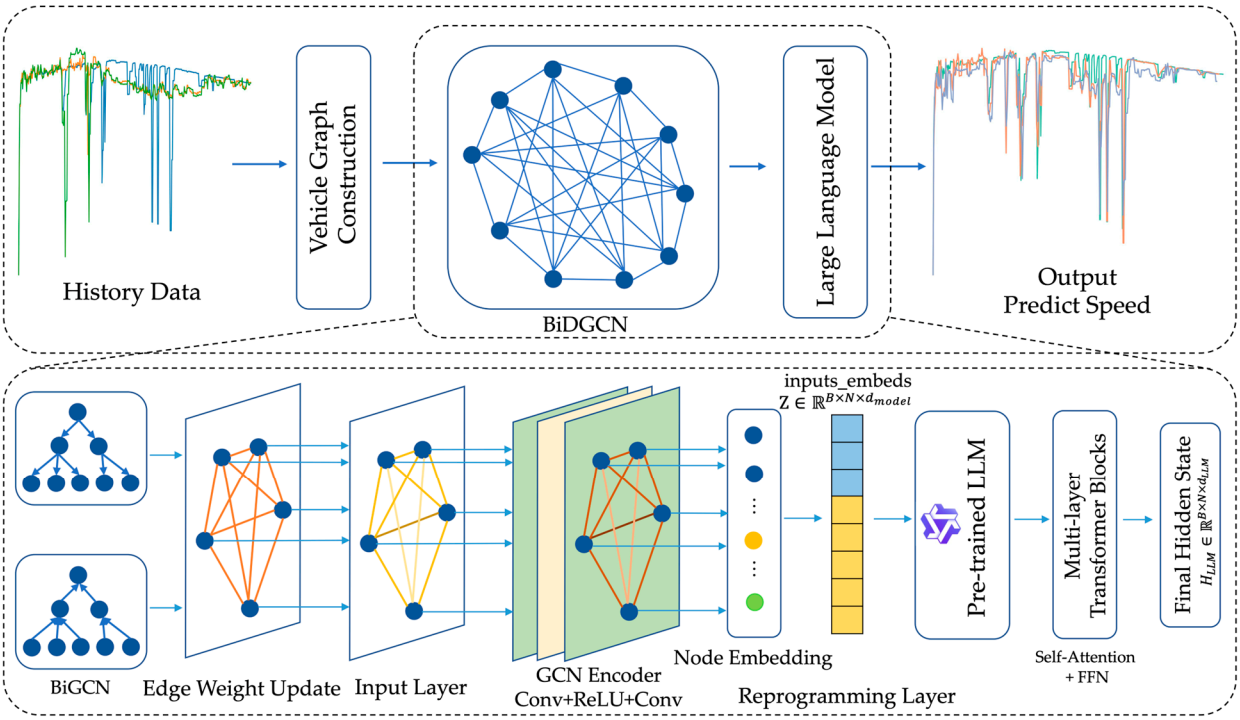


Figure 3. BiDGCNLLM framework.

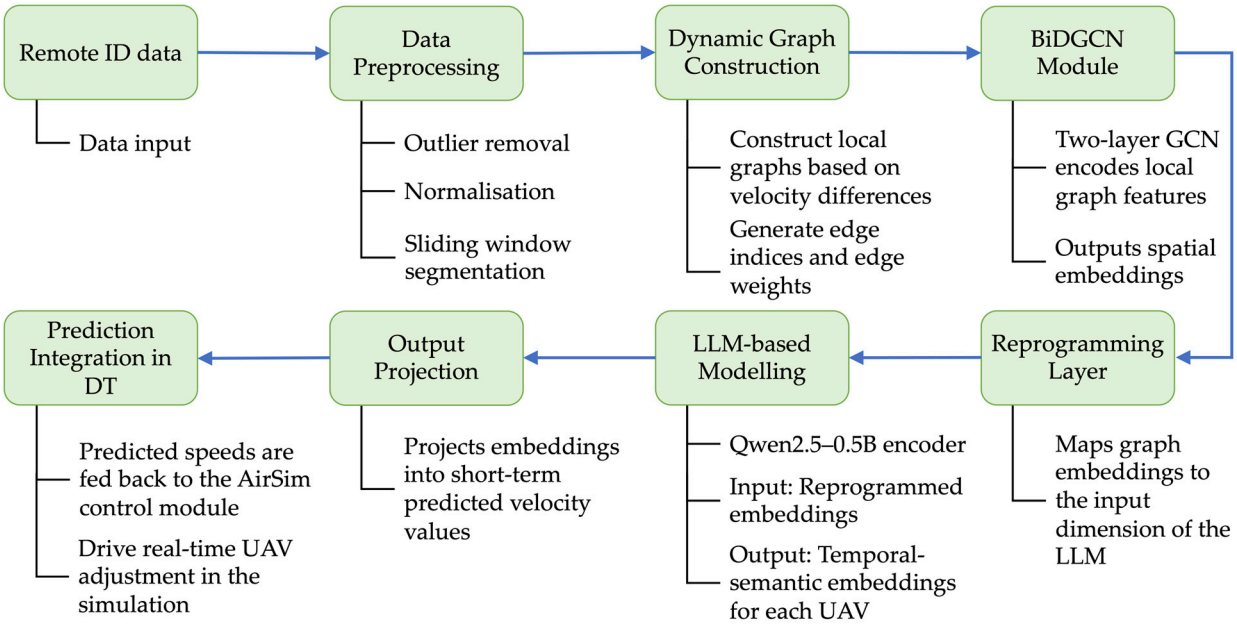


Figure 4. Operation framework and data flow.

The model architecture comprises these key components: Input Layer, Dynamic Graph Constructor, GCN Encoder, Reprogramming Layer, LLM-based Semantic Encoder, and Output Projection.

2.2.2. Data Preprocessing and Dynamic Graph Construction

To enhance the capability of the GNN in modelling dynamic interactions among drones, we design a dynamic graph construction mechanism based on historical velocity differences. This mechanism integrates optimisation preprocessing, sliding-window sampling, and a structure auto-generation strategy to enable end-to-end construction of input graph structures.

Assume the raw UAV velocity data is represented as a two-dimensional time-series matrix $V \in \mathbb{R}^{\mathrm{T_{total}} \times \mathrm{N}}$.

The T_{total} denotes the total number of global time steps, which refer to the discrete, uniformly sampled temporal points covering the entire dataset duration. Each global time step corresponds to one timestamp in the overall velocity time series, which is collected at fixed intervals, and N represents the number of drones. Each training sample corresponds to a velocity segment of all drones over a window of T time steps. After applying sliding-window segmentation to the whole sequence, the processed input forms a tensor: $X \in \mathbb{R}^{B \times N \times T}$.

A local graph structure is constructed within each sample to model the inter-UAV influence relationships under structured flight conditions, such as a priori defined fixed air corridors or formation flying, thereby capturing the spatial connectivity patterns among UAV nodes. Specifically, let the node set for each sample be denoted as $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$.

Each node in the graph represents a UAV, with node indices assigned according to their order in the air route or formation sequence. The edge set $\mathcal{E}$ is constructed using a fixed bidirectional chain topology, where each UAV node v_i is connected bidirectionally to its adjacent drones v_{i-1}, v_{i+1}, forming:

$$\mathcal{E} = \{(i, i+1), (i+1, i) \mid i = 1, 2, \ldots, N-1\} \tag{1}$$

Building upon this structure, to enhance the capacity of the graph to represent motion patterns among UAVs, we introduce an edge-weighting mechanism based on velocity differences. Let the historical velocity sequences of UAVs i and j over a time window T be denoted as $v_i = \left[v_i^1, \ldots, v_i^T\right]$ and $v_j = \left[v_j^1, \ldots, v_j^T\right]$, and the weight w_{ij} of the edge connecting UAVs i and j is then defined as:

$$w_{ij} = \frac{1}{\frac{1}{T}\sum_{t=1}^{T}\left|v_i^t - v_j^t\right| + \epsilon} \tag{2}$$

The ϵ is a small positive constant introduced to prevent division-by-zero errors. This edge weight definition ensures that UAVs exhibiting similar velocity trends are assigned stronger connection strengths, which facilitates more accurate modelling of potential cooperative behaviour during message propagation in the GNN.

The constructed graph structure is encoded in a sparse matrix format to support efficient computation within the GNN framework. Specifically, the edge connectivity is represented using an index matrix as follows:

$$edge_index \in \mathbb{R}^{2 \times E} \tag{3}$$

The corresponding edge weight are stored in a vector format:

$$edge_weight \in \mathbb{R}^E \tag{4}$$

The E denotes the total number of edges in the current sample. Each column in the edge_index matrix specifies the source and target node indices for one edge, while the corresponding edge_weight vector stores the Dynamic Edge Weight. This sparse representation significantly reduces memory overhead and computational redundancy compared to dense adjacency matrices.

If an explicit graph structure is not provided during model training and inference, the system will automatically invoke the graph construction function create_vehicle_graph(). This function dynamically generates the edge index and weight tensors based on the current batch of velocity inputs, enabling end-to-end adaptive integration of graph structures into the learning pipeline.

2.2.3. GCN Encoder

GCN is a deep learning architecture capable of modelling feature dependencies on non-Euclidean structured data, making it well-suited for graph-based modelling of UAV interactions. Considering the characteristics of UAVs operating along fixed air routes, such as localised interactions and strong sequential dependencies, we design a velocity-aware, two-layer GCN encoder to extract spatial interaction features among UAVs.

To capture spatial dependencies, we construct a two-layer GCN Encoder built upon the generated graph structure. Each node in the graph represents a UAV, and edges are built using a fixed bidirectional chain topology, where each UAV is connected to its immediate neighbours. The edge weights are computed from the average absolute velocity difference over a time window, emphasising stronger connections between UAVs with similar movement patterns. This encoder is implemented using the GCNConv module from PyTorch Geometric, which is based on PyTorch 2.2.2. It takes both the node-level time-series features and graph connectivity information as joint input. The output consists of spatial embedding vectors for each UAV node, which encode local neighbourhood interactions conditioned on Dynamic Edge Weight.

$$\mathcal{G} = (\mathcal{V}, \mathcal{E}) \tag{5}$$

$$X \in \mathbb{R}^{N \times T} \tag{6}$$

The N denotes the number of UAV nodes, and T represent the number of historical time steps of speed for each node. The goal of graph convolution is to propagate and aggregate features from neighbouring nodes via the graph adjacency structure, thereby producing embedded representations that capture local interaction patterns. The BiGCN consists of two GCN layers with a ReLU activation function.

The hierarchical structure of the GCN encoder is defined as follows:

- Layer 1: $\mathrm{GCNConv}(\mathrm{T} \rightarrow \mathrm{d_{hidden}}) + \mathrm{ReLU}$
- Layer 2: $\mathrm{GCNConv}(\mathrm{d_{hidden}} \rightarrow \mathrm{d_{model}})$

This study adopts the standard GCN convolutional formulation proposed by Kipf, while incorporating edge weight into the message passing process. At each layer l, node features are updated through the standard weighted GCN propagation rule:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)}\right) \tag{7}$$

2.2.4. Reprogramming Layer

To bridge the semantic and dimensional gap between the output of the GNN and the input requirements of the LLM, we design a lightweight and functionally explicit reprogramming layer. Positioned between the GCN encoder and the LLM, this module is responsible for mapping the spatial embeddings produced by the GCN into the semantic input space expected by the LLM, thereby enabling cross-modal transformation from graph-based representations to language-model-compatible sequences. The output feature tensor from the GCN encoder is:

$$H_{GCN} \in \mathbb{R}^{B \times N \times d_{model}} \tag{8}$$

The B denotes the batch size, N the number of UAVs, and d_{model} the output feature dimension of the GCN encoder. The reprogramming layer first applies a dimensionality up-projection to map the input tensor into a higher-dimensional hidden space, followed by an activation function (ReLU), and then a dimensionality down-projection to match the embedding size required by the language model. The final output is a tensor $Z \in \mathbb{R}^{B \times N \times d_{\text{model}}}$ that aligns with the input specification of the LLM. The entire transformation process can be expressed as:

$$\mathrm{Z} = \mathrm{LayerNorm}\left(\mathrm{W}_{\text{down}} \cdot \mathrm{ReLU}\left(\mathrm{W}_{\text{up}} \cdot \mathrm{H}_{\text{GCN}}\right)\right) + \mathrm{R} \tag{9}$$

The $\mathrm{W}_{\text{up}} \in \mathbb{R}^{d_{\text{model}} \times d_{\text{inter}}}$ denotes the up-projection matrix, and $\mathrm{W}_{\text{down}} \in \mathbb{R}^{d_{\text{inter}} \times d_{\text{LLM}}}$ represents the down-projection matrix used to align with the input dimension of LLM. An optional residual connection R is applied when the input and output dimensions match, facilitating better gradient flow and representation retention.

Under the default parameter configuration, this module uses $\mathrm{d}_{\text{model}} = 256$, $\mathrm{d}_{\text{inter}} = 512$, $\mathrm{d}_{\text{LLM}} = 896$, where $\mathrm{d}_{\text{model}}$ denotes the output feature dimension of the GCN encoder, $\mathrm{d}_{\text{inter}}$ is the intermediate dimensionality of the transformation layer, and d_{LLM} matches the embedding size required by the Qwen2.5–0.5B language model. This design ensures a smooth transition from spatial graph embeddings to temporal semantic representations in the LLM.

Functioning as an interface between the GCN and LLM representation domains, this module transforms graph-based features into a semantic embedding space. This transformation enables the structured input to be interpretable and processable within the language modelling framework. The module is designed with nonlinear activation functions and residual pathways, which improve the capacity for semantic representation. Furthermore, the structure of the module has been formulated with high generalizability, allowing it to be seamlessly integrated with a wide range of GCN and LLM configurations.

In multimodal fusion tasks, the reprogramming layer serves as a stable, efficient, and interpretable cross-modal interface, pivotally enabling graph–semantic collaboration.

As a core component of the proposed BiDGCNLLM architecture, this module embodies the key idea of constructing a structured transformation path between graph and semantic modalities. By providing a unified representational foundation for the subsequent LLM module, it significantly enhances the overall expressive power and generalisation capability of the system.

2.2.5. Large Language Model

This paper introduces an LLM as a temporal modelling module to further improve the modelling ability of the temporal change law of drone speed. Different from traditional sequence modelling methods (such as RNN and Transformer), LLM has stronger context understanding and dynamic semantic construction abilities and can effectively capture the complex temporal dependency structure between multiple nodes.

In terms of model structure, this paper connects the encoder part of a multi-layer pre-trained language model after the GCN encoder and reprogramming layer, and the selected model is Qwen2.5–0.5B. The encoder part of the Qwen2.5–0.5B model is frozen, i.e., its parameters are not updated during training. This design choice preserves the pretrained temporal reasoning capabilities of the LLM while reducing computational cost and preventing overfitting on limited UAV datasets.

In implementation, we use inputs_embeds as the input interface, that is, directly input the semantic embedding $Z \in \mathbb{R}^{B\times N\times d_{\text{model}}}$ of the reprogramming output into the language model, skip the word embedding layer, and retain the self-attention structure inside the model for feature modelling, allowing direct input of graph-derived semantic embeddings. The input of LLM is:

$$Z = [z_1, z_2, \ldots, z_N],\ \ z_i \in \mathbb{R}^{d_{LLM}} \tag{10}$$

The encoding process of the model can be expressed as:

$$\mathrm{H}^{(l+1)} = \mathrm{TransformerBlock}^{(l)}\left(\mathrm{H}^{(l)}\right), \mathrm{H}^{(0)} = \mathrm{Z} \tag{11}$$

$$\mathrm{H}_{\mathrm{LLM}} = \mathrm{H}^{(\mathrm{L})} \in \mathbb{R}^{\mathrm{B}\times\mathrm{N}\times\mathrm{d}_{\mathrm{LLM}}} \tag{12}$$

The L denote the number of encoding layers in the LLM, and $\mathrm{H}^{(l)}$ represent the intermediate hidden state at the number l layer.

Each Transformer block consists of a multi-head self-attention module followed by a feedforward neural network. The core of this structure lies in the self-attention mechanism, which is defined as follows:

$$\mathrm{Attention}(\mathrm{Q},\mathrm{K},\mathrm{V}) = \mathrm{softmax}\left(\frac{\mathrm{QK}^{\mathrm{T}}}{\sqrt{\mathrm{d}}}\right)\mathrm{V} \tag{13}$$

Furthermore:

- Query matrix $Q = ZW^{Q}$, key $\mathrm{K} = \mathrm{ZW}^{\mathrm{K}}$, value $V = ZW^{V}, \mathrm{W}^{\mathrm{Q}}, \mathrm{W}^{\mathrm{K}}, \mathrm{W}^{\mathrm{V}} \in \mathbb{R}^{d_{LLM}\times d_{LLM}}$
- The multi-head mechanism allows the model-to-model temporal dependencies in parallel in different subspaces

Finally, the output LLM representation tensor is $\mathrm{H}_{\mathrm{LLM}} \in \mathbb{R}^{B\times N\times d_{\mathrm{LLM}}}$, each row of which represents the speed semantic representation of a certain drone combined with the global context at the current time step.

This module supports freezing or fine-tuning pre-training parameters during implementation, which can be flexibly selected according to the amount of data and generalisation requirements. In the experimental setting of this article, we use the encoding part of the Qwen2.5–0.5B model for fine-tuning. It is good that context modelling ability significantly enhances the understanding of the trend of drone time series changes in the model.

The pre-trained LLM is used to model the graph node sequence in time series, which overcomes the problem of insufficient long-range dependency capture in traditional time modelling methods while retaining the potential for the fusion of graph structure and language semantics. It also constructs a cross-modal semantic modelling capability suitable for multi-drone speed prediction tasks.

2.2.6. Output Projection

After the LLM completes the high-dimensional encoding of node-level temporal features, the model must transform its output into concrete prediction results. To achieve this, an output projection module is designed, consisting of a flattening operation followed

by a linear mapping, which converts the multi-dimensional embeddings generated by the LLM into the target velocity values.

The semantic representation of each UAV node is initially reshaped to maintain structural alignment along the UAV dimension while reducing the output dimensionality of the LLM. A trainable linear projection layer is then applied to map the resulting flattened vector into the prediction target space, representing the one-step UAV speed.

This linear layer performs an affine transformation, optimised by a weight matrix and a bias term, both of which are jointly optimised during training. Despite its structural simplicity, the layer possesses strong representational capacity and effectively utilises contextual information embedded in the language output of the model, thereby enabling precise mapping from the semantic space to the target value space. The final output is a velocity prediction tensor $\hat{y} \in \mathbb{R}^{B\times N\times 1}$, indicating the predicted speed for each UAV in each sample.

The module is designed to be lightweight and decoupled, offering high generalisability and scalability. In single-step prediction tasks, it efficiently maps LLM-derived embeddings to continuous outputs. In multi-step forecasting scenarios, the structure can be readily extended into a sequence output projection module to accommodate prediction requirements of varying lengths. With this, the model completes the whole pipeline from structured time series input to velocity prediction output, achieving an integrated design that unifies spatial perception, semantic enhancement, and task-level closure.

3. Experimental Results

3.1. Data and Preprocessing

In this study, the data collection process is primarily used to support the BiDGCNLLM in accurately predicting drone speeds. By continuously collecting historical data over the past T seconds, an adjacency matrix that reflects the relationships between leading and following drones is constructed and fed into the BiDGCN module to extract global spatial features. These features are then stacked to form a time series input, which is processed by the LLM module for temporal modelling and speed prediction. This data collection strategy not only ensures that the model fully captures spatial interactions among drones but also provides a stable input foundation for time series modelling, thereby enhancing prediction accuracy.

To ensure data integrity, we perform linear interpolation and outlier removal on the speed, acceleration, and position information of each drone. This process ensures that at any time, all drone observations exist simultaneously to avoid vacancies in the time dimension. The cleaned data is organised into a tensor $X \in \mathbb{R}^{B\times N\times T}$. To eliminate the differences in observation scales between different drones and improve the convergence and stability of model training, the global mean and standard deviation of the training set are used to transform each feature dimension, and all input features are uniformly normalised during the training phase.

To evaluate the performance of the proposed BiDGCNLLM on the UAV speed prediction task, four widely used regression metrics are adopted: Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R^2).

Furthermore, given the model predictions $\hat{y}_t$ and the corresponding ground truth values y_t over T prediction time steps, the evaluation metrics are formally defined as follows:

$$\mathrm{MAE} = \frac{1}{T}\sum_{t=1}^{T}|y_t - \hat{y}_t| \tag{14}$$

$$\mathrm{MSE} = \frac{1}{T}\sum_{t=1}^{T}(y_t - \hat{y}_t)^2 \tag{15}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_t - \hat{y}_t)^2} \tag{16}$$

$$R^2 = 1 - \frac{\sum_{t=1}^{T}(y_t - \hat{y}_t)^2}{\sum_{t=1}^{T}(y_t - \overline{y})^2} \tag{17}$$

3.2. Experimental Setup and Model Training

All experiments were trained using two NVIDIA A100 GPUs and an NVIDIA RTX 3080 Ti GPU (NVIDIA, Santa Clara, CA, USA). In BiDGCNLLM training, Adam was used as the optimiser and MSE Loss was used as the loss function. In addition, the MAE was defined for evaluation indicators, but it was only used for performance evaluation in the validation phase and did not participate in back-propagation optimisation.

In this experiment, we collected dynamic operation data from 20 drones, including Remote ID data. Each vehicle generated 7000 sets of acceleration, speed, and other information, forming a comprehensive dataset. This dataset is used to support the training and evaluation of the BiDGCNLLM, of which 70% of the data is used for model training, 15% for testing, and the remaining 15% for verification.

We compared the proposed BiDGCNLLM with several advanced algorithms and models in the field of time series prediction to comprehensively evaluate its performance in the multi-UAV speed prediction task. The selected baseline models include representative methods with different structural types and modelling mechanisms, aiming to analyse performance from multiple dimensions such as graph structure modelling, time dependency modelling, and language modelling.

This study selected several representative comparison models in the experiment, covering different types of time series modelling methods, including RNN, LSTM, and BiLSTM, to evaluate the traditional time series modelling capabilities; FourierGNN is used to verify the performance of frequency domain graph structure modelling in prediction tasks; TimeMixer and PatchTST represent the efficient Transformer structures proposed in recent years, with strong time-dependent modelling capabilities; TimeLLM introduces a pre-trained LLM to enhance the expression of time features through context understanding capabilities.

The above comparison models cover the three major mainstream directions of LLM, GNN, and convolutional modelling in current time series prediction. They provide a multidimensional baseline for the performance evaluation of BiDGCNLLM, which helps to highlight its advantages in integrating graph structure dynamic modelling and language representation capabilities.

3.3. Experimental Prediction

In the context of UAM, drones in the air corridor need to make decisions quickly and in real time within a short time frame. In such a dense and dynamic environment, long-term predictions are often inaccurate due to frequent changes in traffic patterns and external interference. Therefore, short-term predictions are more in line with the temporal resolution and responsiveness requirements of the system. In addition, inference speed is critical. Short-term predictions require fewer computing resources, thereby reducing system latency, which is critical to ensuring timely conflict resolution and adjustment of flight paths. Therefore, in this study, we set the prediction horizon to one time step to simulate the speed adjustment scenario at critical moments in a multi-unmanned system. To provide sufficient historical context information, we set the input sequence length to

32, allowing the model to capture more accurate trajectory dynamics and speed evolution patterns between the front and rear targets.

The model adopts the BiDGCNLLM structure based on the fusion of graph Convolution and LLM. The GCN is responsible for extracting the dynamic interaction relationship between drones, and the LLM module models the time dependency to achieve high-precision speed prediction. During the training process, we used a feature encoding layer (d_model) with a dimension of 256 and a feedforward network dimension (d_ff) of 512 to maintain a balance between model capacity and training stability; the learning rate was set to 0.00008.

The prediction results are shown in Figure 5. The BiDGCNLLM exhibits excellent predictive performance on the UAV speed forecasting task, with predicted values nearly overlapping the ground truth. To quantify prediction accuracy, four statistical metrics are employed: MSE, RMSE, MAE, and R^2. The model achieves an MSE of 0.0075, RMSE of 0.0867, MAE of 0.0251, and an R^2 score of 0.9992. These indicators reflect low prediction deviation, high fitting precision, and strong linear agreement between predicted and true values. The results confirm that the model not only captures dynamic speed variations between leading and following drones within an air corridor but also ensures high stability and accuracy in short-term forecasting.

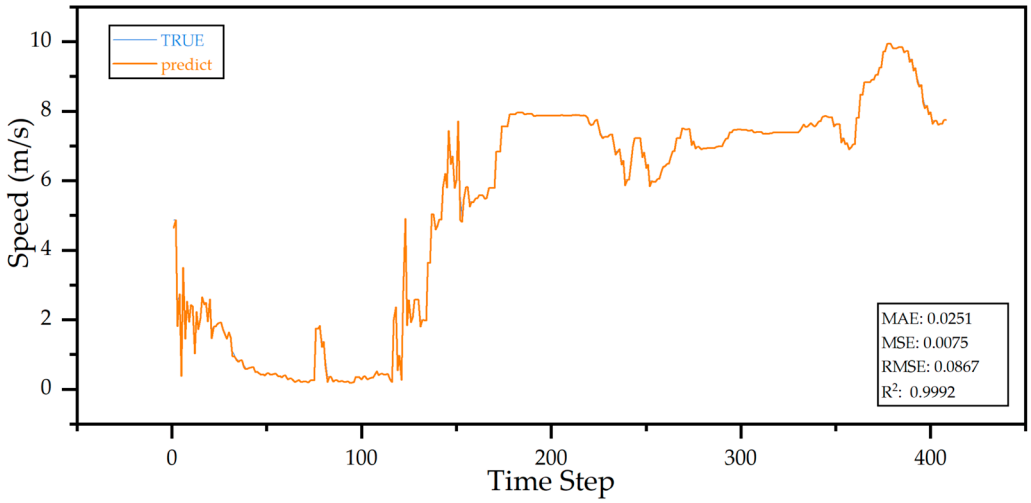


Figure 5. Comparison of true values and prediction values.

3.4. Ablation Study

To further verify the effectiveness of each key module in the proposed BiDGCNLLM framework, this study designed multiple groups of ablation experiments. By constructing several model variants and selectively adding or removing specific components, the contribution of each module to the overall performance was evaluated. Specifically, the following four settings are included:

- Baseline: GCN and LLM are combined as the basic comparison model to verify the effect of the collaborative modelling of the two.
- Baseline + BiGCN: BiGCN is introduced based on Baseline to explore the effect of forward and backwards graph information transmission on prediction performance.
- Baseline + Dynamic Edge Weight: The Dynamic Edge Weight mechanism is introduced on Baseline to model the correlation strength between drones as it evolves over time and evaluate the effect of Dynamic Edge Weight adjustment.

- Full model (BiDGCNLLM): A complete prediction framework that integrates BiGCN, Dynamic Edge Weight, and LLM, representing the final model structure proposed in this paper.

Through the above comparative experiments, the impact of each key module on the final prediction performance can be systematically analysed, thereby verifying the structural rationality and effectiveness of BiDGCNLLM.

As shown in Table 1, each module of BiDGCNLLM contributes to performance improvement. The baseline (GCNLLM) achieves an MAE of 0.0235 and RMSE of 0.0747. Adding a BiGCN (Baseline + BiGCN) slightly increases MAE but reduces RMSE, enhancing stability. Introducing Dynamic Edge Weight (Baseline + Dynamic Edge Weight) further improves all metrics, with MAE reduced to 0.0205 and R^2 reaching 0.9982. The full model (BiDGCNLLM) achieves the best performance (MAE 0.0251, RMSE 0.0867, R^2 0.9992), demonstrating the effectiveness of combining bidirectional spatial encoding and dynamic relational reasoning within an LLM framework.

Table 1. Ablation analysis of different models with the same dataset.

Method	MAE	MSE	RMSE	R^2
Baseline (GCNLLM)	0.0235	0.0056	0.0747	0.9965
Baseline + BiGCN	0.0259	0.0040	0.0636	0.9975
Baseline + Dynamic Edge Weight	0.0205	0.0029	0.0537	0.9982
Fullmodel (BiDGCNLLM)	0.0251	0.0075	0.0867	0.9992

3.5. Comparative Analysis

In order to verify the performance advantages of the proposed model in time series prediction tasks, this paper selected several representative baseline models for comparative experiments, covering different modelling paradigms such as traditional Recurrent Neural Network (RNN), GNN, Transformer variants, and LLM fusion methods.

In terms of classic sequence modelling, three recurrent neural network structures, RNN, LSTM, and BiLSTM, were selected. RNN, as the earliest infrastructure used for time series tasks, serves as a traditional benchmark reference. LSTM significantly alleviates the long-term dependency problem by introducing a gating mechanism, and BiLSTM further enhances the modelling ability of the model for bidirectional time information, allowing for more comprehensive capture of contextual features.

In the direction of GNN, the FourierGNN model is introduced as a representative of combining frequency domain analysis with graph structure modelling. This model enhances the perception of time series patterns through Fourier transforms, improving spectrum modelling performance while maintaining the expression of graph structure.

In response to the new paradigm of time series modelling proposed in recent years, this paper incorporates two high-performance transformer-based architectures, TimeMixer and PatchTST. TimeMixer effectively achieves sequence modelling with low computational overhead by mixing channel and time dimension information; PatchTST uses sliding patches to embed time series as fixed-dimensional inputs, and uses the standard Transformer mechanism to capture temporal dependencies, showing strong generalisation capabilities.

TimeLLM was chosen for comparison, as it integrates a pre-trained LLM with time series modelling to enhance temporal feature representation and exhibits strong cross-task generalisation. This comparison highlights the strengths of the proposed method across diverse modelling paradigms.

The comparative results in Table 2 and Figure 6 demonstrate that BiDGCNLLM offers a strong overall advantage in UAV speed prediction for structured urban airspace. While TimeMixer marginally outperforms it in MAE and by PatchTST in MSE and RMSE,

BiDGCNLLM achieves the highest R^2 score of 0.9992, indicating a near-perfect fit and superior generalisation capability. This suggests that the model excels in capturing global spatio-temporal trends while maintaining competitive local accuracy. Its effectiveness stems from the integration of bidirectional dynamic graph convolutions, which model spatial dependencies among UAVs, and the LLM that captures temporal dynamics. Together, these components form a unified framework capable of learning complex inter-agent interactions and temporal evolution, making BiDGCNLLM an interpretable solution for DT-driven UAM applications, particularly in scenarios requiring high predictive stability and real-time coordination.

Table 2. Comparative analysis of BiDGCNLLM with other methods under the same dataset.

Method Name	MAE	MSE	RMSE	R^2	Ref.
RNN	0.0429	0.0319	0.1786	0.9757	[37]
FourierGNN	0.0332	0.0125	0.1119	0.9904	[38]
LSTM	0.0312	0.0156	0.1250	0.9881	[39]
BiLSTM	0.0209	0.0072	0.0846	0.9945	[40]
TCN-Transformer	0.1057	0.1111	0.3333	0.9151	[41]
Transformer-LSTM	0.0871	0.1373	0.3706	0.8954	[42]
TimeMixer	0.0195	0.0056	0.0750	0.9957	[43]
TimesNet	0.1072	0.1119	0.3346	0.9144	[44]
PatchTST	0.0239	0.0048	0.0695	0.9963	[27]
TimeLLM	0.0404	0.0087	0.0933	0.9947	[30]
BiDGCNLLM	0.0251	0.0075	0.0867	0.9992	/

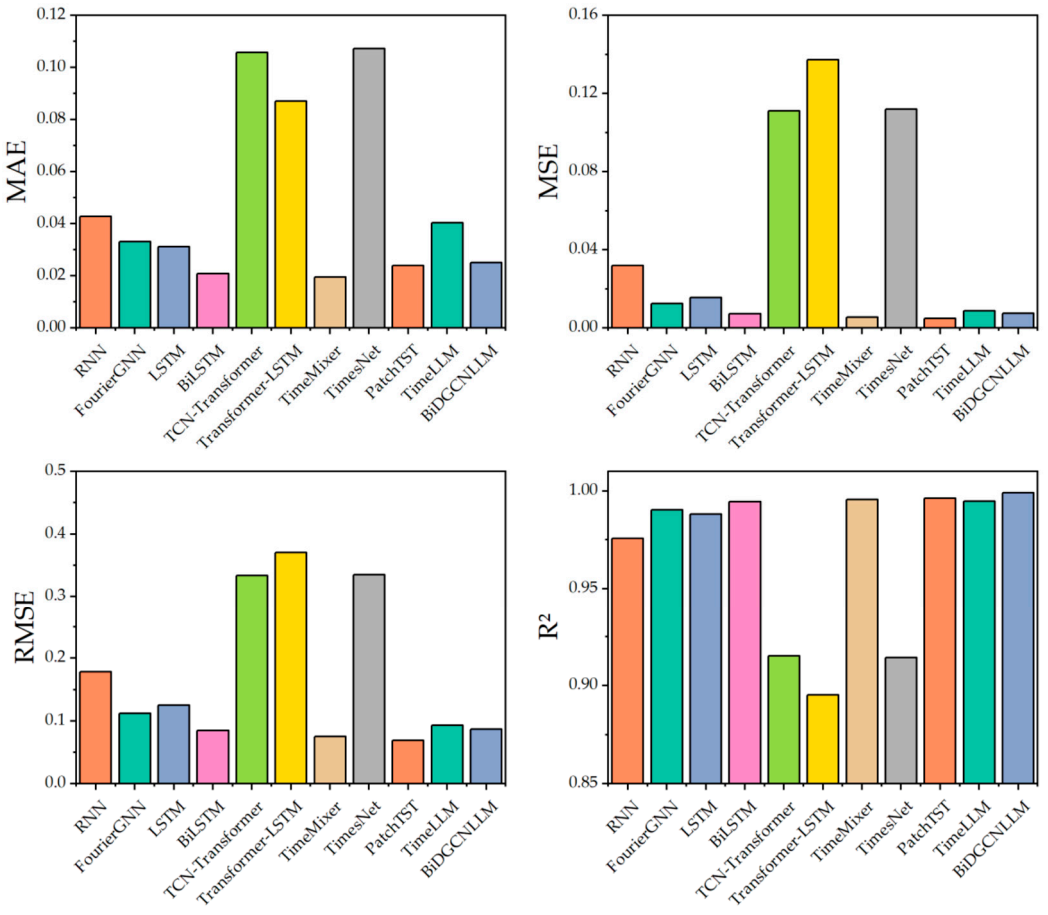


Figure 6. Bar chart of BiDGCNLLM comparative analysis.

3.6. Scaling Law Analysis

To evaluate the scalability and efficiency of the proposed BiDGCNLLM, a scaling law analysis was conducted by examining how performance metrics vary with model size. Specifically, the relationship between the number of model parameters and both prediction accuracy and inference latency was studied.

The results demonstrate a clear correlation between model size and accuracy. As shown in Figure 7a, the MSE exhibits a downward trend as model size increases. When both axes are plotted on a logarithmic scale, the fitted regression line indicates a slope of approximately -0.45, suggesting that larger models generally achieve lower prediction errors. This is consistent with empirical scaling law behaviour observed in other deep learning tasks. The MSE decreases notably with models exceeding one million parameters, with the largest model yielding the highest predictive accuracy. Although minor fluctuations exist among smaller models, they are likely attributable to limited expressive capacity or convergence instability during training.

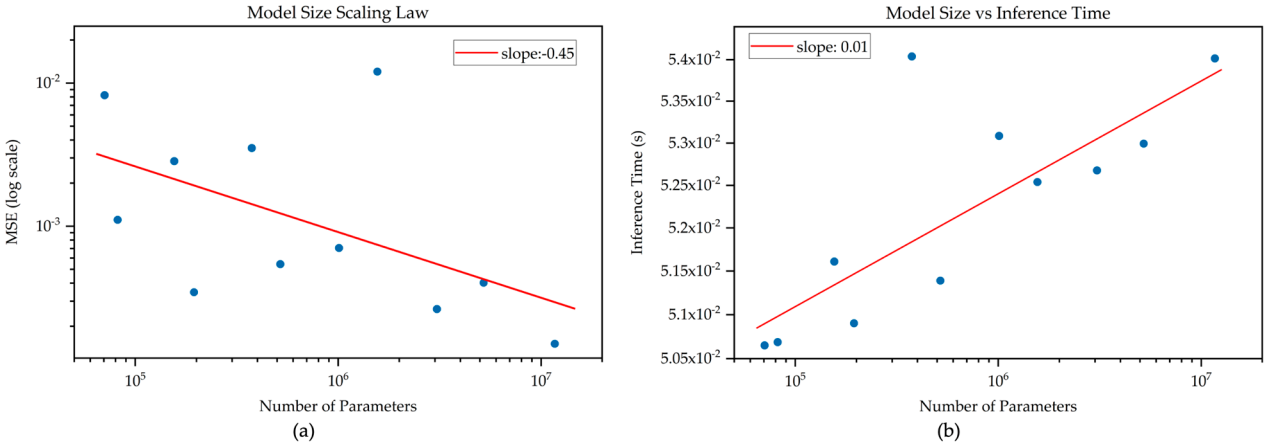


Figure 7. (**a**) Scaling law; (**b**) Model size vs inference time.

In terms of inference efficiency, Figure 7b presents the observed relationship between model size and latency. The fitted line indicates a modest slope of 0.01, showing that inference time increases only slightly as the number of parameters grows. Despite a two-order magnitude expansion in model size, the actual variation in inference time remains within a narrow range (from approximately 0.050 to 0.054 s). This confirms the computational efficiency of the architecture, which benefits from a lightweight GCN encoder and an optimised reprogramming layer that bridges graph embeddings and the LLM input space with minimal overhead.

The combination of accuracy improvement and controlled inference cost implies that the BiDGCNLLM architecture maintains high performance without compromising real-time applicability. These scaling properties make it well-suited for time-sensitive UAM scenarios, where predictive robustness and low-latency response are both essential for ensuring safety and coordination. The results also support future exploration of LLM and graph encoders to further enhance prediction stability, particularly in denser or more dynamic airspace environments.

4. BiDGCNLLM Integration and Digital Twin Visualisation Simulation

4.1. BiDGCNLLM and Digital Twin Integration

Building upon the previously developed DT platform by the authors, this study constructs a multi-UAV simulation and visual verification system tailored for UAM and low-altitude air corridors [45]. The system aims to systematically evaluate the comprehensive

performance of the BiDGCNLLM algorithm in speed prediction. The integrated system framework is illustrated in Figure 8. This DT platform not only supports three-dimensional urban airspace modelling and simulation but also incorporates an algorithm-driven task scheduling and data feedback mechanism, enabling a tight integration between AI models and multi-source simulation modules.

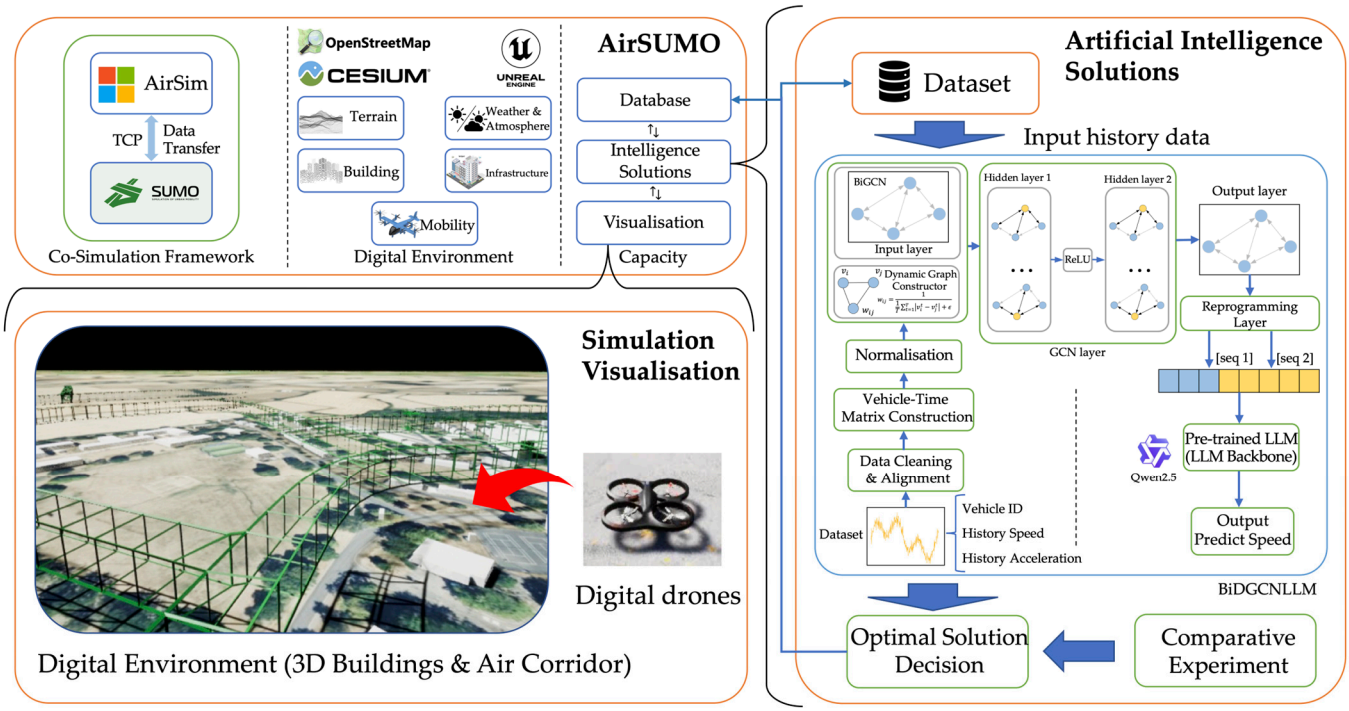


Figure 8. Research architecture, including BiDGCNLLM and DT evaluation.

From a system architecture perspective, the BiDGCNLLM first predicts UAV speeds at future time steps based on historical trajectory graphs and spatio-temporal dependency features. The outputs of the model are then transmitted via a unified data interface to the DT module, where they serve as control-layer inputs to dynamically adjust the speed of each UAV. This process ensures that the prediction results directly influence UAV behaviour in the simulation system, thereby achieving a closed-loop mapping from GNN inference to physical environment response.

4.2. Digital Twin Construction and Deployment

To construct the static layer of the digital environment, the system incorporates the Cesium plugin and the OpenStreetMap plugin, which enable high-precision terrain rendering and visualisation of urban geospatial data integrated with Unreal Engine [46,47]. For the dynamic layer, the open-source and cross-platform UAV simulator AirSim, developed by Microsoft Research, was integrated to simulate multirotor UAV operations and task execution within the environment [48,49]. The AirSim API provides access to the simulated states of the UAS, including position, velocity, and orientation, thereby supporting interactive algorithm testing within the DT framework.

To support speed prediction and operational stability verification for multi-UAV operations along fixed routes, a co-simulation framework, AirSUMO, has been developed, as shown in Figure 9. The figure illustrates the system architecture of the DT simulation platform developed in this study. OpenStreetMap provides urban terrain and road data, which are imported into Unreal Engine via the StreetMap plugin for terrain modelling, or into Blender through Blender-GIS for 3D air corridor modelling. The resulting models are

exported in .fbx format and integrated into the virtual environment. The Cesium plugin enhances the rendering accuracy of global terrain. SUMO transforms OSM data into a simulation-ready transport network using netconvert and connects to AirSim via TCP to support real-time UAV trajectory planning and flight behaviour simulation. Unreal Engine serves as the core visualisation platform, integrating all modules to construct a high-fidelity virtual urban airspace environment that supports the deployment and validation of the BiDGCNLLM prediction model.

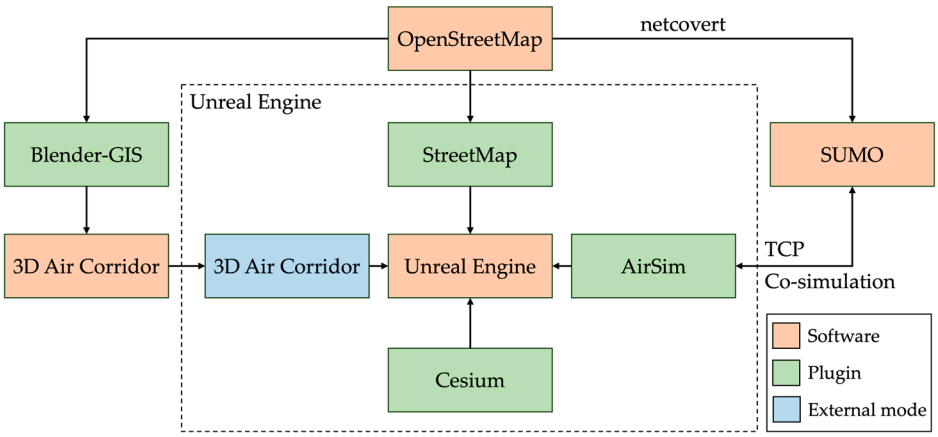


Figure 9. The software architecture of AirSUMO [45].

This framework integrates SUMO and AirSim using the TraCI interface to achieve real-time data exchange: SUMO simulates speed, acceleration, and trajectory behaviours along fixed air routes, while AirSim provides flight dynamics simulation and 3D visualisation. The integrated platform enables real-time updates of individual UAV speeds, allowing researchers to observe flight trajectories from multiple perspectives and compare them against predefined air corridor structures. This facilitates the detection of anomalies such as yaw deviations, drift, or formation breakup, thereby quantitatively validating the prediction stability and engineering applicability of LLM.

The DT environment is geographically based on Cranfield University and its affiliated airport in the UK, selected to ensure alignment with real-world UAM operational settings and regulatory constraints. This high-fidelity virtual environment replicates key infrastructure elements, terrain features, and spatial layouts relevant to typical low-altitude airspace applications. Within this environment, a representative UAM flight mission was carefully designed and deployed, simulating a point-to-point logistics or inspection task. The mission begins at the Cranfield Management Development Centre and terminates at Building 316 (Convey House), tracing a structured route through designated urban air corridors. The specific flight path, which includes several turning points and altitude-holding segments to reflect realistic UAV manoeuvring requirements, is illustrated in Figure 10.

During the flight mission, UAV velocity data were collected using AirSim APIs. The recorded data include velocity components in the x, y, and z directions. Given that the study focuses on UAV operations within air corridors parallel to the ground plane, only the lateral velocity components (i.e., x and y directions) are analysed in this evaluation, while the *z*-axis velocity is excluded. This exclusion simplifies the evaluation process and ensures alignment with the operational assumptions of level-flight scenarios common in low-altitude UAM frameworks.



Figure 10. Multi-drone flight route in Cranfield UAS air corridor.

4.3. Digital Twin Operation Simulation and Optimisation

To validate the effectiveness of BiDGCLLM within a co-simulation DT environment, a fixed-corridor three-independent-drones operation experiment was conducted involving three UAVs arranged in a front–mid–end configuration. Speed tracking was achieved through API-based real-time control. The deployment details, including UAV identifiers and initial positions, are summarised in Table 3.

Table 3. The number and position of drones.

Number	AirSim ID	Position
1	Drones 1	Front (Leader)
2	Drones 2	Middle
3	Drones 3	End

The velocity data processing flow of the drones is shown in Figure 11. By collecting the velocity data of the three drones along the x, y, and z axes, and considering that drones operate within horizontal air corridors in the given context, longitudinal (vertical) speed variations can be neglected. It is noted that the velocity data used for model evaluation were collected from a co-simulation environment based on AirSUMO, where three UAVs operate independently in a structured DT air corridor, and their lateral velocities were calculated from simulated x and y components via API data. Therefore, the horizontal velocity of each UAV is calculated as:

$$v_{Lateral} = \sqrt{{v_x}^2 + {v_y}^2} \tag{18}$$

The flight simulation of drones in the air corridor of the AirSUMO DT environment is shown in Figure 12. To verify the optimisation effect of the LLM, the original flight data of Drones 1, Drones 2, and Drones 3, the predicted data of the BiDGCNLLM, and the DT simulation test data are compared to systematically evaluate the prediction accuracy and actual deployment effect of the model. Among them, the original data comes from the real flight trajectory and represents the flight performance under the unoptimised state. The predicted data is output by BiDGCNLLM under the state of the known front drone, reflecting the ability of the model in multi-drone speed prediction and coordination.

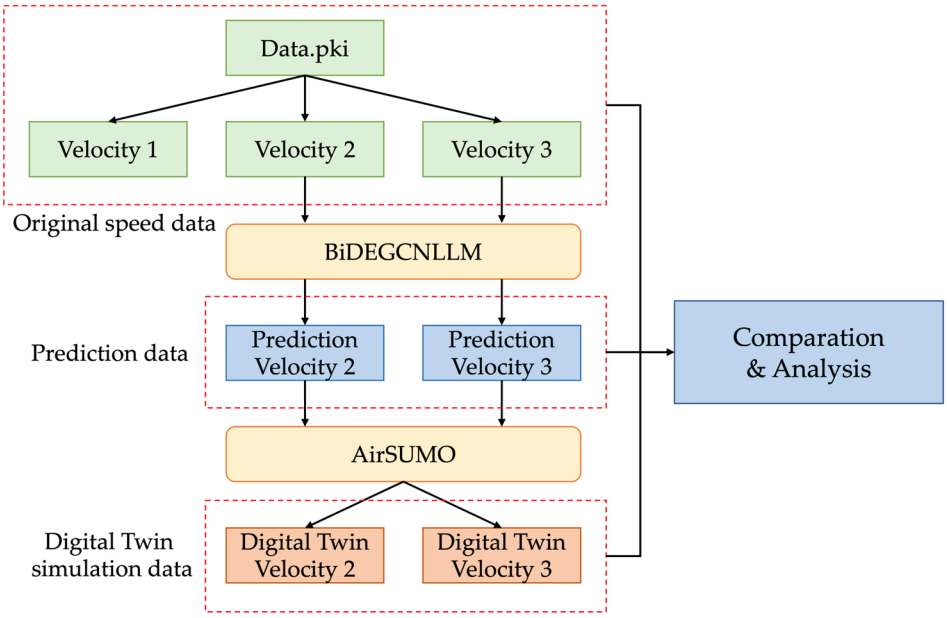


Figure 11. Flow chart of data processing and analysis.

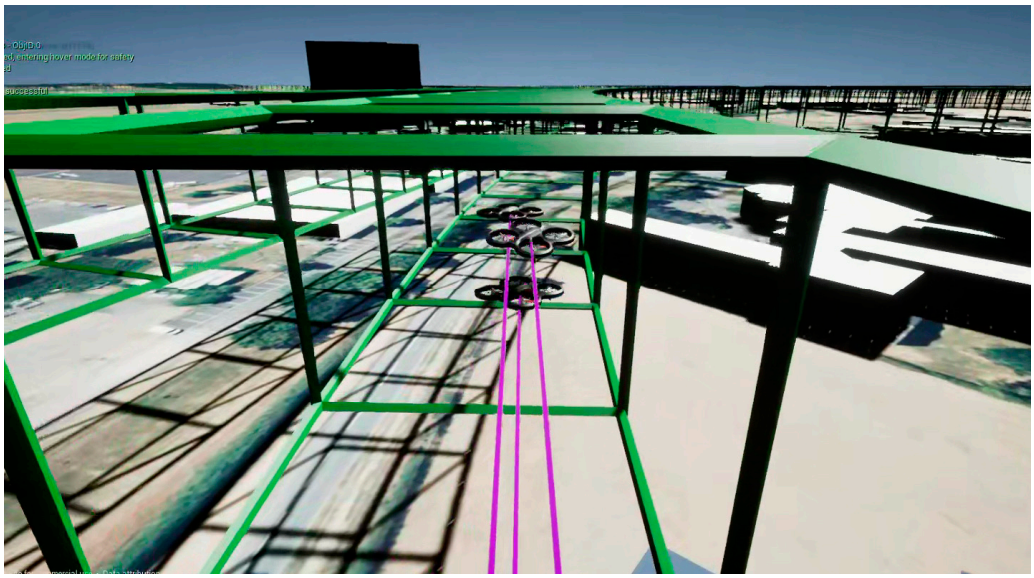


Figure 12. Operation of drones simulation in AirSUMO DT environment.

Based on Equation (18), the UAV flight speeds were computed. The velocity data were collected via APIs in the PKI format and processed to extract the lateral velocity components of Drones 1, 2, and 3, as shown in Figure 13. The comparison indicates that when the UAVs operate independently under uniform speed conditions, the velocities of the three drones remain closely aligned. However, when the leading UAV experiences abrupt speed fluctuations, the trailing UAVs struggle to decelerate effectively in time. This limitation becomes particularly critical under high-speed and high-density operational scenarios, highlighting significant safety concerns within the air corridor in the DT.

The DT simulation environment is built based on AirSUMO to simulate the real-time operation process of the model deployed in the urban airspace scenario. By comparing the algorithm prediction results with the simulation operation data, not only can the fitting performance model in the digital space be verified, but also its drone speed response, following stability, and conflict avoidance capabilities in a near-real environment can be further observed, thereby evaluating its usability and practicality in a complex urban airspace

environment. The optimisation comparison curve of Drones 2 is shown in Figure 14, the optimisation comparison curve of Drones 3 is shown in Figure 15, and the speed curve of the optimised drone in DT is shown in Figure 16.

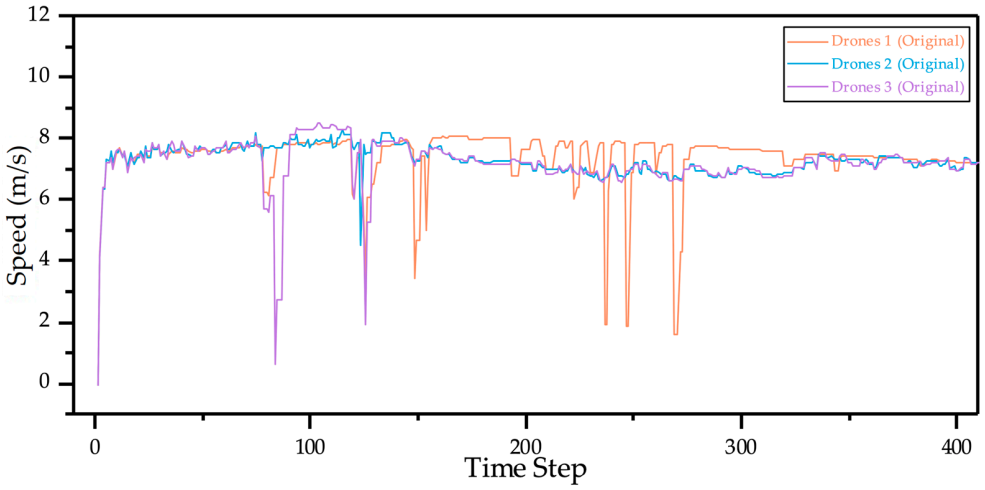


Figure 13. Comparison of original Drones 1, 2, and 3 velocity data.

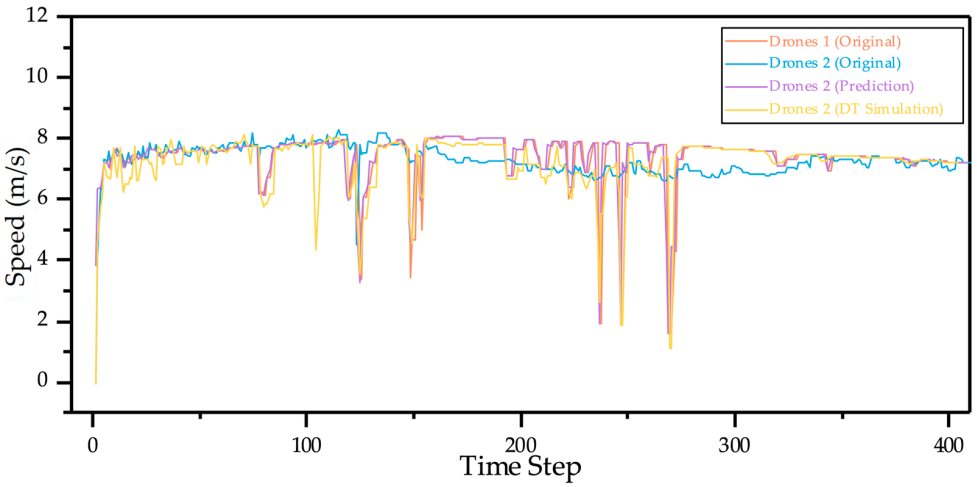


Figure 14. Comparison of original Drones 1 and 2, prediction and DT simulation of Drone 2.

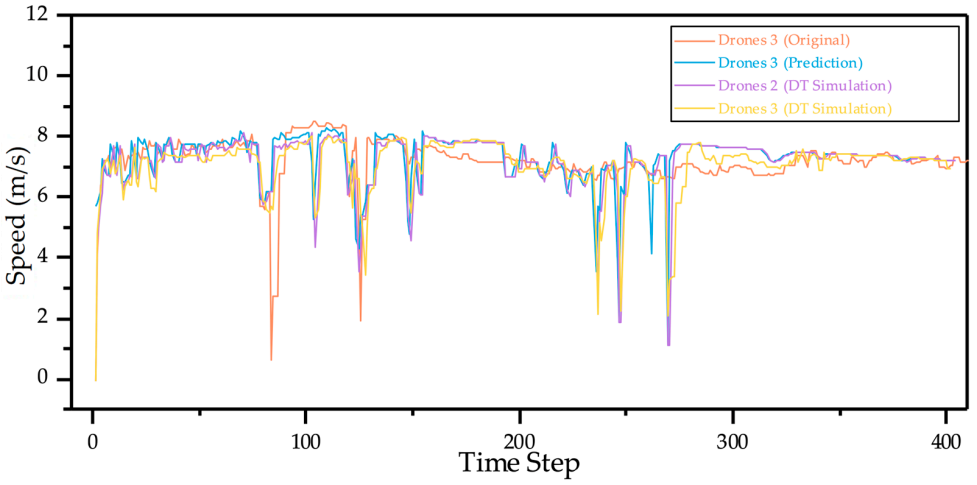


Figure 15. Comparison of original Drones 3, prediction Drones 2, and DT simulation of Drones 2 and 3.

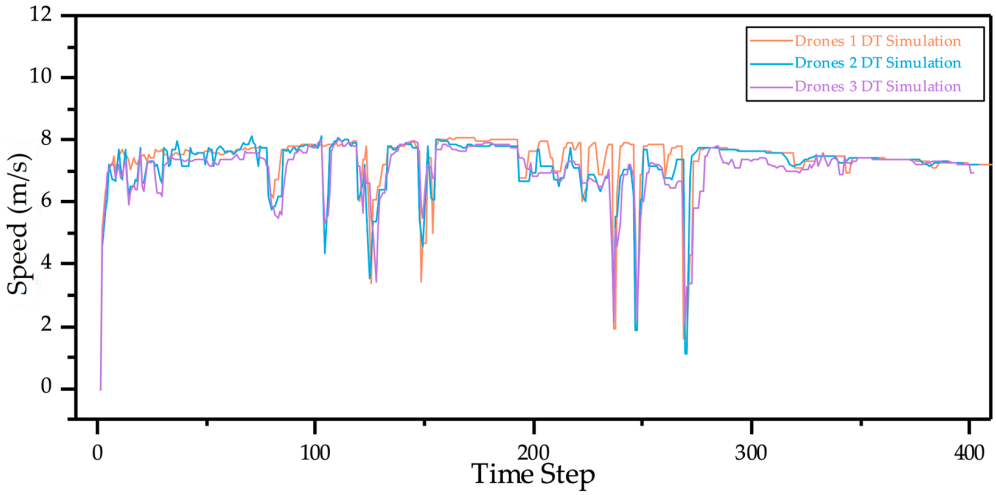


Figure 16. Comparison of original Drones 1 and DT simulation of Drones 2 and 3.

By comparing Figures 13–16, we can see that compared with before LLM optimisation, Drones 2 can give timely feedback and adjust the speed for the sudden speed change during the operation of Drones 1. This section will further analyse the safety of UAS operation for further research.

4.4. Safety Analysis

There is no unified standard for urban low-altitude route design, test environment, and evaluation indicators, no official normative document for reference, and considering the risks of real-world testing, it is difficult to use real drones for verification.

Therefore, drone traffic is simulated through simulation experiments, the risks of using different drones within the route are analysed, and the improvement of BiDGCNLLM optimisation on UAS operation safety is evaluated. The flight speed data of drones uses Original Drones Speed Data and DT Simulation Speed Data, and the design of safety evaluation between drones is referenced; the UAV safety separation as shown in Table 4 [7].

Table 4. UAV safety separation (horizontal) [7].

Type	Mavic Air 2	Inspire 2	MK 300 RTK
Mavic Air 2	10 m	15 m	20 m
Inspire 2	15 m	15 m	20 m
MK 300 RTK	15 m	10 m	25 m

The initial distance between the three drones in the air corridor is set to the safety distance +5 m. The three drones, Mavic Air 2, Inspire 2, and MK 300 RTK, are analysed. The minimum safety separation is known to be 10 m. Therefore, the initial distance of the safety analysis experiment parameters is designed to be 15 m, and the safety distance is 10 m. The potential risks of drone operation in this case are analysed. The drone spacing estimation formula is:

$$d_{i,j}(t) = d_0 + \sum_{k=1}^{t} \left(v_i(k) - v_j(k) \right) \tag{19}$$

where d_0 denotes the initial separation, and $v_i(k)$, $v_j(k)$ represent the velocities of the leading and following UAVs, respectively, at time k.

We specifically examine two critical UAV pairs: Drones 1–2 and Drones 2–3. Under the original control scheme, the relative distance between Drones 1 and 2 drops below the safety threshold in multiple instances, triggering five separate collision-risk events

with durations of 141, 2, 5, 12, and 20 s, respectively. The longest risk episode persists for over two minutes, indicating a severe lack of temporal responsiveness and a potential for rear-end collisions. In contrast, no safety violations are observed in the optimised trajectories. The proposed model effectively regulates inter-drone distances and eliminates all collision risks.

Furthermore, a distance-over-time visualisation reveals that the optimised BiDGCN-LLM predictions maintain a consistently wider buffer, with smoother spacing trajectories. This demonstrates the capability model to proactively adjust UAV velocity in response to the behaviour of neighbouring agents, thereby ensuring robust flight safety even under tightly constrained spacing conditions.

These findings highlight the effectiveness of the optimisation framework in mitigating potential mid-air conflicts in linear UAV formations. The model enhances overall operational robustness in fixed-route urban air corridors by maintaining safe longitudinal separation.

5. Discussion

The BiDGNLLM framework proposed in this study shows excellent speed prediction ability in UAM, which is not only significantly better than traditional time series modelling methods but also shows good generalisation ability and safety assurance potential in the DT environment deployment.

Most existing models fail to fully consider the two-way dynamic dependency between the front and rear drones in the route and lack the modelling and utilisation of broadcast remote sensing data (such as Remote ID) in structured airspace. This study uses Remote ID data as the input source for the first time, augments high-fidelity telemetry information through the AirSUMO simulation environment, and combines multi-level modelling methods to achieve the unity of graph structure perception and language semantic understanding, filling the gap in speed prediction and conflict avoidance research in UAM.

From the experimental results, BiDGCNLLM is significantly better than the comparison models, such as PatchTST and TimeLLM, in multiple key indicators. This performance improvement is mainly attributed to two aspects: First, BiGCN can capture the spatial front-to-back dependency between drones, and is particularly effective in modelling speed linkage characteristics in constrained channels; second, the introduced LLM has strong context perception capabilities, which can extract weak and critical time dynamic features from historical speed sequences, and has higher sensitivity and prediction capabilities for sudden speed fluctuations.

In the Co-simulation DT based on Cranfield University, the prediction results of the BiDGCNLLM are fed back to the AirSim platform. The comparison results show that the optimised Drones 2 and Drones 3 show higher speed responsiveness and trajectory stability during the following process. Especially in the scenario of sudden speed changes, the subsequent drones can adjust their speed in time to avoid serious speed differences. This shows that the model not only has high prediction accuracy but also meets the requirements of real-time reasoning and feedback control, and has the potential to migrate to actual deployment. In addition, the safety analysis results show that the model can actively adjust the following rhythm, has active conflict avoidance capabilities, and improves the stability and safety of the overall airspace operation.

Although the BiDGCNLLM demonstrates strong overall performance, further analysis indicates that certain challenging scenarios remain. In particular, the model exhibits reduced accuracy when the leading drone undergoes abrupt and extreme changes in velocity, such as rapid acceleration or deceleration beyond ± 3 m/s^2. In such cases, the

response of the following drones may display slight latency, potentially compromising short-term coordination. These observations highlight the need for additional strategies to improve model stability in edge cases.

6. Conclusions and Future Work

Based on the internal safety hazards and conflict risks of drones under fixed routes in UAM, this paper proposes a spatio-temporal prediction framework BiDGCNLLM, that integrates dual BiGCN, Dynamic Edge Weight, and LLM. The Remote ID broadcast data was augmented in the DT co-simulation framework built by AirSUMO, integrates spatial topological structure and temporal semantic information, and realises accurate prediction of the evolution trend of drone speed. Experimental results show that this method is significantly better than mainstream time series prediction models in multiple indicators, especially in capturing speed mutations and improving formation response capabilities.

Through deployment verification in the DT built with Cranfield University as the background, BiDGCNLLM shows good optimisation prediction capabilities. The optimised prediction results effectively reduce the conflict risk between drones and maintain stable speed following, providing a new data-driven solution for distributed active collision avoidance in structured airspace. Compared with traditional methods, this study has extended modelling strategies, data sources, and verification mechanisms, providing an important reference for the application of remote sensing broadcast data in low-altitude airspace monitoring and prediction.

Future work will further assess the generalisation capability of the model using diverse Remote ID datasets, while improving its practicality and deployment efficiency in complex urban environments. In addition, operations under adverse conditions, such as engine failure or extreme weather, will be explored to evaluate robustness in safety-critical scenarios. The framework may be extended to core UAM subsystems, including multi-task prediction and congestion alerting, supporting the development of a DT-centred control system. Limitations related to computational cost, real-time feasibility, and generalisation across unfamiliar airspaces will also be addressed through scalable optimisation and broader validation across varied operational settings.

Author Contributions: Conceptualisation, Z.W., J.Z. and A.Z.; methodology, Z.W.; software, Z.W. and J.Z.; validation, Z.W., J.Z., R.W. and A.Z.; formal analysis, A.Z.; investigation, J.Z. and W.B.; resources, J.Z.; data curation, Z.W. and R.W.; writing—original draft preparation, Z.W.; writing—review and editing, Z.W., J.Z., B.K., Y.S., R.W. and A.Z.; visualisation, Z.W., Y.S. and J.Z.; supervision, J.Z., A.Z. and W.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. Kopardekar, P.; Rios, J.; Prevot, T.; Johnson, M.; Jung, J.; Robinson, J.E. Unmanned aircraft system traffic management (UTM) concept of operations. In Proceedings of the AIAA AVIATION Forum and Exposition (No. ARC-E-DAA-TN32838), Washington, DC, USA, 13–17 June 2016. Available online: https://www.faa.gov/sites/faa.gov/files/2022-08/UTM_ConOps_v2.pdf (accessed on 15 July 2025).
2. Goyal, R.; Reiche, C.; Fernando, C.; Serrao, J.; Kimmel, S.; Cohen, A.; Shaheen, S. Urban Air Mobility (UAM) Market Study (No. HQ-E-DAA-TN65181). 2018. Available online: https://ntrs.nasa.gov/api/citations/20190002046/downloads/20190002046.pdf (accessed on 15 July 2025).

3. Belwafi, K.; Alkadi, R.; Alameri, S.A.; Al Hamadi, H.; Shoufan, A. Unmanned aerial vehicles' remote identification: A tutorial and survey. *IEEE Access* **2022**, *10*, 87577–87601. [CrossRef]
4. Ministry of Land, Infrastructure, Transport and Tourism, Japan. Concept of Operations for Advanced Air Mobility. 2024. Available online: https://www.mlit.go.jp/koku/content/001757082.pdf (accessed on 15 July 2025).
5. Zhang, Z.; Zheng, Y.; Li, C.; Jiang, B.; Li, Y. Designing an Urban Air Mobility Corridor Network: A Multi-Objective Optimization Approach Using U-NSGA-III. *Aerospace* **2025**, *12*, 229. [CrossRef]
6. Wang, X.; Yang, P.P.J.; Balchanos, M.; Mavris, D. Urban Airspace Route Planning for Advanced Air Mobility Operations. In Proceedings of the International Conference on Computers in Urban Planning and Urban Management, Montréal, QC, Canada, 12–14 June 2023; Springer: Cham, Switzerland; pp. 193–211. [CrossRef]
7. Li, S.; Zhang, H.; Li, Z.; Liu, H. Air Route Design of Multi-Rotor UAVs for Urban Air Mobility. *Drones* **2024**, *8*, 601. [CrossRef]
8. Altun, A.T.; Hasanzade, M.; Saldiran, E.; Guner, G.; Uzun, M.; Fremond, R.; Tang, Y.; Bhundoo, P.; Su, Y.; Xu, Y.; et al. AMU-LED cranfield flight trials for demonstrating the advanced air mobility concept. *Aerospace* **2023**, *10*, 775. [CrossRef]
9. Shi, Z.; Zhang, J.; Shi, G.; Ji, L.; Wang, D.; Wu, Y. Design of a UAV trajectory prediction system based on multi-flight modes. *Drones* **2024**, *8*, 255. [CrossRef]
10. Li, M.; Huang, Z.; Bi, W.; Hou, T.; Yang, P.; Zhang, A. A fish evasion behavior-based vector field histogram method for obstacle avoidance of multi-UAVs. *Aerosp. Sci. Technol.* **2025**, *159*, 109974. [CrossRef]
11. Duan, X.; Fan, Q.; Bi, W.; Zhang, A. Belief Exponential Divergence for DS Evidence Theory and its Application in Multi-Source Information Fusion. *J. Syst. Eng. Electron.* **2024**, *35*, 1454–1468. [CrossRef]
12. Bagnall, T.M.; Kriz, A.; Briggs, R.; Takamizawa, K. Demonstration and Validation of Remote ID Detect and Avoid. 2025; Federal Aviation Administration. Available online: https://www.faa.gov/uas/programs_partnerships/BAA/BAA004-MosaicATM-Demonstration-and-Validation-of-RID-DAA.pdf (accessed on 15 July 2025).
13. Sacharny, D.; Henderson, T.C.; Marston, V.V. Lane-based large-scale uas traffic management. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 18835–18844. [CrossRef]
14. McCorkendale, Z.; McCorkendale, L.; Kidane, M.F.; Namuduri, K. Digital Traffic Lights: UAS Collision Avoidance Strategy for Advanced Air Mobility Services. *Drones* **2024**, *8*, 590. [CrossRef]
15. Huang, C.; Petrunin, I.; Tsourdos, A. Strategic conflict management using recurrent multi-agent reinforcement learning for urban air mobility operations considering uncertainties. *J. Intell. Robot. Syst.* **2023**, *107*, 20. [CrossRef]
16. Yi, J.; Zhang, H.; Wang, F.; Ning, C.; Liu, H.; Zhong, G. An operational capacity assessment method for an urban low-altitude unmanned aerial vehicle logistics route network. *Drones* **2023**, *7*, 582. [CrossRef]
17. Yi, J.; Zhang, H.; Li, S.; Feng, O.; Zhong, G.; Liu, H. Logistics UAV air route network capacity evaluation method based on traffic flow allocation. *IEEE Access* **2023**, *11*, 63701–63713. [CrossRef]
18. Ruseno, N.; Lin, C.Y.; Guan, W.L. Flight test analysis of UTM conflict detection based on a network remote ID using a random forest algorithm. *Drones* **2023**, *7*, 436. [CrossRef]
19. Cook, B.; Cohen, K.; Kivelevitch, E.H. A fuzzy logic approach for low altitude UAS traffic management (UTM). In Proceedings of the AIAA infotech@ Aerospace, California, CA, USA, 4–8 January 2016; p. 1905. [CrossRef]
20. Neelakandan, D.S.; Al Ali, H. Enhancing trajectory-based operations for UAVs through hexagonal grid indexing: A step towards 4D integration of UTM and ATM. *Int. J. Aviat. Aeronaut. Aerosp.* **2023**, *10*, 5. [CrossRef]
21. Xue, M. Coordination between federated scheduling and conflict resolution in UAM operations. In Proceedings of the AIAA AVIATION 2021 FORUM, Seattle, WA, USA, 2–6 August 2021; p. 2349. [CrossRef]
22. Yahi, N.; Matute, J.; Karimoddini, A. Receding horizon based collision avoidance for uam aircraft at intersections. *Green Energy Intell. Transp.* **2024**, *3*, 100205. [CrossRef]
23. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
24. Yu, B.; Yin, H.; Zhu, Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv* **2017**, arXiv:1709.04875.
25. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.C. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1.
26. Bai, L.; Yao, L.; Li, C.; Wang, X.; Wang, C. Adaptive graph convolutional recurrent network for traffic forecasting. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 17804–17815.
27. Nie, Y.; Nguyen, N.H.; Sinthong, P.; Kalagnanam, J. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv* **2022**, arXiv:2211.14730.
28. Zhao, Y.; Ma, Z.; Zhou, T.; Ye, M.; Sun, L.; Qian, Y. Gcformer: An efficient solution for accurate and scalable long-term multivariate time series forecasting. In Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, Birmingham, UK, 21–25 October 2023; pp. 3464–3473. [CrossRef]
29. Garza, A.; Challu, C.; Mergenthaler-Canseco, M. TimeGPT-1. *arXiv* **2023**, arXiv:2310.03589. [CrossRef]

30. Jin, M.; Wang, S.; Ma, L.; Chu, Z.; Zhang, J.Y.; Shi, X.; Chen, P.Y.; Liang, Y.; Li, Y.F.; Pan, S.; et al. Time-LLM: Time Series Forecasting by Reprogramming Large Language Models. In Proceedings of the International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024. [CrossRef]
31. Ditta, C.C.; Postorino, M.N. Three-Dimensional Urban Air Networks for Future Urban Air Transport Systems. *Sustainability* **2023**, *15*, 13551. [CrossRef]
32. Hohmann, N.; Brulin, S.; Adamy, J.; Olhofer, M. Three-dimensional urban path planning for aerial vehicles regarding many objectives. *IEEE Open J. Intell. Transp. Syst.* **2023**, *4*, 639–652. [CrossRef]
33. Brunelli, M.; Ditta, C.C.; Postorino, M.N. A framework to develop urban aerial networks by using a digital twin approach. *Drones* **2022**, *6*, 387. [CrossRef]
34. Pradhan, P.; Omorodion, J.; Rostami, M.; Venkatesh, A.; Kamoonpuri, J.; Chung, J. Digital Framework for Urban Air Mobility Simulation. In Proceedings of the 2024 IEEE International Symposium on Emerging Metaverse (ISEMV), Bellevue, WA, USA, 21–23 October 2024; pp. 37–40. [CrossRef]
35. Ywet, N.L.; Maw, A.A.; Nguyen, T.A.; Lee, J.W. Yolotransfer-Dt: An operational digital twin framework with deep and transfer learning for collision detection and situation awareness in urban aerial mobility. *Aerospace* **2024**, *11*, 179. [CrossRef]
36. Ancel, E.; Capristan, F.M.; Foster, J.V.; Condotta, R.C. Real-time risk assessment framework for unmanned aircraft system (UAS) traffic management (UTM). In Proceedings of the 17th AIAA AVIation Technology, Integration, and Operations Conference, Colorado, CO, USA, 5–9 June 2017; p. 3273. Available online: https://ntrs.nasa.gov/api/citations/20170005780/downloads/20170005780.pdf (accessed on 15 July 2025).
37. Sherstinsky, A. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Phys. D Nonlinear Phenom.* **2020**, *404*, 132306. [CrossRef]
38. Yi, K.; Zhang, Q.; Fan, W.; He, H.; Hu, L.; Wang, P.; An, N.; Cao, L.; Niu, Z. FourierGNN: Rethinking multivariate time series forecasting from a pure graph perspective. *Adv. Neural Inf. Process. Syst.* **2023**, *36*, 69638–69660.
39. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
40. Siami-Namini, S.; Tavakoli, N.; Namin, A.S. The performance of LSTM and BiLSTM in forecasting time series. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), California, CA, USA, 9–12 December 2019; IEEE: Piscataway, NJ, USA; pp. 3285–3292. [CrossRef]
41. Chen, H.; Tian, A.; Zhang, Y.; Liu, Y. Early time series classification using tcn-transformer. In Proceedings of the 2022 IEEE 4th International Conference on Civil Aviation Safety and Information Technology (ICCASIT), Dali, China, 14–16 October 2022; IEEE: Piscataway, NJ, USA; pp. 1079–1082. [CrossRef]
42. Shi, J.; Wang, S.; Qu, P.; Shao, J. Time series prediction model using LSTM-Transformer neural network for mine water inflow. *Sci. Rep.* **2024**, *14*, 18284. [CrossRef] [PubMed]
43. Wang, S.; Wu, H.; Shi, X.; Hu, T.; Luo, H.; Ma, L.; Zhang, J.Y.; Zhou, J. Timemixer: Decomposable multiscale mixing for time series forecasting. *arXiv* **2024**, arXiv:2405.14616. [CrossRef]
44. Wu, H.; Hu, T.; Liu, Y.; Zhou, H.; Wang, J.; Long, M. Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv* **2022**, arXiv:2210.02186.
45. Wen, Z.; Zhao, J.; Xu, Y.; Tsourdos, A. A co-simulation digital twin with SUMO and AirSim for testing lane-based UTM system concept. In Proceedings of the 2024 IEEE Aerospace Conference, Big Sky, MT, USA, 2–9 March 2024; IEEE: Piscataway, NJ, USA; pp. 1–11. [CrossRef]
46. Unreal Engine. The Most Powerful Real-Time 3D Creation Tool. 2023. Available online: https://www.unrealengine.com/en-US (accessed on 15 July 2025).
47. Cesium. Cesium for Unreal. Cesium. 2022. Available online: https://cesium.com/platform/cesium-for-unreal/ (accessed on 15 July 2025).
48. Conrad, C.; Delezenne, Q.; Mukherjee, A.; Mhowwala, A.A.; Ahmed, M.; Zhao, J.; Xu, Y.; Tsourdos, A. Developing a digital twin for testing multi-agent systems in advanced air mobility: A case study of cranfield university and airport. In Proceedings of the 2023 IEEE/AIAA 42nd Digital Avionics Systems Conference (DASC), Barcelona, Spain, 1–5 October 2023; IEEE: Piscataway, NJ, USA; pp. 1–10. [CrossRef]
49. Zhao, J.; Conrad, C.; Delezenne, Q.; Xu, Y.; Tsourdos, A. A digital twin mixed-reality system for testing future advanced air mobility concepts: A prototype. In Proceedings of the 2023 Integrated Communication, Navigation and Surveillance Conference (ICNS), Herndon, VA, USA, 18–20 April 2023; IEEE: Piscataway, NJ, USA; pp. 1–10. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.



https://dspace.org/

BiDGCNLLM: A graph–language model for drone state forecasting and separation in urban air mobility using digital twin-augmented remote ID data

Wen, Zhang; Wen, Junjie; Zhang, An; Bi, Wenhao; Kuang, Boyu ...

2025-07-19
MDPI
https://doi.org/10.3390/drones9070508
Wen Z, Zhao J, Zhang A, et al., (2025) BiDGCNLLM: A graph–language model for drone state forecasting and separation in urban air mobility using digital twin-augmented remote ID data. Drones, Volume 9, Issue 7, July 2025, Article number 508

Downloaded from a DSpace repository. https://dspace.org/