4D Aircraft Trajectory Prediction with a Generative Deep Learning and Clustering Approach

Haoyuan ZHANG‡, Zhizhao LIU*

The Hong Kong Polytechnic University (PolyU), 181 Chatham Road South, Hung Hom, Kowloon, Hong Kong, P. R. China.

Abstract: Medium- and long-term 4D aircraft trajectory prediction (TP) is a critical task in air traffic management (ATM). This paper addresses the issue of existing medium- and long-term TP methods, which have difficulty accurately fitting aircraft trajectory data distributions. We propose a 4D TP method based on K-medoids clustering and a conditional table generative adversarial network (CTGAN), called C-CTGAN. Comparative experiments with four long short-term memory (LSTM)-based models and the original CTGAN model show that the TP accuracy of the proposed model is significantly greater than that of other models when predicting medium- and long-term trajectories. When using trajectory datasets without circling and a prediction time span of 10 minutes, compared to the convolutional neural network (CNN)-LSTM model, the C-CTGAN model reduces the mean absolute errors (MAEs) of the core trajectory parameters, such as latitude, longitude, geometric altitude, and ground speed, by 69.89%, 15.00%, 74.07%, and 84.21%, respectively. Compared to those of the original CTGAN model, the MAEs are reduced by 20.43%, 39.09%, 31.98%, and 17.07%, respectively, with the new model. When using trajectory datasets with circling, compared to the CNN-LSTM model, the C-CTGAN model yields MAE reductions of 14.08%, 23.68%, 31.46%, and 2.86%, respectively. Moreover, compared to those of the original CTGAN, the MAEs are 34.88%, 2.69%, 23.16%, and 73.91% lower, respectively, when the proposed model is applied.

Keywords: 4D trajectory; GAN; CTGAN; medium- and long-term trajectory prediction; ADS-B; deep learning

1. Introduction

As the air traffic scale continues to grow, the contradiction between aviation demand and airspace

‡ Ph.D. Candidate, Department of Land Surveying & Geo-Informatics (LSGI).

* Professor, Department of Land Surveying & Geoinformatics (LSGI), Corresponding author.

capacity has become increasingly prominent. To alleviate this problem, the International Civil Aviation Operation (ICAO) regards trajectory-based operations (TBOs) as essential to the air traffic management (ATM) system [1]. By sharing an aircraft's trajectory, aircraft trajectory predictions (TPs) can be used as a reference for TBOs to achieve efficient management and utilization of airspace.

According to the selected time scale, aircraft TP can be divided into short-term and medium- and long-term prediction tasks [2]. Generally, short-term prediction refers to TP that occurs within a few minutes or less [3] and is mainly used for conflict detection and resolution [4, 5]. The medium- and long-term prediction tasks refer to TPs with a time scale exceeding ten minutes. These predictions are primarily used for coordinating flight missions and formulating flight plans. The accuracy of flight information decreases with the increase of prediction time interval [6]. Compared with short-term predictions, medium- and long-term predictions can effectively improve airspace utilization and aviation operation efficiency and are crucial for improving situational airspace awareness and flight flow management [7].

This paper is focused on medium- and long-term aircraft TP. This paper addresses this challenge by proposing a four-dimensional (4D) aircraft TP model based on clustering and a conditional table generation adversarial network (C-CTGAN). Experiments demonstrate that this approach can achieve high-precision medium- and long-term TP. The main contributions of this paper are as follows:

(1) A hybrid model is proposed based on the characteristics of ADS-B data. We optimize the original CTGAN model and design, develop, and verify an aircraft TP model based on K-medoids clustering and the CTGAN framework, referred to as C-CTGAN. The trajectory clustering results are used as additional parameters to enhance the accuracy of predictions.

(2) For medium- and long-term TP, evaluation methods and metrics are designed for the corresponding models and used to assess and verify the accuracy and stability of the models.

(3) We conduct a comparative analysis of the proposed model and other existing models, such as long short-term memory (LSTM), CNN-LSTM, CNN-LSTM-Attention, CNN-BiLSTM, and the original CTGAN, to evaluate the performance of the proposed model. The analysis results demonstrate that the proposed C-CTGAN model yields a significant improvement in accuracy in medium- and long-term TPs.

The structure of this paper is as follows. Section 2 provides a literature review of the current background of aircraft TP. Section 3 presents the overall framework of the method developed in this paper, which includes improvements to CTGAN, the selection of clustering models, the design of model evaluation metrics, the model training process, and the generation of predicted trajectories. Section 4

shows the experimental situation and results. Section 5 discusses our contributions and their potential implications. The conclusions of this work are presented in section 6.

2. Related work

The essence of aircraft TP is to utilize a trajectory model to predict aircraft flight status in the future, which can accurately reflect the comprehensive impact of multiple parameters, e.g., flight intention, environmental conditions, aircraft performance, flight rules, and procedures [8]. There are two major types of medium- and long-term aircraft TP methods, namely, aircraft planning- and machine learning-based models [9]. The former is suitable for simple linear track prediction, whereas the latter machine learning models are applicable for complex nonlinear track prediction.

Aircraft planning is a preformulated trajectory planning task that encompasses the relationships among flights, air traffic control, and airports [10]. Pilots and air traffic controllers do their best to ensure that an aircraft's required time of arrival (RTA) matches its flight plan. Aircraft planning-based models are widely used for medium- and long-term TP [11]. However, this type of method does not consider flight uncertainty and is susceptible to external influences caused by particular circumstances, such as temporary airspace control and weather changes [12]. The results of aircraft planning-based TP often significantly deviate from the actual trajectories [13].

Machine learning models have advantages over flight planning-based models. They can extract trajectory features, identify hidden patterns, and obtain TPs from historical trajectory data, and they have become a new research direction [6, 14]. This type of model generally uses a filter-based method [15] or a deep learning-based method [16] to predict a trajectory by extracting trajectory change characteristics through data analysis. Machine learning models can be categorized into three main types: regression models, neural network models, and others. In aircraft TP, the commonly used regression methods include local-weighted linear regression and local-weighted polynomial regression. For instance, Leege et al. [17] predicted aircraft arrival times using aircraft type, ground speed, altitude, and meteorological data as model inputs, employing a stepwise regression approach to systematically determine the inputs and their functions included in the predictive model based on interpretability. Hamed et al. [18] used a standard point mass model and statistical regression methods to predict aircraft climbing altitudes. Due to the ability of neural networks to effectively approximate arbitrary continuous mappings, this approach is preferable to general linear regression. Hence, in recent years, many researchers have used neural networks to address TP problems.

In addition to regression models and neural networks, other machine learning methods, such as genetic algorithms (GAs), ant colony algorithms, and support vector machines (SVMs), have also emerged. Furthermore, clustering algorithms [19,20], such as K-means and density-based clustering, are used for aircraft TP. These algorithms require the use of appropriate trajectory similarity metrics to enhance clustering effectiveness. For example, Barratt et al. [20] studied a probabilistic ballistic trajectory generation model for terminal airspace, initially clustering trajectories using the K-means method and constructing Gaussian mixture models from the clusters to achieve precise trajectory inference. Combining clustering with machine learning prediction methods can significantly improve the prediction accuracy of large-scale clusterable datasets. Therefore, applying machine learning and clustering to aircraft TP is a valuable and meaningful research topic [21]. Currently, neural network models are commonly used in aircraft TP. In practical applications, hybrid models are typically adopted to address specific problems according to different application scenarios.

In recent years, LSTM neural network models have been developed to effectively capture sequence correlations. This approach has been widely used for aircraft TP [22-25]. Although LSTM models have excellent time series memorization ability, their prediction ability is still limited by chaotic phenomena in the TP process [26]. As the prediction period becomes longer, the output probability density of the predicted trajectory becomes more uniform. This phenomenon originates from flight uncertainty. Thus, it is difficult for most LSTM models to provide accurate medium- and long-term forecasts. In response to this problem, various hybrid LSTM models have been proposed, such as attention-LSTM [27] and convolutional neural network-bidirectional LSTM (CNN-BiLSTM) [28], to improve accuracy for medium- and long-term TP. However, limited by the logic of the LSTM algorithm, it is challenging for these models to substantially improve the performance of medium- and long-term TP.

Recently, generative adversarial networks (GANs), as a new neural network technology, have been widely used in image generation [29], data generation [30], image resolution enhancement [31], trajectory synthesis [32], and other fields. This type of network can learn the distribution of given data based on a competition between a generator and a discriminator. The outstanding features and performance of this algorithm have attracted attention from researchers, especially regarding its applications in aircraft TP. GANs have been proven useful for flight trajectory distribution learning [33] and short-term TP considering weather factors [34]. In a recent study, Wu et al. [35] converted aircraft time series data into RGB images and then used an image-based GAN for long-term 4D TP. In that study,

the number of calculations was extremely high due to the conversion of aviation data to images. To ensure the calculation efficiency of the method, only a small number of 4D trajectory points (200 in that paper) were selected from one route as inputs for each prediction, so long-term TP accuracy could not be guaranteed. The above research based on GANs preliminarily verified the feasibility of applying this methodology to aircraft TP.

The most recent algorithm developed based on GAN technology is CTGAN. It has made significant breakthroughs in processing unbalanced data [36] and detecting network attacks [37]. The architecture of CTGAN combines the advantages of the conditional GAN (CGAN) and the table GAN (TGAN) [38]. Notably, tabular data can be directly used as an input considering the unique problems of tabular data, such as the non-Gaussian distributions of continuous columns, the unbalanced distributions of discrete columns, and mixtures of discrete and continuous columns [39]. This model outperforms traditional GANs in aspects such as data feature mining [40]. Notably, in tabular data processing, the CTGAN has shown to outperform the TGAN, which uses the LSTM algorithm in its generator [41].

Automatic dependent surveillance-broadcast (ADS-B) is the main data source used for aircraft TP [27]. These data are usually stored in ASTERIX class 023 format. They are easily converted into tabular datasets and are characterized by both non-Gaussian and unbalanced distributions. The characteristics of ADS-B data coincide with the outstanding advantages of the CTGAN. Furthermore, compared to GANs, CGANs, and LSTMs, the CTGAN does not have specific requirements for organizing sample data. GANs and CGANs require consistent sample data lengths, which for ADS-B data means the same number of trajectory points in a flight trajectory. LSTM models require sample data to be organized in consistent intervals to ensure TP accuracy. Thus, whether using GANs, CGANs, or LSTMs, considerable preprocessing is needed for ADS-B data, increasing the overall processing time. In this context, the use of the CTGAN to assess medium- and long-term aircraft TPs based on ADS-B data has significant potential. However, to date, no research has reported the use of CTGAN-based hybrid models for four-dimensional aircraft TP.

In summary, the use of machine learning to support medium- and long-term aircraft TP is still in the exploratory research phase, and obtaining TPs with high accuracy and reliability is still a challenge. For these reasons, we propose a model for medium- and long-term aircraft TP by combining clustering algorithms with the CTGAN architecture.

3. Methods

The overall framework of the 4D aircraft TP method proposed in this study, which is based on trajectory clustering and the CTGAN (called C-CTGAN in the paper), is shown in **Figure 1**. It includes four parts: data preprocessing, trajectory clustering, model training, and trajectory prediction.

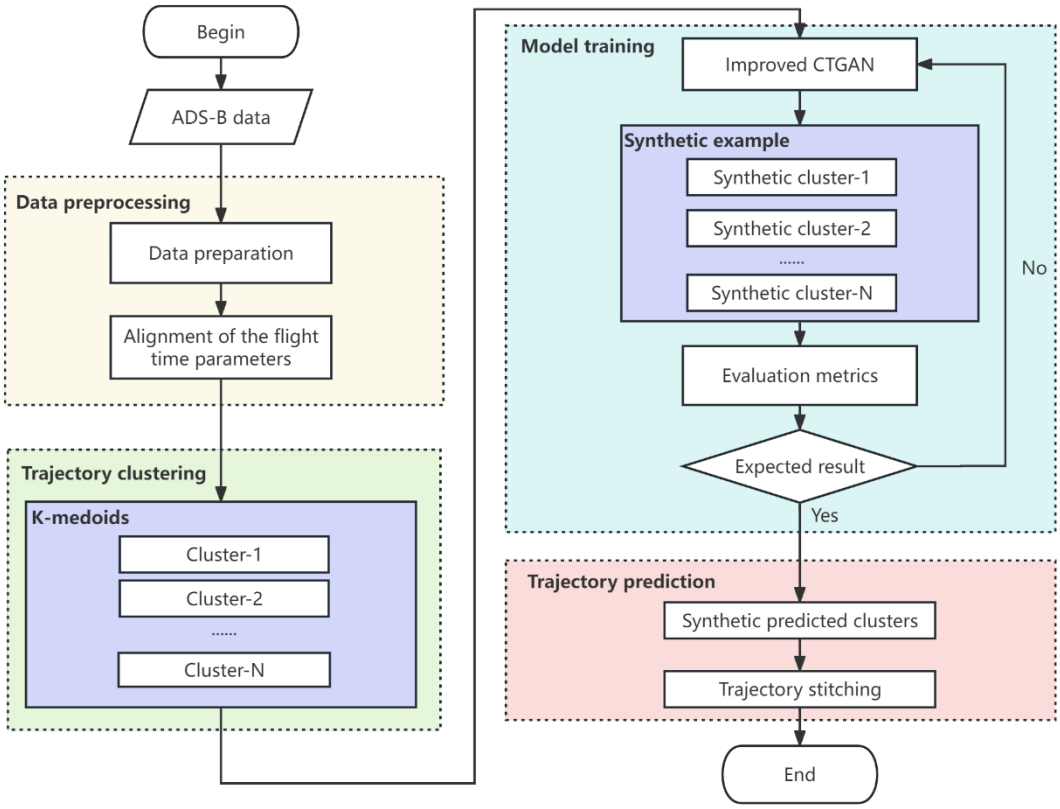

Figure 1. The general framework of the 4D aircraft TP method based on clustering and an improved CTGAN architecture.

3.1 Data preprocessing

3.1.1 Data preparation

Our primary data source in this study was ADS-B data stored in the ASTERIX category 023 format. ADS-B is a surveillance technology used in aviation that enables aircraft to broadcast essential information. The raw ADS-B data, used as the foundation for our analysis, are chronological records of aircraft broadcast messages, each representing an individual unit. These messages include aircraft identification, transmission time, latitude, longitude, altitude, heading, ground speed, and data quality information.

The raw data are characterized by factors such as data gaps, anomalies, and duplicate information. To address these issues, an initial data cleaning process was employed to eliminate trajectory data units with incomplete or irregular attributes. Subsequently, we chronologically categorized and organized the cleaned data based on call signs. In the final step, we refined the dataset by retaining only parameters relevant to constructing a four-dimensional flight trajectory. This involved using the reception time as the temporal parameter, the geometric altitude as the altitude parameter, the ground speed as the speed parameter, and the latitude and longitude as parameters representing the aircraft's position. Additionally, the heading angle information was retained.

With this meticulous approach, we transformed the raw, unprocessed ADS-B data into distinct flight datasets in tabular form. This not only facilitated data organization but also ensured that the characteristics and accuracy of the flight data were preserved to the greatest extent possible for subsequent analyses.

3.1.2 Alignment of the flight time parameters

Considering that there are differences in the time of each flight on the same route when using data from multiple flights to train the CTGAN model, it is necessary to align the time parameter. Thus, we selected the takeoff time of the aircraft as the relative time origin for each departing flight. For arriving flights, we selected a must-pass waypoint on the same route as the reference point used to calculate the relative time. Depending on the situation, we added an offset to serve as the time origin for each arriving flight to ensure that the time parameter for trajectory points was positive. Increasing the trajectory point density near a specified waypoint was crucial for preventing significant errors from arising from the use of long time intervals between tracking points. Consequently, it was necessary to increase the trajectory point density near the specified waypoint when calculating the relative time origin for each trajectory. Then, we selected the time of the nearest trajectory point within a certain radius of the waypoint as the relative time origin, ensuring the accuracy of time alignment.

3.2 Trajectory clustering

Clustering, an unsupervised machine learning method, has been widely used in various fields [42], such as data mining, pattern recognition [43], image analysis, bioinformatics, and recommendation systems. Trajectory clustering can be used to discover hidden trajectory rules in an airspace model that are not easy for humans to identify. The use of such hidden rules helps improve the accuracy of TP [44].

For the CTGAN architecture, existing studies have shown that the CTGAN model integrated with the K-means method can effectively learn the distribution of data and then address imbalance issues in tabular data [36]. This is because clustering can be used to divide and classify data, decompose complex problems into relatively simple ones, and reduce the difficulty of problem processing. Based on this classification approach, the CTGAN can effectively learn the distribution characteristics of the data. For flights on the same route, an aircraft's trajectory parameters are primarily related to the aircraft's planar position (latitude and longitude). Considering this property, clustering is used to extract the plane position features from ADS-B data. The clustered trajectories are input into the CTGAN model as additional parameters to improve the TP accuracy.

3.2.1 K-medoids clustering algorithm

In this paper, we use the K-medoids clustering method for trajectory clustering, which can effectively distinguish the plane features of aircraft trajectories. Compared with the K-means clustering method, K-medoids clustering is insensitive to noise and isolated points and has strong robustness [45]. Compared with the density-based spatial clustering of applications with noise (DBSCAN) clustering method, K-medoids clustering only needs to adjust the number of clusters to obtain a good clustering effect. Considering that outliers inevitably exist in ADS-B data and that aircraft trajectory clustering is more suitable than DBSCAN clustering, we selected the K-medoids clustering method.

3.2.2 Cluster quantity selection

The key to K-medoids clustering is to choose an appropriate number of clusters K. In this paper, the elbow method and silhouette coefficient (SC) [44] are used to select the expected number of clusters. The elbow method generally uses the sum of squared errors (SSE) to calculate the overall loss of each cluster, and the formula is as follows:

$$SSE = \sum_{i=1}^{k} \sum_{p \in C_i} |p - m_i|^2 \quad (1)$$

where C_i refers to the i_{th} cluster, p represents a sample point in C_i, and m_i is the centroid of C_i. When the appropriate number of clusters is reached for the given data, the changes in the SSE tend to be stable. K is the expected number of groups, which are generally located at the elbow of the curve, as shown in **Figure 2**.

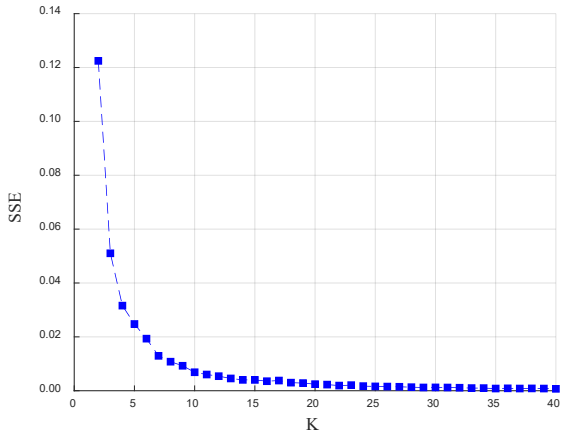

Figure 2. The SSE curve for determining the appropriate K value via the elbow method.

The silhouette coefficient reflects the quality of trajectory clustering based on cluster closeness and the separation between clusters; it is expressed as:

$$\text{effect}.SC(x) = \frac{b(x)-a(x)}{max\{a(x),b(x)\}} \quad (2)$$

where a(x) is the average distance between x and the other trajectories in the cluster and b(x) is the average distance between x and the other trajectories in the nearest-neighbor cluster. Generally, the closer SC is to 1, the better the clustering effect.

3.3 Model training

3.3.1 Improved CTGAN architecture

In this work, three improvements are made to the original CTGAN to improve the accuracy of TP by combining the characteristics of the ADS-B data and the results of aircraft trajectory clustering.

The first improvement addresses the aircraft characteristics contained in the ADS-B data. We perform sampling with the actual data frequencies instead of logarithmic frequencies. The purpose of this strategy is to prevent the synthesizer from unrealistically oversampling rare classes. In this context, rare classes refer to specific categories or attributes within the ADS-B data that occur relatively infrequently. Oversampling these occasional classes could disrupt model convergence [46]. Therefore, actual data frequencies are adopted to maintain a balanced and representative dataset for improved model training and performance.

In addition, we use target encoding instead of one-hot encoding to cluster the trajectory results. When the number of cluster centers increases, one-hot encoding adds many dimensions to the dataset but does

not provide much information. These dimensions can lead to very sparse results from the generator, making optimization challenging, especially for neural networks. Even worse, each pair of information-sparse columns is linearly related. This linear relationship means that one variable can be easily predicted using other variables, resulting in problems of parallelism and multicollinearity in high-dimensional cases. In contrast, target encoding only occupies one feature space for processing trajectory clusters, so it can represent the relationships between categorical and target variables more directly. Therefore, we choose target encoding.

The third improvement is that we input clustering data in the generator as additional parameters to improve the TP accuracy. **Figure 3** shows the architecture of the improved CTGAN. Differences from the original CTGAN are marked in red.

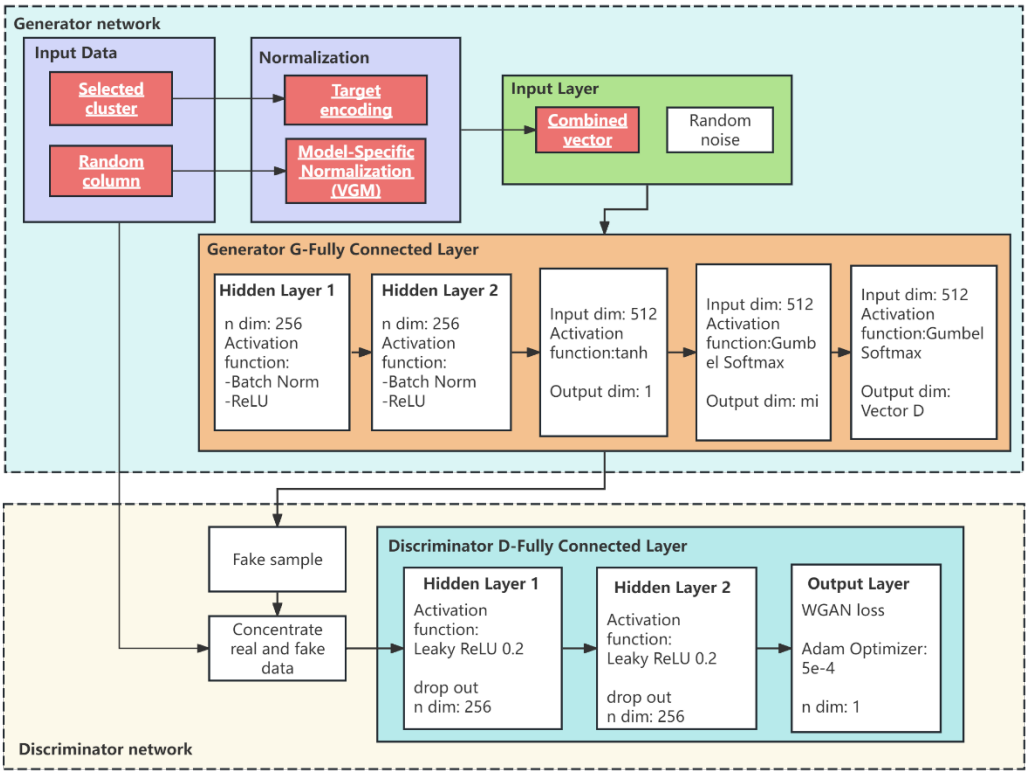

Figure 3. Improved CTGAN architecture based on the K-medoids clustering method.

The improved CTGAN architecture consists of a generator network and a discriminator network. In the generator network, the columns of the ADS-B data are normalized using a variational Gaussian mixture model. The selected clusters are converted to binary columns via target encoding and then concatenated with the normalized column vectors to form a combined vector. We input this vector and random noise into the fully connected generator layer as inputs. The first and second hidden layers are

fully connected with 256 nodes, and the rectified linear unit (ReLU) function (set to 750 dimensions to produce stable and efficient updates) is applied for batch normalization. [47]. The last layer converts the neurons from 512 dimensions to 1 output dimension. The alpha values of the Gumbel softmax [48] function and the tanh function are set to 0.2, the same as in the original CTGAN. Consequently, Equation 3 can be used to describe the generator network:

$$\begin{cases} h_0 = z \oplus cond \\ h_1 = h_0 \oplus \mathrm{ReLU}\left(\mathrm{BN}\left(\mathrm{FC}_{|cond|+|z|\to 256}(h_0)\right)\right) \\ h_2 = h_1 \oplus \mathrm{ReLU}\left(\mathrm{BN}\left(\mathrm{FC}_{|cond|+|z|+256\to 256}(h_1)\right)\right) \\ \hat{\alpha}_i = \tanh\left(\mathrm{FC}_{|cond|+|z|+512\to 1}(h_2)\right) \qquad 1 \le i \le N_c \\ \hat{\beta}_i = \mathrm{gumbel}_{0.2}\left(\mathrm{FC}_{|cond|+|z|+512\to m_i}(h_2)\right) \quad 1 \le i \le N_c \\ \hat{d}_i = \mathrm{gumbel}_{0.2}\left(\mathrm{FC}_{|cond|+|z|+512\to |D_i|}(h_2)\right) \quad 1 \le i \le N_d \end{cases} \tag{3}$$

where $\oplus$ represents the vector concatenation operation; z represents random noise; $\mathrm{FC}_{u\to v}(x)$ denotes the application of a linear transformation to a u-dimensional input to obtain a v-dimensional output; and $\mathrm{gumbel}_{\tau}(x)$ represents the application of the Gumbel softmax function with a parameter τ to a vector x. The $\tanh$, BN and ReLU functions are used for batch normalization.

The discriminator network adopts the same settings as those of the original CTGAN architecture. The PacGAN framework [49] is employed, and 10 samples are used in each Pac to prevent model loss. We uniformly set the leaky ReLU parameter to 0.2 and the dropout function [50] parameter to 0.3. Finally, the loss is determined based on a Wasserstein GAN (WGAN) and used to linearly transform the gradient [51]. The learning rate of adaptive moment estimation (Adam) optimization is set to $5 \cdot 10^{-4}$. This discriminator network can be interpreted as follows:

$$\begin{cases} h_0 = r_1 \oplus .. \oplus r_{10} \oplus cond_1 \oplus .. \oplus cond_{10} \\ h_1 = \mathrm{drop}(\mathrm{leaky}_{0.2}(\mathrm{FC}_{10|r|+10|cond|\to 256}(h_0))) \\ h_2 = \mathrm{drop}\left(\mathrm{leaky}_{0.2}\left(\mathrm{FC}_{256\to 256}(h_1)\right)\right) \\ C(\cdot) = \mathrm{FC}_{256\to 1}(h_2) \end{cases} \tag{4}$$

where drop denotes the dropout, $\mathrm{leaky}_{\gamma}(x)$ represents applying leaky ReLU activation to x with a leaky ratio of γ, and $C(\cdot)$ represents the architecture of the discriminator (with 10 samples).

3.3.2 Model evaluation metrics

During model training, the CTGAN treats each row of tabular input data as an independent sample for

random sampling without considering the order between rows. It learns to extract the overall distribution characteristics of the data through these samples. Based on the input data and the learned distribution characteristics, it outputs generated data in tabular form, where the rows are randomly arranged. For aircraft TP models, the input to the CTGAN generator is typically actual time series trajectory data. The generated output data consist of nontime-series random data that reflect the distribution characteristics of the trajectories. Therefore, there is no direct one-to-one correspondence between them. For the evaluation of model training results, if the distributions of two datasets are very close, the means of their trajectory parameters will also be very close. The mean values can approximately reflect the gap between the actual distribution and the generated distribution, so the mean values can be used to reflect the similarity between the generated data and the input dataset. To accurately evaluate the model, we use the maximum mean discrepancy (MMD) [52] from the data distribution perspective and the mean Euclidean distance (MED) from the trajectory distance perspective.

(1) Maximum mean discrepancy

The MMD is a commonly used GAN evaluation indicator. It finds continuous functions f in the sample space and calculates the mean difference among different distribution functions f to obtain the average f between the analyzed datasets. To determine whether the two distributions are similar, we use the mean of the generated data to calculate the MMD. If a Gaussian kernel function is used to replace f, the formula can generally be interpreted as:

$$MMD^2(X,Y) = \left\| \frac{1}{n^2}\sum_i^n \sum_{i'}^n k\left(x_i, x_i'\right) - \frac{2}{nm}\sum_i^n \sum_j^m k\left(x_i, y_j\right) + \frac{1}{m^2}\sum_j^m \sum_{j'}^m k\left(y_j, y_j'\right) \right\|_H \quad (5)$$

$$k(u,v) = e^{\frac{-\|u-v\|^2}{2\sigma^2}} \quad (6)$$

where X refers to a random variable in the generated distribution; Y refers to a random variable in the actual distribution; n represents the maximum number of variables of *X*; m represents the maximum number of variables of Y; k represents the Gaussian kernel function; and σ is the width parameter of the kernel function. The smaller the MMD is, the more accurate the model is.

(2) Mean Euclidean distance

The MED is a typical metric for calculating the distance between trajectories. The ADS-B data can provide information such as longitudes, latitudes, and altitudes (geometric altitudes in ADS-B) in the WGS-84 coordinate system. The latitude and longitude are measured in degrees, and altitude is usually measured in feet. The latitude and longitude ranges are $[0, \pm 90]^\circ$ and $[0, \pm 180]^\circ$, respectively, and the

altitude can be hundreds or thousands of times the magnitude of the longitude and latitude. Therefore, a unified dimension is needed to reduce dimensional differences. We first transform these three parameters into a Cartesian coordinate system. Then, based on a point-by-point calculation of the Euclidean distance between the two datasets, the MED is obtained by taking the average. The calculation equation is as follows:

$$MED = \frac{1}{n}\sum_{i=1}^{n}|s_i - \widehat{s_i}| \tag{7}$$

where s_i represents the average trajectory point of the generated dataset and $\widehat{s_i}$ represents the average trajectory point of the real dataset.

3.4 Trajectory prediction

The goal of 4D TP is to predict the spatiotemporal state sequence of an aircraft through a trained model. Every track point S_i is represented by time t, latitude x, longitude y, geometric height z, ground speed v_{xy}, and track angle h, as shown below:

$$S_i = \{t, x, y, z, v_{xy}, h\} \tag{8}$$

If the 4D trajectory is regarded as a random variable P, the historical trajectory set can be expressed as $T = \{T_1 = s_{1:N}^1, \ldots T_k = s_{1:N}^k, \ldots, T_M = s_{1:N}^M\}$, where N denotes the number of points in the track and M denotes the number of historical tracks. Assuming that T follows the distribution of $P = P(s_{1:N}; \theta)$, where θ is a parameter, the 4D TP problem can be expressed as $P(s_{1:N}; \theta)$. According to the maximum likelihood estimation, the 4D trajectory distribution is expressed as:

$$^{argmax}_{\theta} p(D|\theta) = {}^{argmax}_{\theta} \prod_{i=1}^{M} P(s_{1:N}^i; \theta) \tag{9}$$

In this article, the essence of model training is to learn the distribution of $P(s_{1:N}^i; \theta)$. Therefore, the distribution of the input trajectories is learned among clusters, and the corresponding predicted trajectories are generated according to the clusters and spliced according to time order. The result is a predicted trajectory that corresponds to the input trajectory distribution.

4. Experiments and analyses

4.1 Data preparation

4.1.1 Experimental data

The volume of daily traffic at Hong Kong International Airport is enormous and its airspace structure is complex and representative of those of major airports. In this paper, to ensure the effect of model

training, we use the ADS-B data for the sector near the Hong Kong International Airport to perform aircraft TP; specifically, the dataset collected by the Hong Kong Observatory (HKO) from December 23, 2016, to December 27, 2016, is used.

After completing data preprocessing, we obtained 2100 complete flight trajectories (including inbound and outbound flights) from the selected ADS-B dataset, and each flight trajectory contained at least 5000 trajectory points. The average time interval between the ADS-B messages in this dataset is between 0.5 and 1.5 s, and the maximum interval is less than 10 s, so it can genuinely reflect the aircraft trajectories in the area. Since the flights from the KADLO waypoint to the VHHH waypoint (airport) in our dataset account for 1/5 of all the data and the flight time of a single flight on this route is more than one hour, we focused on this route to conduct medium- and long-term TP experiments and analyses. **Table 1** shows some selected data samples after preprocessing the data for flight CPA405. Time indicates the relative time after alignment (refer to section 3.1.2), Lat stands for latitude, Lon stands for longitude, GH represents the geometric height, GS denotes the ground speed, and TA signifies the track angle. **Figure 4** shows the trajectories of the inbound flights on this route.

Table 1. Sample of CPA405 data.

Time (s)	Lat (°)	Lon (°)	GH (ft)	GS (NM/s)	TA (°)
0.8984	22.9492	118.5319	36075	0.122925	237.4585
1.8203	22.9481	118.5301	36075	0.122925	237.4585
2.2968	22.9477	118.5293	36075	0.122925	237.4585
3.3906	22.9464	118.5271	36075	0.123047	237.3486
4.8281	22.9448	118.5245	36050	0.123047	237.3486

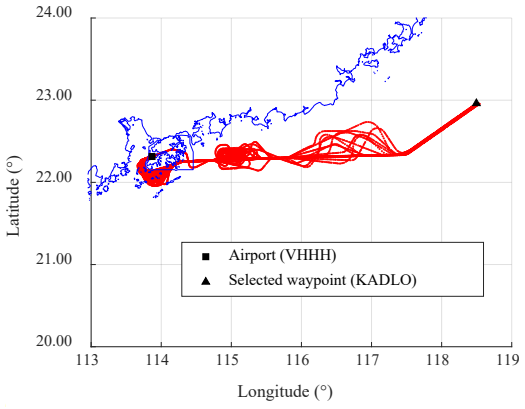

Figure 4. Inbound flight trajectories from KADLO to VHHH from December 23, 2016, to December 27, 2016, in the Hong Kong region.

4.1.2 Data processing and grouping

First, following the method outlined in section 3.1.1, ADS-B data are initially converted from the ASTERIX category 023 format to tabular form, followed by data cleaning, trajectory organization, and time alignment processing. Next, to validate the accuracy, adaptability, and stability of the TP model, two groups of data are selected from the preprocessed trajectory dataset, each consisting of 30 trajectories. The first sample data group consists of trajectories for flights that directly arrive and land, with relatively simple trajectory forms and no holding pattern data. The second sample data group consists of trajectories for flights with circling, with more complex trajectory forms. The trajectory conditions for each group are shown in **Figure 5**. Furthermore, the LSTM model indirectly estimates the time parameter through the arrangement of trajectory points. Therefore, the sample data must be processed into a trajectory composed of a series of trajectory points with the same time interval. To compare with those of LSTM-based models, the trajectory points of both data groups are recalculated and organized with equal time intervals (set as 1 second). Finally, for metric evaluation, each sample data group is further divided into training and testing sets at a ratio of 0.8 to 0.2.

For the C-CTGAN model, we express the basic characteristics of the data distributions using probability density functions (PDFs) of the generated dataset and the real dataset. We use the proposed model evaluation metrics, MMD and MED, to validate the model's accuracy. In a comparative experiment, we used the mean absolute error (MAE) to uniformly measure the accuracy of the prediction results of the C-CTGAN, an LSTM-based model, and the original CTGAN method.

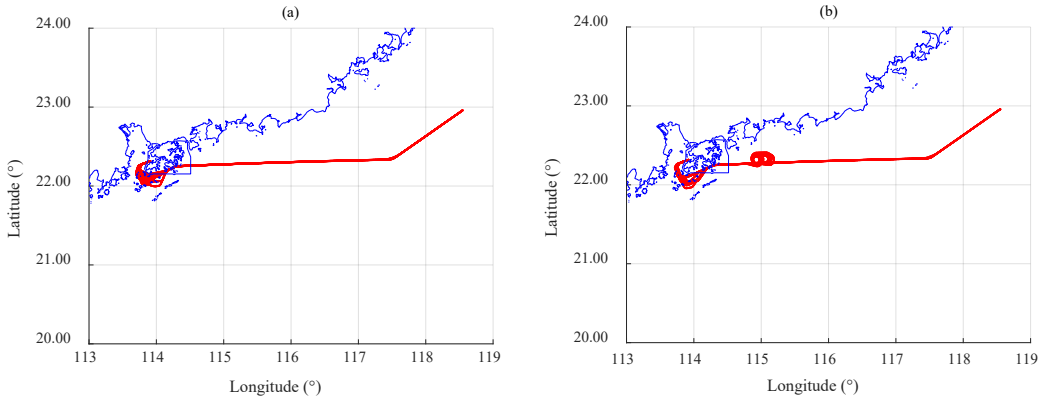

Figure 5. The selected flight trajectories from waypoints KADLO to VHHH: (a) without circling, (b) with circling.

4.2 Test environment and model parameter settings

The experiment is performed on a PC with Microsoft Windows 10 (CPU: 12th-Gen Intel(R) Core i9-12900KF at 3.20 GHz, RAM: 64.0 GB, GPU: NVIDIA GTX 3080). The CTGAN-based models were built using Keras and TensorFlow. The parameters of CTGAN architecture are based on our previous research [53]. The detailed training parameters are shown in **Table 2**.

Table 2. Training parameters for the C-CTGAN.

Parameter	Description	Value
Embedding dim.	Size of the random sample passed to the generator	128
Generator dim.	Size of the output samples for each one of the residuals	(256, 256)
Discriminator dim.	Size of the output samples for each one of the discriminator layers	(256, 256)
Generator lr	Learning rate for the generator	$5 \cdot 10^{-4}$
Generator decay	Generator weight decay for the Adam optimizer	$1 \cdot 10^{-6}$
Discriminator lr	Learning rate for the discriminator	$5 \cdot 10^{-4}$
Discriminator decay	Discriminator weight decay for the Adam optimizer	$1 \cdot 10^{-6}$
Batch size	Number of data samples to process in each step	750
Step ratio	Number of discriminator updates to perform for each generator update	1
Epochs	Number of training epochs	50,000
Pac	Number of samples to group together when applying the discriminator	10

For the LSTM-based models, we use MATLAB to run the prediction algorithms for comparison purposes, and the proportions of the training set and test set are also 0.8 and 0.2, respectively. Previous studies [13] have shown that the accuracy of BP neural networks for medium- and long-term aircraft TP is far inferior to that of LSTM neural networks. Therefore, we selected four commonly used neural networks—LSTM, CNN-LSTM, CNN-LSTM-Attention, and CNN-BiLSTM—for comparison analysis. The remaining parameters were set as follows: 100 hidden layers, 2000 epochs, a learning rate of 0.9, and a learning rate reduction period of 1600 for LSTM; a 2-layer convolutional network and a convolution kernel size of [3,1], where the first convolutional layer has 32 feature maps, the second layer has 64 feature maps, and both apply ReLU activation, 100 hidden layers, 1000 epochs, a minibatch size of 200, a learning rate of 0.005, a learning rate reduction factor of 0.9, and a learning rate reduction

period of 800 for CNN-LSTM; the same as CNN-LSTM, with 100 hidden layers, 80 epochs, a minibatch size of 100, a learning rate of 0.005, a learning rate reduction factor of 0.9, and a validation frequency of 10 for CNN-LSTM-Attention. The parameters of CNN-BiLSTM are the same as those of CNN-LSTM, except that the BiLSTM layer is used instead of the LSTM layer, the learning rate is changed to 0.001, and the learning rate reduction factor is set to 0.1.

4.3 Model testing and analysis

4.3.1 C-CTGAN model testing and analysis

First, we cluster the two sample datasets according to the method outlined in section 3.2. Based on equations (1) and (2), the optimal number of clusters for both datasets is 6. The clustered trajectories are shown in **Figure 6**, where different colors represent different clusters.

Next, following the method described in section 3.3, we train the model using the two sample datasets and calculate their respective MMD and MED based on equations (5) and (7). For the first group of samples shown in **Figure 6**(a), the MMD is 0.0415, and the MED is 0.0049. For the second group of examples shown in **Figure 6**(b), the MMD is 0.0422, and the MED is 0.0069. Small values of MMD and MED indicate high accuracy in generating trajectories. When they are both equal to 0, the generated data are identical to the actual data. The parameter calculation results show that the predicted trajectory can simulate the actual trajectory well and is characterized by high prediction accuracy.

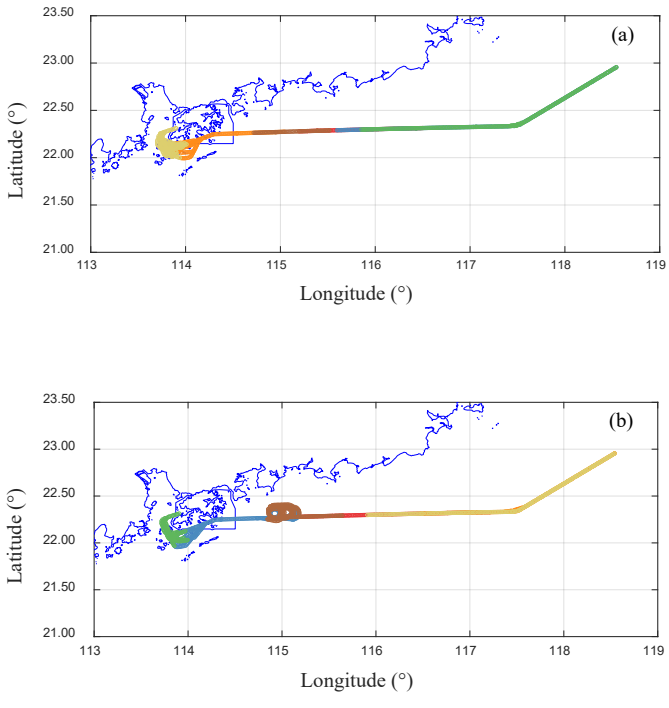

Figure 6. The clustered flight trajectories from waypoints KADLO to VHHH: (a) without circling, (b) with circling.

To intuitively observe the similarity between the model-generated data and the real data distribution, we calculate the probability density distribution with the kernel function as the PDF for all parameters of the two sets of data. This probability density distribution difference intuitively reflects the distance between the generated and real data distributions. **Figure 7** shows the PDFs of the time, latitude, longitude, geometric height, ground speed, and track angle variables for the two datasets.

We found that there is a good fit between the blue line representing the real data distribution and the red line representing the generated data distribution except for the TA parameter. This result indicates that the real and generated data have a high degree of similarity in their distributions. This observation aligns well with the calculated evaluation metrics.

Comparing **Figure 7**(b) to **Figure 7**(a), we notice a significantly poorer fit in the case of **Figure 7**(b), indicating that trajectories with flight circling can impact the accuracy of the C-CTGAN model. Regarding the TA parameter, there is a situation in which the fit is poor at a particular position, regardless of the presence of flight circling data. This is because the trajectory angle values undergo a bidirectional mutation from 360 degrees to 0 degrees, significantly affecting the output of the C-CTGAN model, which relies on probability density for prediction. Fortunately, in long-term trajectory prediction, parameters such as Lat, Lon, GH, and GS are core parameters, while TA is an auxiliary parameter that can be derived from adjacent trajectory points. As a result, this approach has a very limited impact on trajectory prediction.

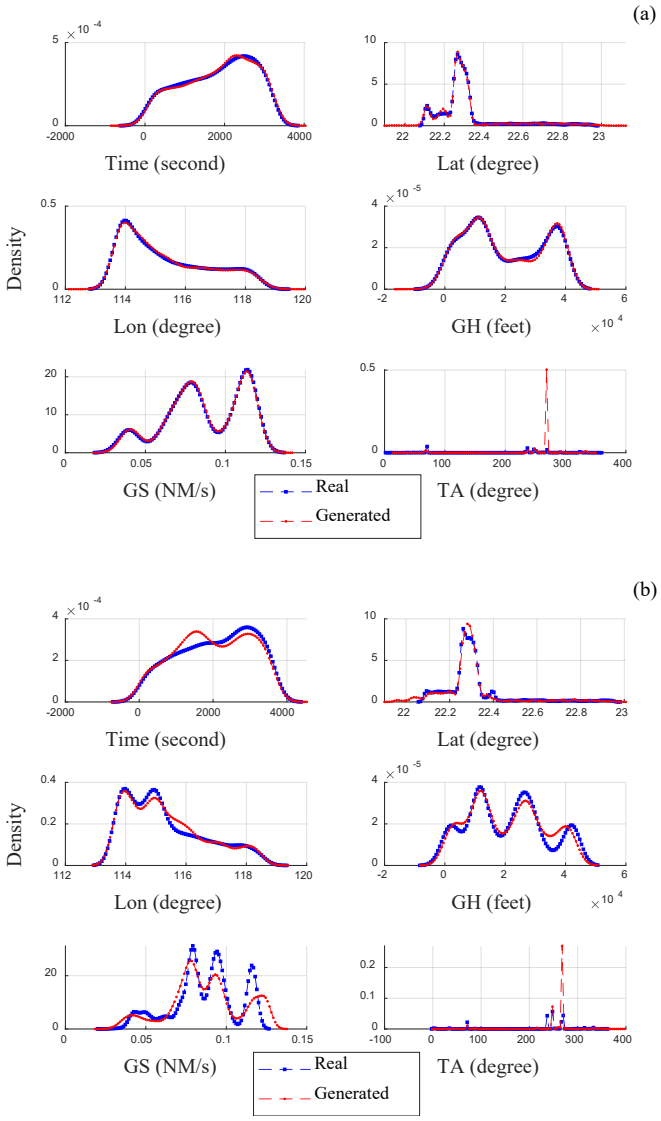

Figure 7. Distribution comparison between the generated and real data: (a) Flight trajectories without circling, (b) flight trajectories with circling.

The experimental results demonstrate that the trajectories generated by the C-CTGAN model exhibit a high degree of similarity with the actual trajectories. They can effectively approximate actual data, particularly for simple trajectory patterns, for which the model displays a high degree of fitting accuracy.

4.3.2 Comparison experiments and analyses

To further demonstrate the trajectory prediction accuracy of the C-CTGAN, we conducted comparative experiments with two sets of sample data using the LSTM, CNN-LSTM, CNN-LSTM-Attention, CNN-BiLSTM, and original CTGAN models. Since the MAE provides a direct indication of the degree of conformity between the generated and actual trajectories, in the comparative experiments, we calculated the MAE for each of the above methods to analyze and compare the accuracy of different trajectory

prediction methods.

(1) Determination of the time span for the LSTM-based models

Previous research has shown that due to the inherent algorithmic limitations of LSTM models, their prediction accuracy decreases rapidly with increasing prediction time [21]. To address this issue, we conducted experiments using LSTM to predict trajectories with time spans of 1, 3, 5, 7, 10, and 15 minutes, as shown in **Figure 8**. This illustrates the trend of the MAE values for various relevant parameters; notably, the error rapidly increases with the extension of the prediction time span after z score standardization, consistent with previous research findings. Since medium- to long-term prediction refers to predictions with a time span of more than 10 minutes, 10 minutes is selected for the LSTM prediction models for comparative analysis.

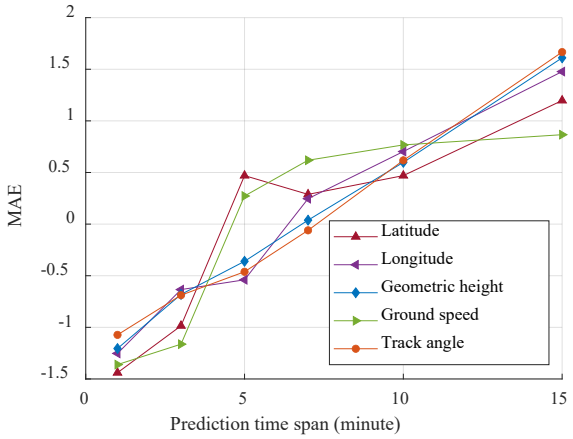

Figure 8. The relationship between the prediction error of the LSTM model and the prediction time span. The ordinate represents the MAEs of different parameters.

(2) Comparison of trajectory prediction models

Table 3 and **Table 4** present the MAEs for TP using the models based on the data with and without circling, respectively. In these tables, Lat represents the latitude, Lon represents the longitude, GH represents the geometric height, GS represents the ground speed, and TA represents the track angle.

Table 3. MAE comparison among the LSTM, CNN-LSTM, CNN-LSTM-Attention, CNN-BiLSTM, original CTGAN, and C-CTGAN models based on flight trajectories without circling.

Model	**MAE**				
	Lat (°)	**Lon (°)**	**GH (ft)**	**GS (NM/s)**	**TA (°)**

LSTM	0.0116	0.0357	684.3974	0.0045	5.0389
CNN-LSTM	**0.0093**	**0.0340**	**789.4817**	**0.0038**	**8.7183**
CNN-LSTM-Attention	0.0101	0.0580	814.9366	0.0044	8.7585
CNN-BiLSTM	0.0106	0.0346	700.9499	0.0047	7.6823
Original CTGAN	**0.0043**	**0.0297**	**266.3775**	**0.0023**	**3.2114**
C-CTGAN	**0.0028**	**0.0289**	**204.6852**	**0.0006**	**2.0433**

Table 4. MAE comparison among the LSTM, CNN-LSTM, CNN-LSTM-Attention, CNN-BiLSTM, original CTGAN, and C-CTGAN models based on the flight trajectories with circling.

Model	**MAE**				
	Lat (°)	**Lon (°)**	**GH (ft)**	**GS (NM/s)**	**TA (°)**
LSTM	0.0216	0.0782	2025.96	0.0054	15.9191
CNN-LSTM	**0.0213**	**0.0929**	**1449.65**	**0.0035**	**13.8093**
CNN-LSTM-Attention	0.0339	0.2300	3250.20	0.0068	31.2471
CNN-BiLSTM	0.0299	0.1810	2972.91	0.0072	27.3524
Original CTGAN	**0.0230**	**0.1164**	**1460.80**	**0.0041**	**17.5870**
C-CTGAN	**0.0183**	**0.0709**	**993.61**	**0.0034**	**18.5150**

1) Comparative analysis of LSTM-based models

Table 3 and **Table 4** indicate that the CNN-LSTM model performs the best. Based on the sample without circling data, the MAEs of various parameters (Lat, Lon, GH, GS, TA) for the LSTM-based models are generally small, and the MAEs for the same parameters are similar overall. This indicates that all models yield a high level of TP accuracy. Based on the sample with circling data, the MAEs of trajectory-related parameters for the LSTM-based models increase significantly, and the differences in the MAEs for the same parameter are notably different among different models. This indicates a significant difference in TP accuracy for different models. The results also suggest that different models have varying degrees of adaptability to the sample data.

2) Comparative analysis between the C-CTGAN, original CTGAN and CNN-LSTM models

Since CNN-LSTM is the best-performing model of the LSTM family, we use it as the representative to perform a comparative analysis with other models. The data in **Table 3** and **Table 4** show that the MAEs of the trajectory-related parameters, except TA, in the C-CTGAN model are smaller than the corresponding values of the original CTGAN and CNN-LSTM models. The C-CTGAN model

significantly outperforms the original CTGAN and CNN-LSTM models in terms of key TP parameters. For the data without circling, the TP accuracy of the C-CTGAN is considerably better than that for the data with circling.

When using the sample dataset without circling trajectories, the C-CTGAN model reduces the MAEs of core trajectory parameters such as latitude, longitude, geometric altitude, and ground speed by 69.89%, 15.00%, 74.07%, and 84.21%, respectively, compared to those of the CNN-LSTM model. Compared to those of the original CTGAN, the MAEs of the core parameters are reduced by 20.43%, 39.09%, 31.98%, and 17.07%, respectively.

When using the sample dataset with circling trajectories, the C-CTGAN model shows significant improvements in predicting parameters such as latitude, longitude, geometric altitude, and ground speed, with MAE reductions of 14.08%, 23.68%, 31.46%, and 2.86%, respectively, when compared to those of the CNN-LSTM model. Compared to those of the original CTGAN, the MAEs of the core parameters are reduced by 34.88%, 2.69%, 23.16%, and 73.91%, respectively.

Table 3 and **Table 4** also indicate that for noncircling sample data, the TP accuracy of the original CTGAN is significantly better than that of the CNN-LSTM model. However, in the case of the circling dataset, the TP accuracies of the two models are essentially equivalent.

(3) Typical trajectory prediction visualization

To compare the medium- and long-term TP effects of different models in an intuitive manner, we select flight CPA405 (trajectory without circling) and flight ANA859 (trajectory with circling) as examples to examine the differences between the TP parameters among the CNN-LSTM, CTGAN, and C-CTGAN models. We smoothed the trajectory parameters based on the CTGAN and the C-CTGAN models for easy visualization.

Figure 9 displays the prediction results of the original CTGAN, C-CTGAN, and CNN-LSTM models based on the CPA405 dataset. According to the visualized experimental results, the C-CTGAN model can fit the actual trajectories and predict the overall trajectory trend effectively. Compared to those of the C-CTGAN model, the predictions of the original CTGAN and CNN-LSTM models roughly align with the direction of the actual trajectory. However, there is still a particular gap in fitting accuracy. This result is consistent with the results in **Table 3**. Compared to the original CTGAN, C-CTGAN exhibits a smaller average distance deviation, better meeting the accuracy requirements for TP.

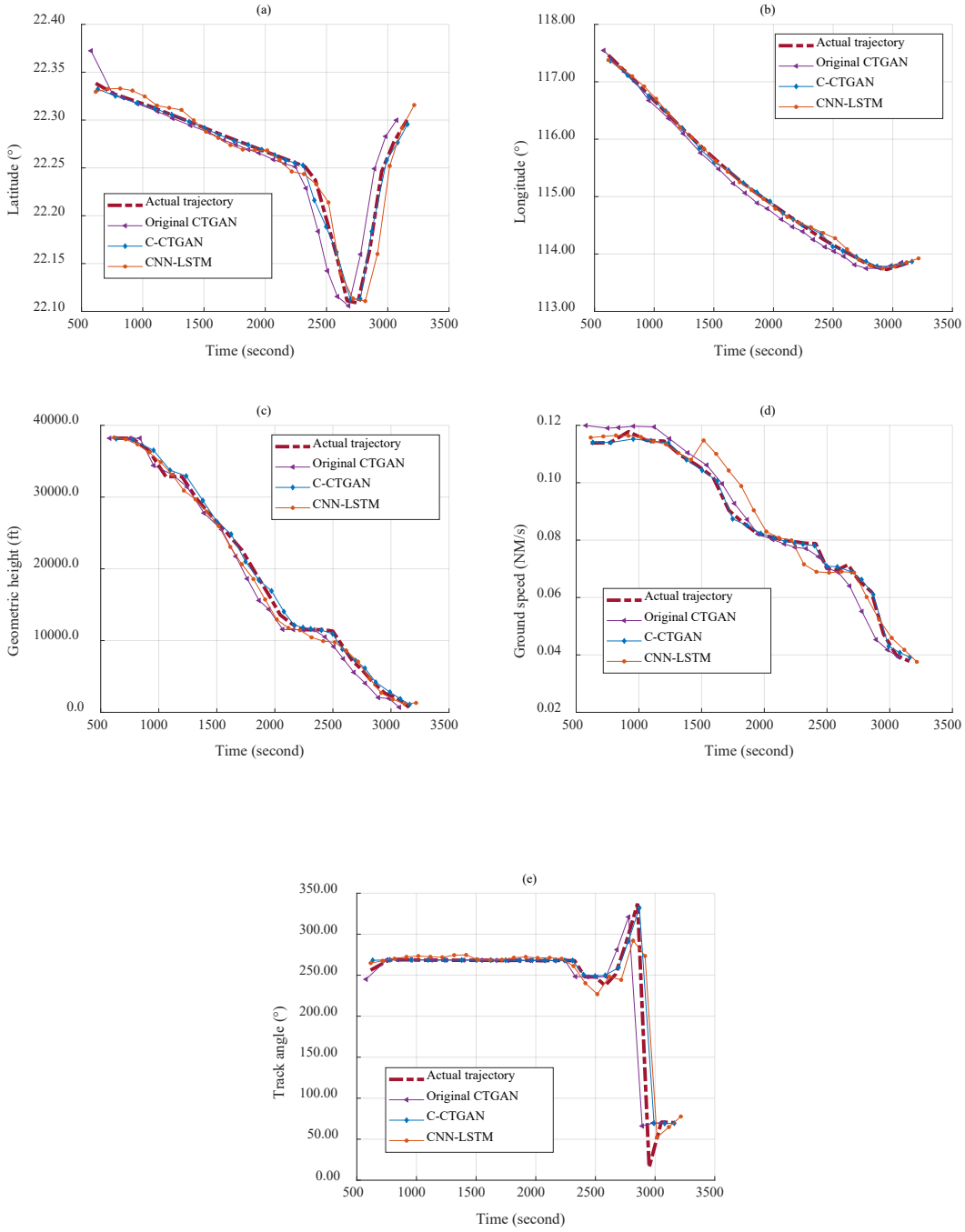

Figure 9. Comparison of the prediction results of the CNN-LSTM, CTGAN, and C-CTGAN models for flight CPA405 (trajectory without circling).

Using the actual flight trajectory of ANA859 as a reference, we examined the differences in the predicted trajectory parameters among the original CTGAN, C-CTGAN, and CNN-LSTM models, as shown in **Figure 10**. With the introduction of clustering as an additional step, C-CTGAN is better equipped to handle parameter fluctuations caused by circling and learn the data distribution compared to other models. Overall, the C-CTGAN model achieves the highest prediction accuracy, which is consistent

with the results in **Table 4**.

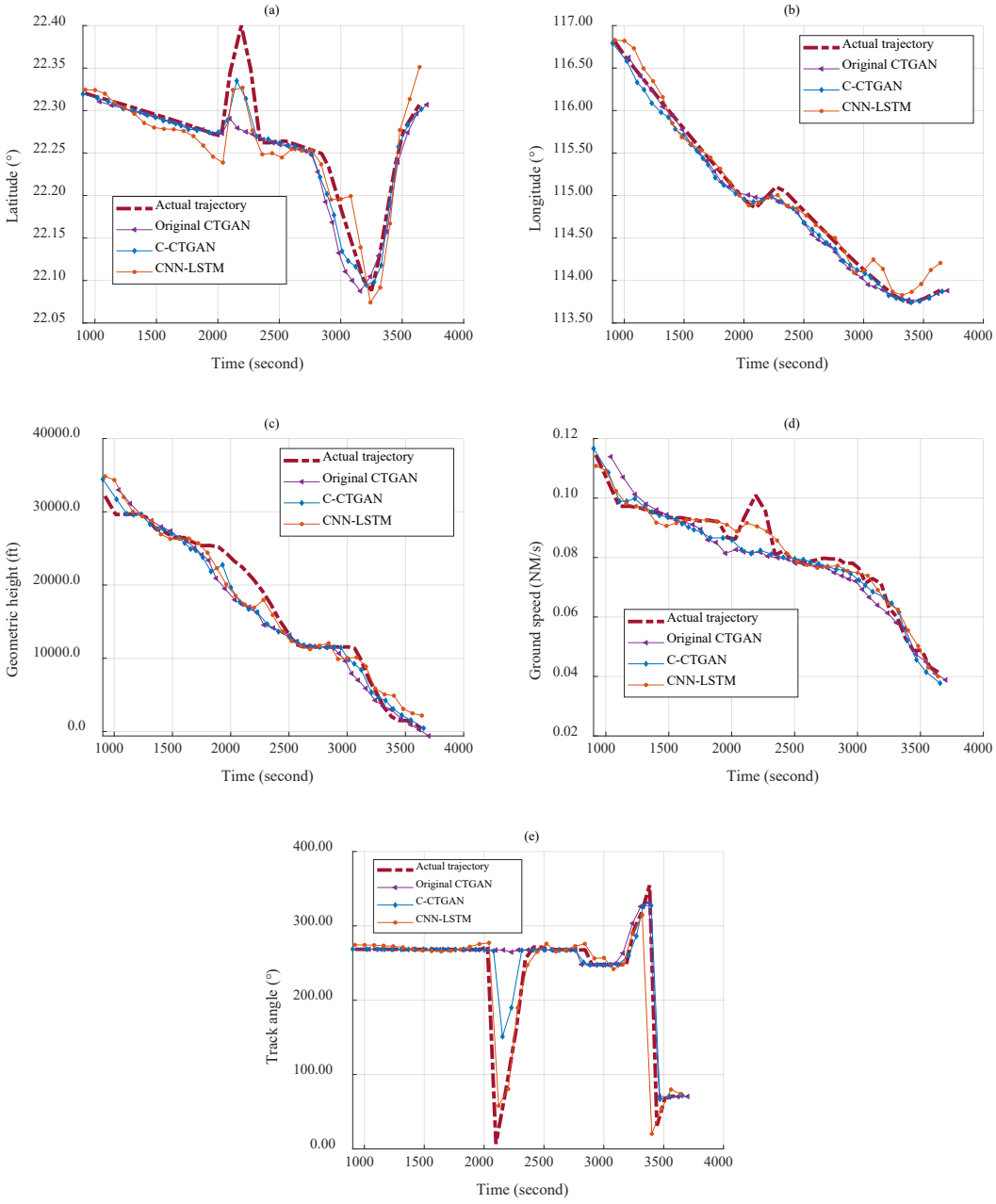

Figure 10. Comparison of the prediction results of the CNN-LSTM, CTGAN, and C-CTGAN models for flight image ANA859 (trajectory with circling).

By comparing **Figure 9** and **Figure 10**, it can be observed that the predicted trajectory parameters in **Figure 9** generally better fit the real trajectory parameters than those in **Figure 10**. This finding aligns with the findings in **Table 3** and **Table 4**, indicating that the circling trajectory data have a significant impact on the TP accuracy.

5. Discussion

5.1 Methodological innovations

Our study introduces the C-CTGAN model, a novel approach combining K-medoids clustering with CTGAN, to address the limitations in existing TP models. This integrated approach enhances the TP accuracy, demonstrating the potential of merging GANs with clustering for improved performance. Comparative experiments revealed the superior performance of the C-CTGAN over established models, particularly LSTM-based models. Reduced MAEs for trajectory parameters underscore the model's effectiveness, offering a promising alternative for medium- and long-term TP.

5.2 Contributions and limitations

This study advances air traffic management by presenting a robust approach to 4D aircraft trajectory prediction. Tailored for ADS-B data, C-CTGAN's incorporation of trajectory clustering provides an avenue for hybrid modeling, contributing to a new research direction. The high accuracy of the C-CTGAN model indicates the model’s potential for applications in ATM. Improved TP directly contributes to enhanced airspace awareness, more effective flow management, and increased operational efficiency, addressing the challenges posed by increases in air traffic.

6. Conclusions and future work

This paper introduces a 4D TP model based on K-medoids clustering and the CTGAN framework, referred to as the C-CTGAN model. Compared to LSTM-based models and the original CTGAN, this method delivers higher accuracy for medium- to long-term TP. It is a feasible machine learning model that can support situational airspace awareness, flight traffic management, airspace utilization, and efficient aviation operations.

The model's accuracy is evaluated from the perspectives of data distribution and trajectory distance using MMD and MED evaluation metrics. Although MMD and MED can be used to evaluate GAN-based models well, they are unsuitable for assessing LSTM-based models, and the MAE is suitable for both models. Therefore, the MAE is used as an evaluation metric to compare the C-CTGAN model with the LSTM, CNN-LSTM, CNN-LSTM-Attention, CNN-BiLSTM, and original CTGAN models. The results show that our model significantly outperforms the LSTM-based and original CTGAN models in predicting mid- to long-term trajectories. Clustering proves to be effective for enhancing the TP precision

of the CTGAN model.

When using a sample dataset without circling trajectories, the C-CTGAN model reduces the MAEs of core trajectory parameters such as latitude, longitude, geometric altitude, and ground speed by 69.89%, 15.00%, 74.07%, and 84.21%, respectively, compared to those of the best-performing CNN-LSTM model in the LSTM family. Compared to those of the original CTGAN, the MAEs of the proposed method are 20.43%, 39.09%, 31.98%, and 17.07% lower in latitude, longitude, geometric altitude, and ground speed, respectively.

When using a sample dataset with circling trajectories, the C-CTGAN model shows significant improvements in predicting parameters such as latitude, longitude, geometric altitude, and ground speed, with MAE reductions of 14.08%, 23.68%, 31.46%, and 2.86%, respectively, compared to those of the CNN-LSTM model. Compared to those of the original CTGAN, the MAEs of the proposed method are reduced by 34.88%, 2.69%, 23.16%, and 73.91% in terms of latitude, longitude, geometric altitude, and ground speed, respectively.

In the comparison experiments, the LSTM-based models yields prediction results with a time span of 10 minutes. As the prediction time span increases, the advantages of the C-CTGAN model over the LSTM-based models become more apparent for TP. Comparative experiments also indicate that the circling trajectory segment significantly affects the TP accuracy of the models under consideration.

Acknowledgments: The grant support from the Hong Kong Research Grants Council (RGC) General Research Fund (GRF) (15212622/B-Q94L) is greatly acknowledged.

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

References

[1] R. Sáez, X. Prats, T. Polishchuk, and V. Polishchuk, “Traffic synchronization in terminal airspace to enable continuous descent operations in trombone sequencing and merging procedures: An implementation study for Frankfurt airport,” *Transportation Research Part C: Emerging Technologies*, vol. 121, 2020, p. 102875. https://doi.org/10.1016/j.trc.2020.102875

[2] X. Guan, X. Zhang, D. Han, Y. Zhu, J. Lv, and J. Su, “A strategic flight conflict avoidance approach based on a memetic algorithm,” *Chinese Journal of Aeronautics*, vol. 27, no. 1, 2014, pp. 93–101. https://doi.org/10.1016/j.cja.2013.12.002

[3] G. Chatterji, “Short-term trajectory prediction methods,” AIAA Guidance, Navigation, and Control Conference and Exhibit, 1999, p. 4233. https://doi.org/10.2514/6.1999-4233

[4] X. Guan, X. Zhang, J. Wei, I. Hwang, Y. Zhu, and K. Cai, “A strategic conflict avoidance approach based on cooperative coevolutionary with the dynamic grouping strategy,” *International Journal of Systems Science*, vol. 47, no. 9, 2016, pp. 1995–2008. https://doi.org/10.1080/00207721.2014.966282

[5] H. Shafienya and A. C. Regan, “4D flight trajectory prediction using a hybrid Deep Learning prediction method based on ADS-B technology: A case study of Hartsfield–Jackson Atlanta International Airport (ATL),” *Transportation Research Part C: Emerging Technologies*, vol. 144, 2022, p. 103878. https://doi.org/10.1016/j.trc.2022.103878

[6] W. Zeng, X. Chu, Z. Xu, Y. Liu, and Z. Quan, “Aircraft 4D Trajectory Prediction in Civil Aviation: A Review,” *Aerospace*, vol. 9, no. 2, 2022, p. 91. https://doi.org/10.3390/aerospace9020091

[7] D. J. Brudnicki and D. B. Kirk, “Trajectory modeling for automated en route air traffic control (AERA),” presented at the Proceedings of 1995 American Control Conference-ACC’95, IEEE, 1995, pp. 3425–3429. https://doi.org/10.1109/ACC.1995.532247

[8] J. Garcia-Chico, R. Vivona, and K. Cate, “Characterizing intent maneuvers from operational data: Step towards trajectory prediction uncertainty estimation,” AIAA Guidance, Navigation and Control Conference and Exhibit, 2008, p. 6520. https://doi.org/10.2514/6.2008-6520

[9] D. Delahaye, S. Puechmorel, P. Tsiotras, and E. Féron, “Mathematical models for aircraft trajectory design: A survey,” *Air Traffic Management and Systems*, Springer, 2014, pp. 205–247.

[10] A. Harada, T. Ezaki, T. Wakayama, and K. Oka, “Air traffic efficiency analysis of airliner scheduled flights using collaborative actions for renovation of air traffic systems open data,” *Journal of Advanced Transportation*, vol. 2018, 2018. https://doi.org/10.1155/2018/2734763

[11] S. Ruiz, L. Guichard, N. Pilon, and K. Delcourte, “A new air traffic flow management user-driven

prioritisation process for low volume operator in constraint: Simulations and results," *Journal of Advanced Transportation*, vol. 2019, 2019.
https://doi.org/10.1155/2019/1208279
[12] E. Andrés, D. González-Arribas, M. Soler, M. Kamgarpour, M. Sanjurjo-Rivo, and J. Simarro, "Iterative graph deformation for aircraft trajectory planning considering ensemble forecasting of thunderstorms," *Transportation Research Part C: Emerging Technologies*, vol. 145, 2022, p. 103919.
https://doi.org/10.1016/j.trc.2022.103919
[13] J. Zhou, H. Zhang, W. Lyu, J. Wan, J. Zhang, and W. Song, "Hybrid 4-Dimensional Trajectory Prediction Model, Based on the Reconstruction of Prediction Time Span for Aircraft en Route," *Sustainability*, vol. 14, no. 7, 2022, p. 3862.
https://doi.org/10.3390/su14073862
[14] I. Dhief, S. Alam, N. Lilith, and C. C. Mean, "A machine learned go-around prediction model using pilot-in-the-loop simulations," *Transportation Research Part C: Emerging Technologies*, vol. 140, 2022, p. 103704.
https://doi.org/10.1016/j.trc.2022.103704
[15] S. QIAO, N. Han, X. ZHU, H. SHU, J. ZHENG, and C. YUAN, "A dynamic trajectory prediction algorithm based on Kalman filter," *Acta Electonica Sinica*, vol. 46, no. 2, 2018, p. 418.
https://doi.org/10.3969/j.issn.0372-2112.2018.02.022
[16] Z.-J. Wu, S. Tian, and L. Ma, "A 4D trajectory prediction model based on the BP neural network," *Journal of Intelligent Systems*, vol. 29, no. 1, 2020, pp. 1545–1557.
https://doi.org/10.1515/jisys-2019-0077
[17] De Leege A., van Paassen M., and Mulder M., "A machine learning approach to trajectory prediction," AIAA Guidance, Navigation, and Control (GNC) Conference, August 2013.
https://doi.org/10.2514/6.2013-4782
[18] Ghasemi Hamed M., Gianazza D., Serrurier M. and Durand N., "Statistical prediction of aircraft trajectory: Regression methods vs point-mass model," 10th USA/Europe Air Traffic Management Research and Development Seminar, June 2013.
[19] Tang X., Chen P., and Zhang Y., "4D trajectory estimation based on nominal flight profile extraction and airway meteorological forecast revision," *Aerospace Science and Techno*logy," vol. 45, 2015, pp. 387-397.
https://doi.org/10.1016/j.ast.2015.06.001
[20] Barratt, Shane T., Mykel J. Kochenderfer, and Stephen P. Boyd, "Learning probabilistic trajectory models of aircraft in terminal airspace from position data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, 2018, pp. 3536-3545.
https://doi.org/10.1109/TITS.2018.2877572
[21] Gallego, C. E. V., Comendador, V. F. G., Nieto, F. J. S., Imaz, G. O., and Valdés, R. M. A., "Analysis of air traffic control operational impact on aircraft vertical profiles supported by machine learning," *Transportation research part C: emerging technologies*, vol. 95, 2018, pp. 883-903.
https://doi.org/10.1016/j.trc.2018.03.017Get rights and content
[22] Z. Ma, M. Yao, T. Hong, and B. Li, "Aircraft surface trajectory prediction method based on LSTM with

attenuated memory window," Journal of Physics: Conference Series, IOP Publishing, 2019, p. 012003. https://doi.org/10.1088/1742-6596/1215/1/012003

[23] Z. Xu, W. Zeng, X. Chu, and P. Cao, "Multi-aircraft trajectory collaborative prediction based on social long short-term memory network," *Aerospace*, vol. 8, no. 4, 2021, p. 115. https://doi.org/10.3390/aerospace8040115

[24] W. Zeng, Z. Quan, Z. Zhao, C. Xie, and X. Lu, "A deep learning approach for aircraft trajectory prediction in terminal airspace," *IEEE Access*, vol. 8, 2020, pp. 151250–151266. https://doi.org/10.1109/ACCESS.2020.3016289

[25] Z. Zhao, W. Zeng, Z. Quan, M. Chen, and Z. Yang, "Aircraft trajectory prediction using deep long short-term memory networks," COTA (Chinese Overseas Transportation Association) International Conference of Transportation Professionals (CICTP) series, 2019.

[26] L. Lin, W. Li, H. Bi, and L. Qin, "Vehicle Trajectory Prediction Using LSTMs With Spatial–Temporal Attention Mechanisms," *IEEE Intelligent Transportation Systems Magazine*, vol. 14, no. 2, 2021, pp. 197–208. https://doi.org/10.1109/MITS.2021.3049404

[27] P. Jia, H. Chen, L. Zhang, and D. Han, "Attention-LSTM Based Prediction Model for Aircraft 4-D Trajectory," Scientific reports, 2022, 12(1): 15533.

[28] Y. Wu, H. Yu, J. Du, B. Liu, and W. Yu, "An Aircraft Trajectory Prediction Method Based on Trajectory Clustering and a Spatiotemporal Feature Network," *Electronics*, vol. 11, no. 21, 2022, p. 3453. https://doi.org/10.3390/electronics11213453

[29] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," *Advances in Neural Information Processing Systems*, vol. 29, 2016. https://doi.org/10.48550/arXiv.1609.02612

[30] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," International conference on machine learning, PMLR, 2016, pp. 1060–1069. https://doi.org/10.48550/arXiv.1605.05396

[31] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," arXiv preprint arXiv:1809.11096, 2018. https://doi.org/10.48550/arXiv.1809.11096

[32] W. Ding, W. Wang, and D. Zhao, "A multi-vehicle trajectories generation for vehicle-to-vehicle encounters," 2019 IEEE International Conference on Robotics and Automation (ICRA), 2019. https://doi.org/10.48550/arXiv.1809.05680

[33] G. Jarry, N. Couellan, and D. Delahaye, "On the use of generative adversarial networks for aircraft trajectory generation and atypical approach detection," ENRI International Workshop on ATM/CNS, Springer, 2019, pp. 227–243.

[34] Y. Pang and Y. Liu, "Conditional generative adversarial networks (CGAN) for aircraft trajectory prediction considering weather effects," AIAA Scitech 2020 Forum, 2020, p. 1853. https://doi.org/10.2514/6.2020-1853

[35] X. Wu, H. Yang, H. Chen, Q. Hu, and H. Hu, "Long-term 4D trajectory prediction using generative

adversarial networks," *Transportation Research Part C: Emerging Technologies*, vol. 136, 2022, p. 103554. https://doi.org/10.1016/j.trc.2022.103554

[36] C. An, J. Sun, Y. Wang, and Q. Wei, "A K-means Improved CTGAN Oversampling Method for Data Imbalance Problem," 2021 IEEE 21st International Conference on Software Quality, Reliability and Security (QRS), IEEE, 2021, pp. 883–887. https://doi.org/10.1109/QRS54544.2021.00097

[37] A. A. Alqarni and E.-S. M. El-Alfy, "Improving Intrusion Detection for Imbalanced Network Traffic using Generative Deep Learning," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 4, 2022. https://doi.org/10.14569/IJACSA.2022.01304109

[38] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling tabular data using conditional gan," *Advances in Neural Information Processing Systems*, vol. 32, 2019. https://doi.org/10.48550/arXiv.1907.00503

[39] L. Xu and K. Veeramachaneni, "Synthesizing tabular data using generative adversarial networks," arXiv preprint arXiv:1811.11264, 2018. https://doi.org/10.48550/arXiv.1811.11264

[40] M.-P. Cote, B. Hartman, O. Mercier, J. Meyers, J. Cummings, and E. Harmon, "Synthesizing property & casualty ratemaking datasets using generative adversarial networks," arXiv preprint arXiv:2008.06110, 2020. https://doi.org/10.48550/arXiv.2008.06110

[41] J. S. LEE and O. LEE, "CTGAN VS TGAN? Which one is more suitable for generating synthetic EEG data," *Journal of Theoretical and Applied Information Technology*, vol. 99, no. 10, 2021.

[42] J. Park, J. Jeong, and Y. Park, "Ship trajectory prediction based on bi-LSTM using spectral-clustered AIS data," *Journal of Marine Science and Engineering*, vol. 9, no. 9, 2021, p. 1037. https://doi.org/10.3390/jmse9091037

[43] S. J. Corrado, T. G. Puranik, O. P. Fischer, and D. N. Mavris, "A clustering-based quantitative analysis of the interdependent relationship between spatial and energy anomalies in ADS-B trajectory data," *Transportation Research Part C: Emerging Technologies*, vol. 131, 2021, p. 103331. https://doi.org/10.1016/j.trc.2021.103331

[44] S. Zhang, L. Wang, M. Zhu, S. Chen, H. Zhang, and Z. Zeng, "A bi-directional lstm ship trajectory prediction method based on attention mechanism," 2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), IEEE, 2021, pp. 1987–1993. https://doi.org/10.1109/IAEAC50856.2021.9391059

[45] P. Arora and S. Varshney, "Analysis of k-means and k-medoids algorithm for big data," *Procedia Computer Science*, vol. 78, 2016, pp. 507–512. https://doi.org/10.1016/j.procs.2016.02.095

[46] R. A. Nugraha, H. F. Pardede, and A. Subekti, "Oversampling based on generative adversarial networks to overcome imbalance data in predicting fraud insurance claim: 10.48129/kjs. splml. 19119," *Kuwait Journal of Science*, 2022.

https://doi.org/10.48129/kjs.splml.19119
[47] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” International conference on machine learning, PMLR, 2015, pp. 448–456.
https://doi.org/10.48550/arXiv.1502.03167
[48] E. Jang, S. Gu, and B. Poole, “Categorical reparameterization with gumbel-softmax,” arXiv preprint arXiv:1611.01144, 2016.
https://doi.org/10.48550/arXiv.1611.01144
[49] Z. Lin, A. Khetan, G. Fanti, and S. Oh, “Pacgan: The power of two samples in generative adversarial networks,” *Advances in Neural Information Processing Systems*, vol. 31, 2018.
https://doi.org/10.48550/arXiv.1712.04086
[50] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, no. 1, 2014, pp. 1929–1958.
[51] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, “Improved training of wasserstein gans,” *Advances in Neural Information Processing Systems*, vol. 30, 2017.
https://doi.org/10.48550/arXiv.1704.00028
[52] A. Borji, “Pros and cons of gan evaluation measures,” *Computer Vision and Image Understanding*, vol. 179, 2019, pp. 41–65.
https://doi.org/10.1016/j.cviu.2018.10.009
[53] H. ZHANG and Z. LIU, “Four-Dimensional Aircraft Trajectory Prediction Based on Generative Deep Learning,” *Journal of Aerospace Information Systems*, vol. 0, 2024, pp. 1-12.
https://doi.org/10.2514/1.I011333