algorithms

MDPI

Article

A Deep Gaussian Process-Based Flight Trajectory Prediction Approach and Its Application on Conflict Detection

Zhengmao Chen, Dongyue Guo and Yi Lin *

College of Computer Science, Sichuan University, Chengdu 610065, China; chenzhengmao@scu.edu.cn (Z.C.); dongyueguo@stu.scu.edu.cn (D.G.)

* Correspondence: yilin@scu.edu.cn; Tel.: +86-135-4790-3121

Received: 18 September 2020; Accepted: 10 November 2020; Published: 11 November 2020

Abstract: In this work, a deep Gaussian process (DGP) based framework is proposed to improve the accuracy of predicting flight trajectory in air traffic research, which is further applied to implement a probabilistic conflict detection algorithm. The Gaussian distribution is applied to serve as the probabilistic representation for illustrating the transition patterns of the flight trajectory, based on which a stochastic process is generated to build the temporal correlations among flight positions, i.e., Gaussian process (GP). Furthermore, to deal with the flight maneuverability of performing controller's instructions, a hierarchical neural network architecture is proposed to improve the modeling representation for nonlinear features. Thanks to the intrinsic mechanism of the GP regression, the DGP model has the ability of predicting both the deterministic nominal flight trajectory (NFT) and its confidence interval (CI), denoting by the mean and standard deviation of the prediction sequence, respectively. The CI subjects to a Gaussian distribution, which lays the data foundation of the probabilistic conflict detection. Experimental results on real data show that the proposed trajectory prediction approach achieves higher prediction accuracy compared to other baselines. Moreover, the conflict detection approach is also validated by a obtaining lower false alarm and more prewarning time.

Keywords: confidence interval; conflict detection; deep Gaussian process; nominal flight trajectory; neural network; trajectory prediction

1. Introduction

The trajectory prediction (TP), as a core technique in air traffic studies, has been attracting more and more attention from all over the world. It is the foundation of performing many air traffic applications, such as traffic flow prediction and conflict detection [1,2]. TP aims at predicting the flight trajectory in the near future based on its flight plan and motion patterns. As a part of our previous study, it was also applied to check the conformance of the flight trajectory [3], which ensures the safety of the flight operation. Machine learning-based algorithms were proposed to cope with the high stochasticity of the flight environment [4–6], typically a sequential regression task. In general, existing approaches can be summarized as the following categories:

(a) Kinematics and dynamics-based approaches (KDAs): this type of approach divides the flight operation into different phases based on its flight profile, typically climb, cruise and descent [7,8]. Considering the flight dynamics and aircraft performance constraints, several kinematics equations are built to illustrate the flight transition patterns for each phase [9], which are further solved to predict the flight trajectory. However, the KDA approach highly depends on the phase definition and fails to cope with the influence of the complicated and time-varying flight environment on kinematic and

dynamic characteristics [10]. Consequently, the accuracy and stability of the prediction results failed to meet the requirements of air traffic research.

(b) Filter-based approaches and its variations: the Kalman filter (KF) was first proposed to build the motion state transitions between two consecutive prediction instants [11]. Meanwhile, the correction rules were also proposed to build more accurate transition equations after receiving real-time observations (flight positions) [12,13], which is expected to improve the finally prediction performance. In succession, some KF-based variations were proposed to improve the applicability for dealing with flight trajectories with heavy maneuverability [12–15], such as climb, acceleration. Unfortunately, the representation of the transition patterns among flight motion states is always an obstacle for dealing with the drastic changes of the flight operation.

(c) Machine learning-based approaches: the core idea of this type of approach is mining frequent transition patterns from historical trajectories [5,16], which is further used to build the trajectory patterns of the predicted flights. In general, historical trajectories of a certain flight are proven to be feasible, safe and effective [4,17]. Therefore, based on the learned transition patterns, a machine learning-based approach can obtain higher prediction accuracy than that of other hand-crafted models [18]. On one hand, some well-designed probabilistic model-based machine learning approaches were proposed to predict flight trajectory, such as the hidden Markov model [19,20], Gaussian mixture model [21]. Gaussian process was also studied to check the conformance of flight trajectory in [22]. On the other hand, neural networks were also developed to predict flight trajectory by fitting the data distribution (i.e., flight transition patterns) from training samples automatically, such as deep neural networks (DNNs) [23,24], recurrent neural networks (RNNs) [7,25,26], and long short-term memory block [27–29]. The machine learning-based approaches have recently been the most popular ones and obtained promising performance for many applications.

It is clear that the learning mechanism is indispensable for dealing with the stochastic flight environment. In this work, we mainly focus on proposing a novel trajectory prediction approach that is able to illustrate the positional distribution integrally. Learning from existing state-of-the-art approaches, a deep Gaussian process (DGP)-based framework is proposed to achieve the trajectory prediction, and further to implement a probabilistic based short-term conflict detection (STCD) algorithm. A stochastic probabilistic process is capable of building the temporal dependencies of flight trajectory by fitting a probabilistic distribution at each prediction instant. That is to say, a probabilistic process is the joint distribution of sequential probabilistic distributions with respect to the prediction instant [30,31]. Based on the specificities of the Gaussian distribution, the mean and standard deviation of the prediction serve as the predicted nominal flight trajectory (NFT) and its confidence interval (CI), respectively. The hierarchical architecture of the deep neural network (DNN) is introduced to improve the capability on modeling nonlinear features of the flight trajectory when performing the controller's instructions, such as climb, speed regulating, etc. According to a previous work [32], a Gaussian distribution is suitable for illustrating the trajectory positions by analyzing a huge number of real flight trajectories. Therefore, combining the probabilistic process and Gaussian distribution, the Gaussian process (GP) is formulated to build the transition patterns of flight trajectory. The proposed DGP model is an improved DNN architecture, in which GPs are applied to compute the nonlinear mappings between neurons [33,34]. By training with historical trajectories, the parameters of the DGP model can be optimized for flight prediction, in which the flight patterns are learned in an implicit manner.

With the proposed techniques, the flight trajectory prediction for four typical operations, including cruise, heading regulating, speed regulating (acceleration/deceleration) and altitude regulating (climb/descent), are integrated into a single framework. The framework of the DGP-based trajectory prediction is sketched in Figure 1.

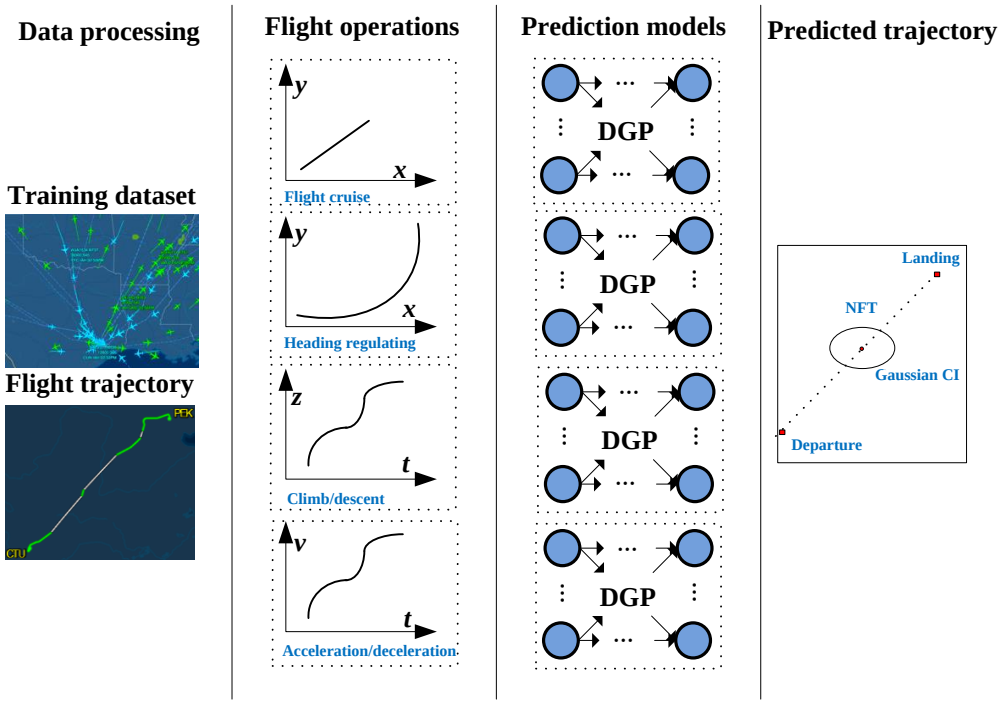

Figure 1. The framework of the proposed trajectory prediction approach. DPG: deep Gaussian process (DGP); NFT: nominal flight trajectory (NFT); CI: confidence interval (CI).

Historical trajectories are applied to optimize the parameters of the DGP models, in which different components of flight trajectory are designed to fit the model for different flight operations. Basically, the positional components are applied to build the transition patterns of the flight cruising and turn operation. Similarly, the altitude and velocity components serve as the motivation for predicting the flight trajectory when regulating flight altitude and speed, respectively, which generates a two-stage DGP solution. The information of the flight plan, including the waypoint sequence, cruise speed and altitude, are also extracted to build the constraints of the proposed DGP model for selecting different training samples and components and further learning different trajectory patterns. Based on the proposed DGP trajectory prediction approach, a probabilistic conflict detection approach is implemented pair-wisely, in which the Monte Carlo method is applied to simplify the computation by sampling flight positions from the predicted DGP models. The proposed trajectory prediction models and conflict detection algorithm are finally validated on real operating data. In short, the main contributions of this work can be summarized as follows:

(a) A DGP framework built with a Gaussian process and deep neural network is proposed to address the flight trajectory prediction issue in air traffic study. The stochastic probabilistic process is applied to build the temporal correlations of the flight trajectory, which greatly improve the modeling accuracy between predicting instants.

(b) The deep architecture is proposed to improve the feature representations by using GPs as the nonlinear activation. The machine learning mechanism is applied to learn frequent transition patterns from historical trajectory and further support the prediction of time-varying specificities. The proposed DGP model has the advantages of both the Gaussian process and DNN block, which improves the prediction accuracy and lays the foundation for probabilistic conflict detection.

(c) Trajectory prediction for different flight operations are considered in the proposed framework, which simplifies the TP task in air traffic study. The proposed approaches for regulating flight altitude and speed provide a solution of predicting the flight trajectory with high maneuverability, which is able to improve the TP performance for the whole flight process.

(d) The proposed approach not only predicts a deterministic flight trajectory (NFT), but also estimates its CIs at different predicting instants, which greatly improves the applicability of the prediction results. Based on the GP characteristics, the positional distribution of flight trajectory at a different instant subjects to a Gaussian distribution.

(e) A probabilistic conflict detection algorithm is implemented based on the proposed DGP trajectory prediction approach. The Gaussian distribution provides required probabilistic elements of trajectory positions and further supports the conflict detection, in which the Monte Carlo sampling is applied to simplify the integral solution.

2. Materials and Methods

2.1. Preliminary

2.1.1. Gaussian Process

GP is a statistical distribution, whose observations are defined in a continuous temporal or spatial domain. Each input observation is associated with a random variable that subjects to a multivariant Gaussian distribution [35,36]. Thus, the GP distribution is regarded as a joint distribution of all input random variables. A GP is mathematically denoted by Equation (1), in which $\mu(\vartheta)$ and $k(\vartheta, \vartheta*)$ are the mean and covariance function, respectively. In addition, the covariance function can be further represented by Equation (2) [36]:

$$f(\vartheta) \sim GP(\mu(\vartheta), k(\vartheta, \vartheta*)) \tag{1}$$

$$k(\vartheta, \vartheta*) = \mathbb{E}[(f(\vartheta) - \mu(\vartheta))(f(\vartheta*) - \mu(\vartheta*))] \tag{2}$$

Given the known training data $\{(\boldsymbol{\vartheta}_i, \boldsymbol{\xi}_i)\}_{i=1}^{n}$ and the observation is $\Phi = [\vartheta_1, \cdots, \vartheta_n]^{\mathrm{T}}$, the joint distribution of the observations is represented as Equation (3), where both $\boldsymbol{\mu} = [\mu(\vartheta_1), \cdots, \mu(\vartheta_n)]^{\mathrm{T}}$ and $\boldsymbol{K}$ are $n \times n$ matrix. The $(i, j)^{\mathrm{th}}$ element in $\boldsymbol{K}$ is obtained by Equation (4):

$$[f(\vartheta_1), \cdots, f(\vartheta_n)]^{\mathrm{T}} \sim \boldsymbol{\mathcal{N}}(\boldsymbol{\mu}, \boldsymbol{K}) \tag{3}$$

$$\boldsymbol{K}_{ij} = k(x_i, x_j) \tag{4}$$

It is also known as the kernel function in machine learning approaches. Several classical kernel functions for GP are summarized as follows [33,37]:

(a) Constant: $k_C(x, y) = C$;

(b) Linear: $k_L(x, y) = x^{\mathrm{T}} y$;

(c) Gaussian noise: $k_{GN}(x, y) = \sigma^2 \delta_{xy}$;

(d) SE: $k_{SE}(x, y) = a^2 \exp(-\frac{\|x-y\|^2}{2\lambda^2})$, i.e., squared exponential.

Based on the Gaussian property, for a new input observation $O = [o_1, \cdots, o_m]^{\mathrm{T}}$, the GP predicts its distribution $\boldsymbol{f}_* = f(O)$ by using a joint distribution with that of the known data:

$$\begin{bmatrix} \boldsymbol{y} \\ \boldsymbol{f}_* \end{bmatrix} \sim \boldsymbol{\mathcal{N}}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu}_* \end{bmatrix}, \begin{bmatrix} K(\Phi, \Phi) + \delta_n^2 I & K(O, \Phi)^{\mathrm{T}} \\ K(O, \Phi) & K(O, O) \end{bmatrix}\right) \tag{5}$$

It can be seen that the joint distribution of the prediction is also a Gaussian distribution, whose mean is denoted by $\boldsymbol{\mu}_* = [\mu(o_1), \cdots, \mu(o_m)]^{\mathrm{T}}$. The covariance matrix of the predicted distribution is determined by that of both the known data and unknown ones. The $K(\cdot, \cdot)$ is the covariance matrix regarding specific random variables. A GP is able to illustrate a high-level data distribution by considering the temporal transition of the defined random variables between prediction instants.

2.1.2. Trajectory Database

In this work, a flight trajectory is a typically temporal sequence, as shown in Equation (7). The x, y, z are the flight location in the three-dimensional space, while the $v_{\cdot}^{\cdot}$ is its velocity in corresponding dimensions. The combination of the flight location and velocity is called as the motion state which is the modeling object of the proposed DGP model. Thereby, a flight trajectory database is organized with a top-down architecture, as shown in Equation (6), where F is the number of flights in the database. To unify the unit of flight positions, x, y and z are converted into a same projected coordinate, which are measured by meter [38]:

$$TD = \{Tr_1, \cdots, Tr_F\} \tag{6}$$

$$Tr_j = \left\{t_i^j \rightarrow (x_i^j, y_i^j, z_i^j, v_x^{ij}, v_y^{ij})\right\}_{i=1}^{T_j}, j = 1, \cdots, F \tag{7}$$

In general, the flight trajectories for different historical operations usually share high similarity by travelling along the same planning route and flied over the same waypoint sequence [2,39]. The historical trajectories are proved to be safe and feasible, which provides sufficient data foundation for the trajectory prediction in this work. More specifically, the flight trajectory of each operation for the same flight has similar transition patterns, such as velocity or altitude change. Even facing irregular conditions, such as a flight returning or landing at an alternating airport, the flight trajectory also has routine schemes, i.e., high trajectory similarity. In addition, the airspace configuration, including the waypoint architecture and the environmental requirements, promotes the machine learning-based trajectory prediction approach, in which the trajectory diversity also allows the model to learn a probabilistic distribution to achieve the conflict detection task.

2.1.3. Deep Gaussian Process

Deep Gaussian process (DGP) is a combination of the deep neural network (DNN) and the GP. It is a type of DNN organized with the hierarchical architecture, in which GPs serve as the nonlinear activation between two neurons. A DGP with only the input layer and one neuron is degenerated into a standard GP. From the perspective of the DNN model, there are three types of layer in a DGP model, as shown below, in which $\Re^*$ is the dimension of data space:

(a) The output layer: $\boldsymbol{Y} \in \Re^{N \times D}$;
(b) The input layer: $\boldsymbol{Z} = \boldsymbol{X}_1 \in \Re^{N \times Q_Z}$;
(c) The hidden layer: $\boldsymbol{X}_h \in \Re^{N \times Q_h}, h \in [1, H]$. H is the number of layers in the DGP.

Considering a DGP with only one hidden layer, the graphic model is depicted in Figure 2 [33]. The f^x and f^y are the nonlinear activation of a Gaussian process between layers, and the inference rules of the model are formulated as Equations (8) and (9):

$$y_{nd} = f_d^Y(x_n) + \varepsilon_{nd}^Y, d \in [1, D], x_n \in \Re^Q \tag{8}$$

$$x_{nq} = f_q^X(\boldsymbol{z}_n) + \varepsilon_{nq}^X, q \in [1, Q], \boldsymbol{z}_n \in \Re^{Q_Z} \tag{9}$$

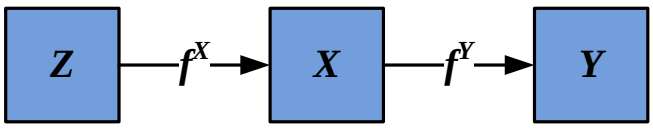

Figure 2. Graphic model of a DGP with one hidden layer.

In the Bayesian framework, the training of a DGP model can be implemented by optimizing the target function (Equation (10)). Due to the intractability of nonlinear propagation (GP) between neurons, Jensen's inequality is applied to replace it with a variational lower bound $\mathcal{F}_v \leq \log p(\boldsymbol{Y})$,

as shown in Equations (11) and (12) [33]. $\boldsymbol{F}^Y$ and $\boldsymbol{F}^X$ are the GP distribution of the nonlinear between different layers and neurons. However, the conditional distribution of $p(\boldsymbol{F}^Y|\boldsymbol{X})$ and $p(\boldsymbol{F}^X|\boldsymbol{Z})$ still limits the target optimization. An alternative solution of this dilemma is to build extra latent space [40], i.e., inducing points. The detailed inference of the DGP training under the Bayesian framework is illustrated in [35]:

$$\log p(\boldsymbol{Y}) = log \int_{\boldsymbol{X},\boldsymbol{Z}} p(\boldsymbol{Y}|\boldsymbol{X})p(\boldsymbol{X}|\boldsymbol{Z})p(\boldsymbol{Z}) \tag{10}$$

$$\mathcal{F}_v = \int_{\boldsymbol{X},\boldsymbol{Z},\boldsymbol{F}^Y,\boldsymbol{F}^X} \varsigma \log \frac{p(\boldsymbol{Y},\boldsymbol{F}^Y,\boldsymbol{F}^X,\boldsymbol{X},\boldsymbol{Z})}{\varsigma} \tag{11}$$

$$p(\boldsymbol{Y},\boldsymbol{F}^Y,\boldsymbol{F}^X,\boldsymbol{X},\boldsymbol{Z}) = p(\boldsymbol{Y}|\boldsymbol{F}^Y)p(\boldsymbol{F}^Y|\boldsymbol{X})\ p(\boldsymbol{X}|\boldsymbol{F}^X)p(\boldsymbol{F}^X|\boldsymbol{Z})p(\boldsymbol{Z}) \tag{12}$$

Obviously, just like the DNN model, a DGP can be extended in the vertical or horizontal dimension, as shown in Figure 3. With a vertical extension, the DGP is built with a deeper architecture to mine the higher-level representations of the input features. When it comes to the horizontal extension, more latent spaces are considered as the conditional independencies in the same hidden layer. In Figure 3, the arrows between layers indicate the information transmission with GPs in the DGP model, and each arrow corresponds to a GP prior with a unique parameter set. The DGP training is to optimize the parameters for each GP prior based on the patterns of the training samples.

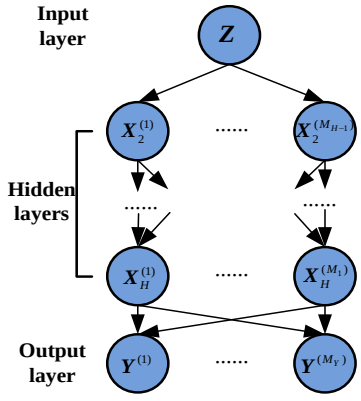

Figure 3. Graphic model of an extended DGP.

2.2. Trajectory Prediction

In the proposed trajectory approach, GP is applied to build the motion state transitions of the flight trajectory, which is also a machine learning-based flight trajectory prediction algorithm. The DNN architecture is applied to enhance the representation of the nonlinear features, and further improve the prediction performance [23]. The following rules indicate how the GP is applied to predict the flight trajectory in this work:

(a) Data pairs with respect to the flight time t and motion states in historical trajectories (Tr) are regarded as the known training data, i.e., $(\boldsymbol{x},\boldsymbol{y})$, to optimize the DGP parameters;

(b) Basically, the time instants in the prediction period serve as the input observation, i.e., Z. Consequently, the mean and standard variation of the predicted Gaussian distribution at different instants illustrate the distributed possibility of trajectory positions in the 3D space;

(c) The predicted deterministic trajectory connecting the means of all prediction instants is called the nominal flight trajectory (NFT);

(d) The covariance matrix indicates the probabilistic distribution of the flight position at a different instant by a sequence of Gaussian distribution with different parameters.

In this section, three models are designed to address the common trajectory prediction issues in air traffic studies. The paradigm for different flight operations is shown in Figure 4.

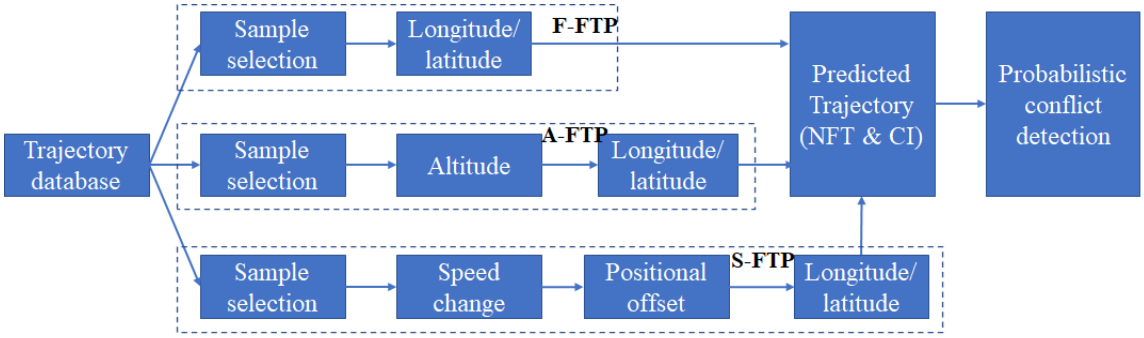

Figure 4. The paradigm of the proposed approach for the different flight operations.

2.2.1. Free Flight Trajectory Prediction (F-FTP)

The flight is performed freely by maintaining current motion states (velocity). Waypoint sequence is parsed from the flight plan to determine the change of the flight heading. As described before, the F-FTP is implemented by building the transition of the temporal sequence of flight trajectory, i.e., $t \rightarrow (x, y, z)$. Historical trajectories of a given flight serve as the training data, in which the flight time t is converted into the relative time (based on the takeoff time) to align the input time sequence.

2.2.2. Altitude-Driven Flight Trajectory Prediction (A-FTP)

A-FTP is mainly proposed to predict the trajectory when the aircraft performs an altitude instruction, i.e., climb or descent. As shown in Figure 5, the flight altitude goes through the elevation of the departure airport to cruise the altitude and further to the elevation of the arrival airport, which allows the model to learn the altitude transitions for the whole flight process.

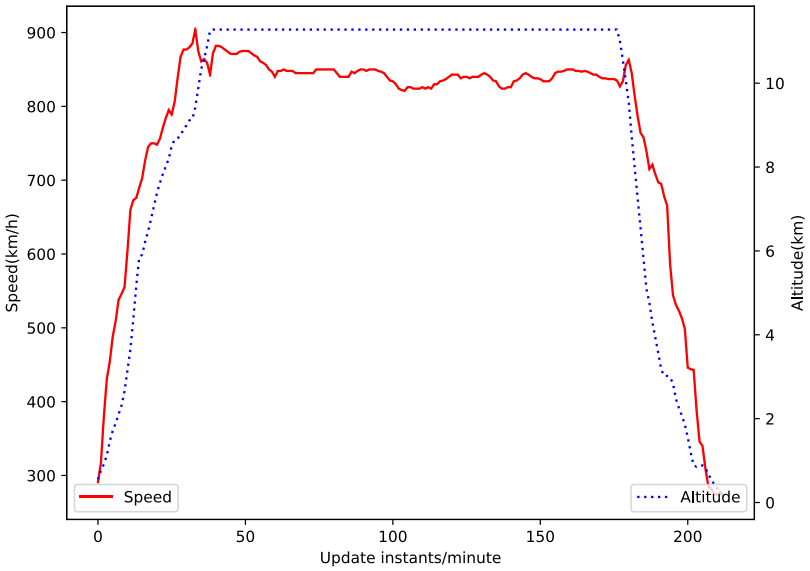

Figure 5. The speed and altitude of a certain flight during its operation.

It is clear that the altitude change drives the flight trajectory with considerable high maneuverability during the climb or descent flight phase. Therefore, the flight trajectory in altitude dimension is required to be predicted before obtaining the trajectory in the horizontal plane, which formulates a two-stage flight trajectory prediction approach, as shown in Equation (13). For instance, to predict the

flight trajectory of climbing from 8400 to 8900 m, the flight time of adjusting altitude is firstly estimated by a DGP model as $[z_a, \cdots, z_b] \rightarrow T_c + [0, \cdots, t_{za \rightarrow zb}]$, where the T_c is the current flight time, za and zb are 8400 and 8900, respectively. Afterwards, the trajectory in the horizontal plane is predicted by another DGP model as $(x_t, y_t)_{t=1}^{t_{za \rightarrow zb}}$, just like F-FTP. Finally, the predicting trajectory of the A-FTP in three-dimensional space is $Tr = \{(t \rightarrow (Tr_t^h, Tr_t^v)\}_{t=1}^{t_{za \rightarrow zb}}$:

$$\underbrace{\{z_i \rightarrow t_i\}_{i=za}^{zb}}_{Tr_v} \Rightarrow \underbrace{\{t_i \rightarrow (x_i, y_i)\}_{i=za}^{zb}}_{Tr_h} \tag{13}$$

2.2.3. Speed-Driven Flight Trajectory Prediction (S-FTP)

S-FTP is proposed to predict the flight trajectory for performing speed instructions in the horizontal plane, i.e., acceleration or deceleration. As shown in Figure 5, in the flight operational progress, the aircraft accelerates from minimum takeoff speed to its cruise speed and further decelerates to the maximum landing speed. Unlike the A-FTP, a DGP model cannot be optimized from flight positions directly since the speed instructions can be performed at any time and location. To this end, a three-stage S-FTP approach is proposed in this work, which is summarized in detail below:

(a) The flight time corresponding to the aircraft speed is firstly predicted by a DGP model, i.e., $\{(v_i, fl_i) \rightarrow t_i\}_{i=va}^{vb}$, in which va and vb are the starting and target speed of the aircraft, respectively. Since the aircraft performance (acceleration) is limited by aerodynamics, fl_i is also considered to build discriminative characteristics for speed regulating at different flight levels.

(b) Based on the predicted speed and aircraft heading, the positional offset of the flight trajectory is predicted regarding the flight time by another DGP, as shown below:

$$\Delta Tr = \left\{(t_i, v_{xi}, v_{yi}) \rightarrow (\Delta x_i, \Delta y_i, \Delta z_i)\right\}_{i=1}^{t_{va \rightarrow vb}} \tag{14}$$

(c) Based on the current flight position $Tr_0 = [x_0, y_0, z_0]^{\mathrm{T}}$, the final predicted trajectory can be accumulated as follows, which is the final predicted trajectory for speed regulating flights:

$$\begin{aligned} Tr &= [Tr_0, \cdots, Tr_i]^{\mathrm{T}}, \forall i \in [1, t_{va \rightarrow vb}] \\ Tr_i &= [x_{i-1} + \Delta x_i, y_{i-1} + \Delta y_i, z_{i-1} + \Delta z_i]^{\mathrm{T}} \end{aligned} \tag{15}$$

2.3. Conflict Detection

Based on the predicted nominal trajectory and its positional distribution, a probabilistic STCD approach is implemented from a pairwise view (between any two aircraft) [41]. As mentioned before, the predicted positions of the two aircraft subject to the Gaussian process with unique parameters, denoted by GP_i and GP_j, the conflict probability of the two aircraft is evaluated as Equation (16):

$$p_{ij} = \int_{r_{ij} \leq R} f(r_{ij}) dr_{ij}, r_{ij} = \|GP_i - GP_j\| \tag{16}$$

In general, the safety intervals of two aircraft are varied in three-dimensional earth space. Therefore, the target function is rewritten as Equation (17), in which the distance of two aircraft in the 3D space are computed separately. However, the integration is too complicated to obtain an analytic solution for the joint distribution of positional components. In this work, a Monte Carlo method is applied to obtain the solution by sampling positions from the predicted distributions of the two aircraft. Markov Chain Monte Carlo (MCMC) [42] is applied to sample positions and further to estimate the conflict probability. Drawn N position pairs (large enough) from their predicted GP distributions randomly, if there are n position pairs that their distance is less than the safety interval, the conflict probability of the two

aircraft is evaluated as n/N. Two flights are determined as a conflict pair only if the distance is less than the safety interval on all three dimensions. Finally, a probability threshold is pre-set to compute the conflict occurrence time and further obtain the pre-warning time:

$$\begin{aligned} p_{ij} = \int_{\substack{\Delta x \le Rx \\ \Delta y \le Ry \\ \Delta z \le Rz}} & f(\Delta x, \Delta y, \Delta z) d_{\Delta x} d_{\Delta y} d_{\Delta z} \\ \Delta x, \Delta y, \Delta z = & \left|x_i - x_j\right|, \left|y_i - y_j\right|, \left|z_i - z_j\right| \end{aligned} \tag{17}$$

3. Results and Discussions

3.1. Experimental Configurations

In this section, several experiments are designed to validate the proposed approaches, which are categorized into two groups, i.e., trajectory prediction and conflict detection:

(a) Trajectory prediction: four scenarios, namely cruise, right turn, descent, and deceleration, are designed to validate the proposed trajectory prediction approach. Three baselines are applied to compare the prediction accuracy, including KDA [9], Kalman [11] and GP [22]. The GP approach serves as a type of machine learning approach, which is also applied to validate the deeper architecture in this work. The prediction accuracy is evaluated by the dynamic time warping (DTW) [38,43] between the predicted trajectory and real trajectory. The final measurement is MEDTW, which is the mean error of DTW for all test flight trajectories.

(b) Conflict detection: four-hour historical data (surveillance data and flight plan) is replayed to prove the effectiveness of the conflict detection algorithm, in which three real potential conflicts occurred in this period. A total of 50 potential conflicts are also simulated to increase the number of test cases. The false alarm (FA) is applied to evaluate the algorithm performance. In this work, the FA means that how many false alarms are detected when all the real and simulated conflicts are detected. In addition, the mean prewarning time (MPWT) is also used to evaluate the timeliness of the proposed approach, which is denoted by the difference between the occurrence time and predicting time. We also conduct a baseline to validate the proposed conflict detection algorithm, which is based on a machine learning-based trajectory prediction approach.

The replayed scenes (with conflicts) occurred on 17 November 2018, in an area control center, China. The training data were collected from 17 October 2018, to 16 November 2018, i.e., about one-month. The flights performed from 11:00 a.m. to 15:00 p.m. on 17 November 2018, are regarded as the test data to validate the proposed trajectory prediction approach, about 227 flights in this period. The predicted trajectories were further applied to detect potential flight conflicts.

In this work, only one hidden layer is designed for each DGP model, and the number of inducing points is set to 40. The kernel function is SE. The confidence interval of the prediction results is 95%. The safety interval of two aircraft is 5 nautical miles and 150 m in the horizontal and vertical plane, respectively.

3.2. Results and Discussions of Trajectory Prediction for Cruise Phase

In this section, all the mentioned four approaches are applied to predict the flight trajectory. The following figures report the experiment results of predicting the trajectory for a cruising flight at 8400 m, including prediction accuracy, predicted NFT, and its CIs (Figure 6).

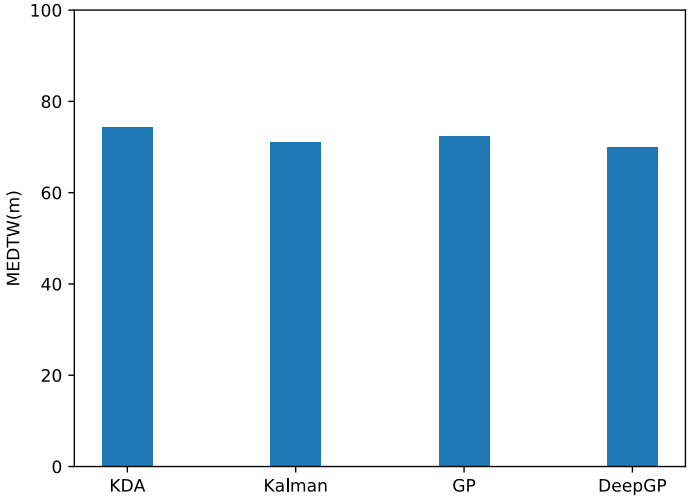

Figure 6. Experimental results for a cruise flight.

It can be seen from the experiment results that the four approaches obtain a comparable performance for the cruise flight due to the high linearity of the cruising trajectory. In addition, the proposed approach is slightly better than that of other baseline approaches. The following figure (Figure 7) shows the predicted trajectory of the proposed approach in the horizontal plane, including some of the training samples and the predicted NFT, as shown as the legend. The dash ellipse illustrates the probabilistic distribution of the flight position at a certain instant. The semi-axis of the ellipse is about 468 and 440 m in the X and Y dimensions, respectively, in which the predicted flight positions are subject to a Gaussian distribution. In addition, the 95% CIs of the X and Y dimensions are also displayed in Figure 8 (up) and Figure 8 (down). As can be seen from the figure, almost all flight positions locate in the confidence intervals for the cruise flight, which validates the proposed trajectory prediction approach.

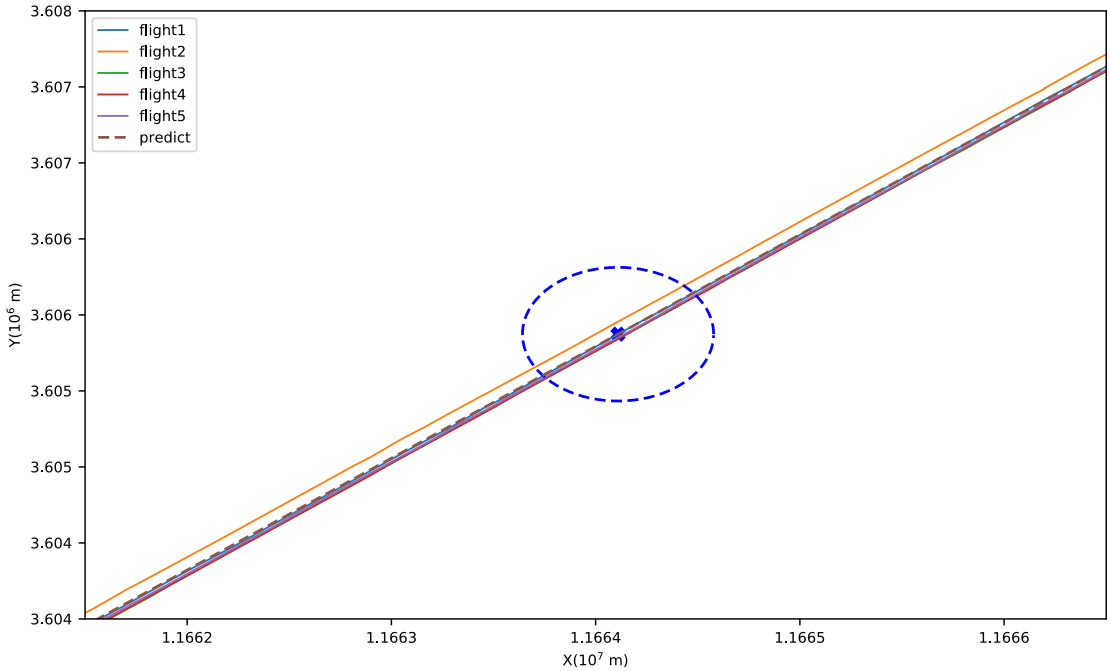

Figure 7. Predicted trajectory and its probabilistic distribution for a cruise flight.

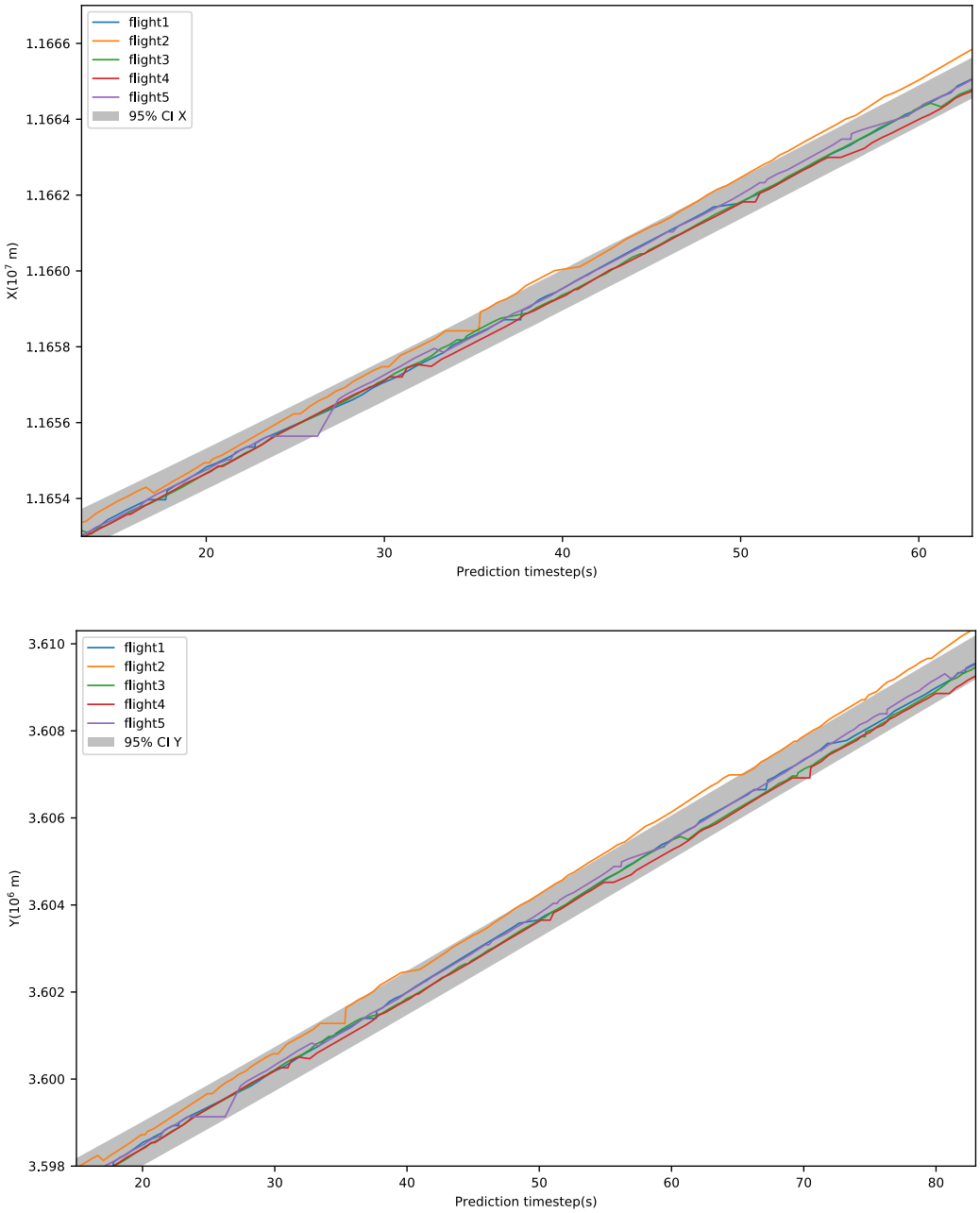

Figure 8. CIs (95%) of the predicted trajectory for a cruise flight (up: X dimension; down: Y dimension).

3.3. Results and Discussions of Trajectory Prediction for Turn Phase

The predicted MEDTWs of the different approaches for the flight turn are reported in Figure 9. The KDA suffers from the largest prediction error because it fails to fully consider the influence of maneuverability of the real-time flight environment when the aircraft is regulating its heading. As the same reason, the Kalman approach also has a large prediction error. Thanks to the learning mechanism of the GP, the GP and DGP approaches obtain higher prediction accuracy compared to other baselines. In addition, the DGP approach obtains more accurate prediction because of the promotion of the hierarchical architecture.

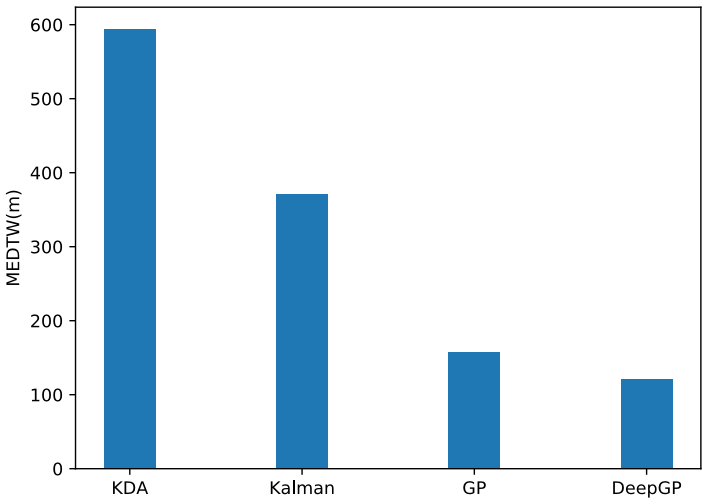

Figure 9. Experimental results for flight turn.

The predicted trajectory and its CIs are reported in the Figures 10 and 11, respectively. Due to the aircraft maneuverability during flight turn, the flight trajectories show huge divergence. Some flight positions are even distributed outside of the predicted CIs. Fortunately, the GP has the ability to capture frequent transition patterns from real operating data, which is further applied to improve the accuracy of the prediction results. Furthermore, the predicted CIs of GP-based approaches allow us to analyze the trajectory patterns in a probabilistic manner, not only for a deterministic one.

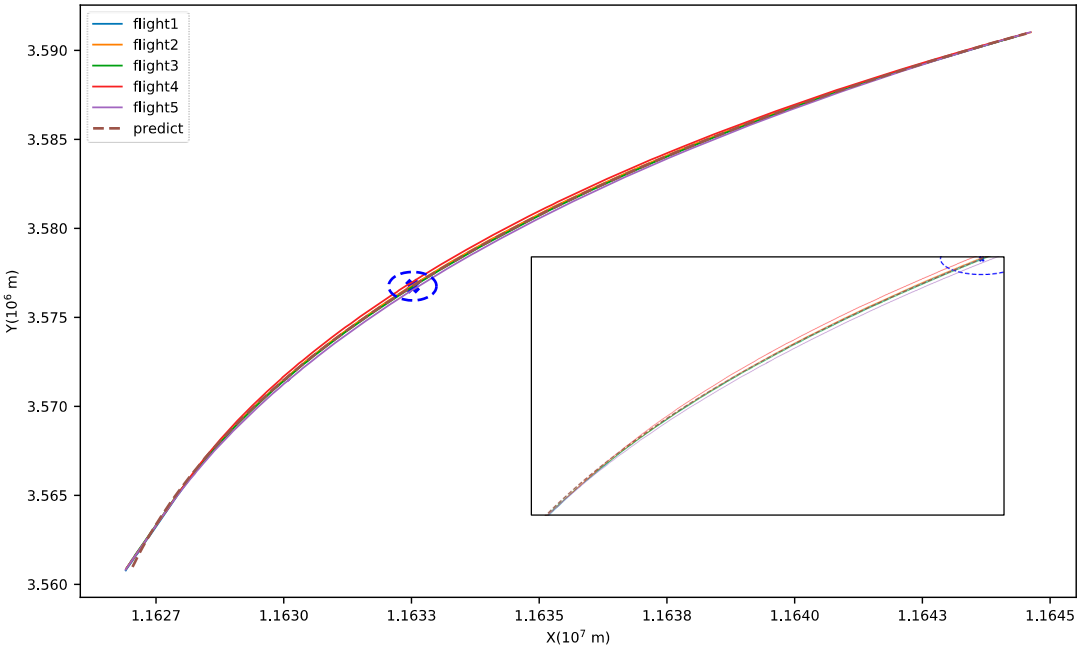

Figure 10. Predicted trajectory for the flight turn (the subfigure is the partial enlargement of the predicted trajectory).

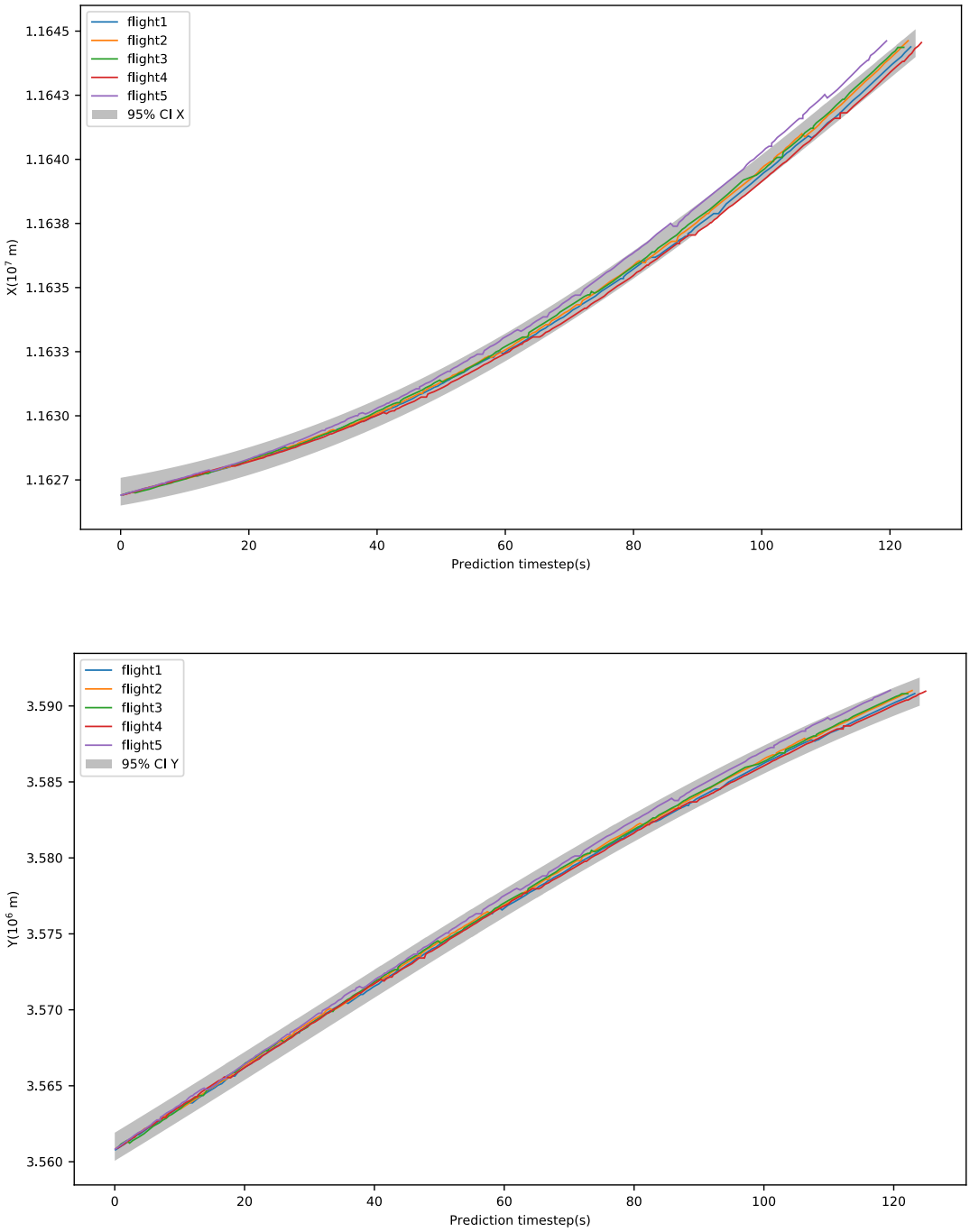

Figure 11. CIs (95%) of the predicted trajectory for flight turn (up: X dimension; down: Y dimension).

3.4. Results and Discussions of Trajectory Prediction for Descent Phase

In this section, several experiments are conducted to predict the flight trajectory during its descent from 8100 to 7500 m. In this section, the Kalman-based approach is not optimized for the prediction due to its state transition rules. The prediction errors of the flight time and trajectory are reported in Figure 12. The A-FTP is achieved by two steps: predicting the altitude and the flight time, and its horizontal trajectory. As can be seen from the results, GP-based approaches obtain higher accuracy (both altitude, flight time and horizontal trajectory) compared to the KDA approach since the learning mechanism considers the real operational patterns from historical trajectories. In practice, since the flight climb or descent cannot affect passenger' comfort, the climb or descent rate of the aircraft

is moderate and basically remains a constant one. The fact of the climb or descent rate enhances the predictability of the flight trajectory, which benefits the final accuracy. As can be proven by the experimental results (Figure 13), the real flight time regarding to the flight altitude appears to be a striking regularity and almost all of them locate in the predicted CI. Based on the predicted flight time and altitude, the horizontal trajectory can be obtained by applying the proposed F-FTP approach.

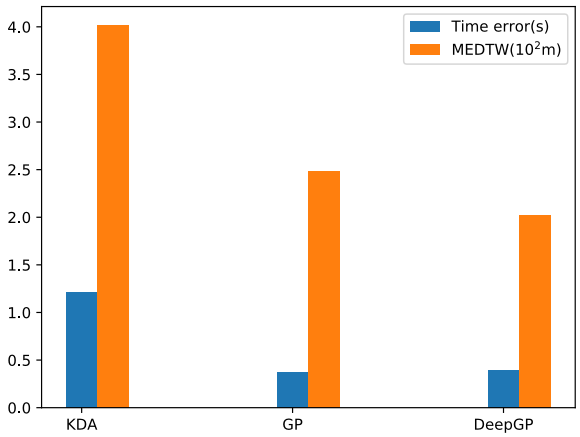

Figure 12. Experimental results for a descent flight.

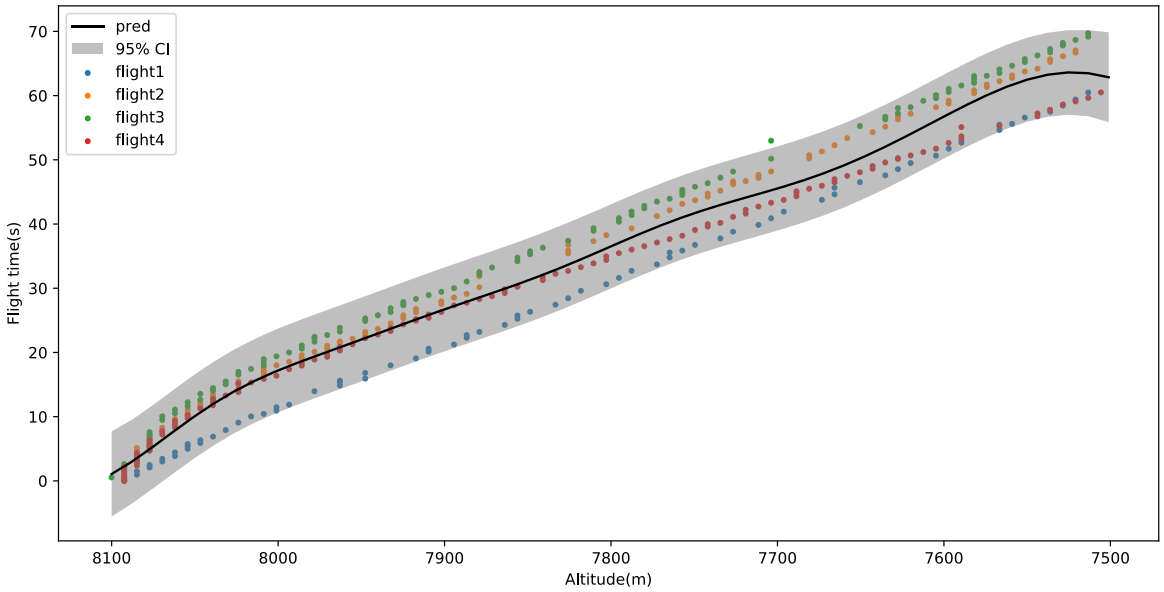

Figure 13. The predicted flight time of its CI for a descent flight from 8100 to 7500 m.

3.5. Results and Discussions of Trajectory Prediction for Deceleration Phase

Based on the experimental design, the proposed approach is applied to predict the trajectory for a flight decelerating from 216 to 186 km/h. The prediction errors of the flight time and trajectory are reported in Figure 14, from which we can see that GP-based approaches obtain higher prediction accuracy than that of the baseline. Since regulating aircraft speed has more variable parameters, the prediction error of flight time is obviously larger than that of regulating flight altitude, as shown in Figure 15. After predicting the flight time with respect to the flight speed, the positional offsets of the flight trajectories are estimated based on a DGP model. Finally, the flight trajectory is obtained by accumulating the positions of last instant and its positional offset. In Figure 16, the positional offsets at a different instant are shown to illustrate the transition patterns of flight trajectory for

performing the flight deceleration. In general, the positional offsets are distributed in a divergent interval, which indicates that the aircraft maneuverability degenerates the prediction accuracy of the flight trajectory. Fortunately, the intrinsic advantages of the GP promote the prediction performance of the proposed approach.

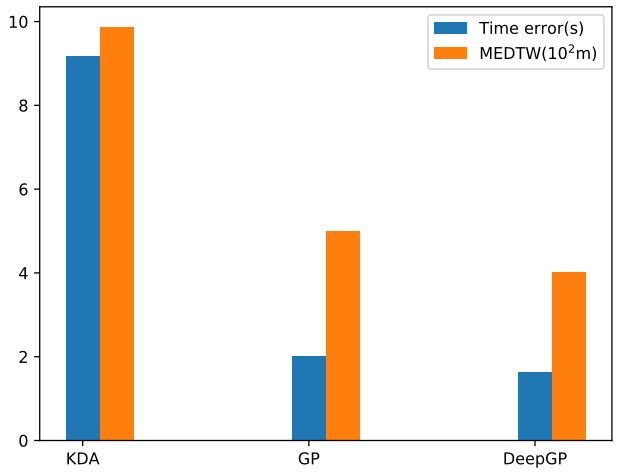

Figure 14. Experimental results for a decelerating flight.

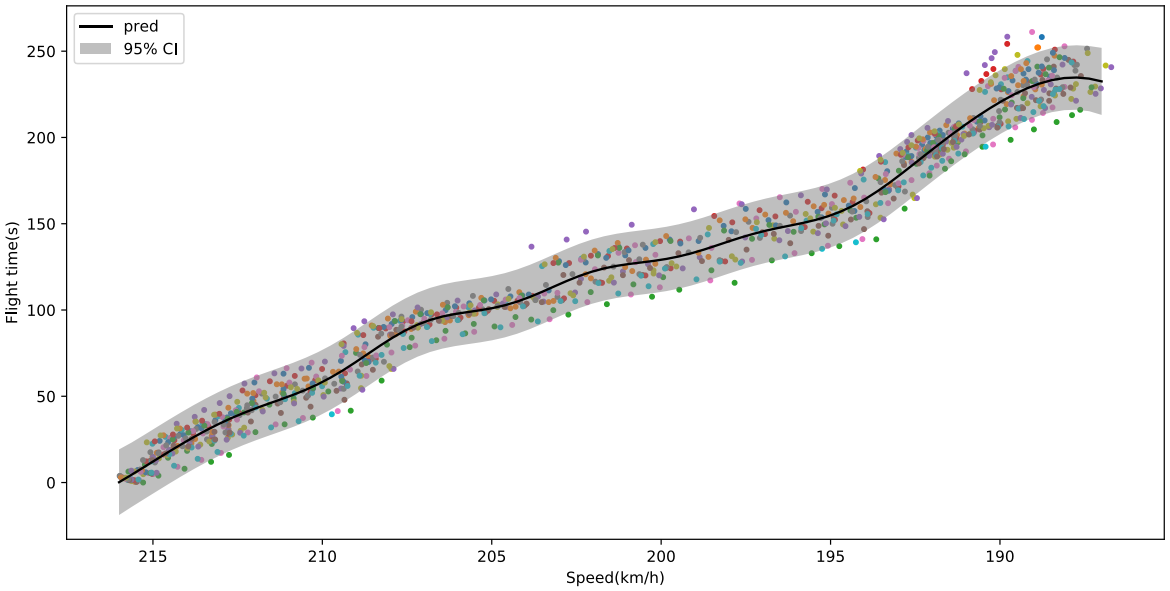

Figure 15. The predicting flight time and its CI with respect to the flight speed (the dots in a different color are extracted from historical trajectories).

3.6. Results and Discussions for Conflict Detection

After validating the proposed trajectory prediction approach, the conflict detection algorithm is also tested based on the replaying data and simulated conflicts. The flight trajectory in the next 3 min is predicted to illustrate the positional distribution and further support the flight conflict detection. To improve the prediction accuracy, the trajectory of test flights is predicted at each updating interval, of about 4 s. The experimental results are summarized in Table 1, in which a baseline approach is also implemented based on a probabilistic trajectory prediction algorithm [4]. The experimental purpose is to minimize the FA in the case of detecting all real and simulated conflicts. The values in the "Results" column indicate that 15 and 26 false alarms are detected when performing the approach to detect all

53 conflicts. It can be seen from the results that the proposed conflict detection approach can obtain a lower FA (22.06 vs. 32.91) compared to the baseline approach. Moreover, the proposed approach provides more prewarning time (+5.7 s) to concerned controllers for dealing with the emergency. Thus, the experimental results validate the effectiveness and efficiency of the proposed approach for the conflict detection task in air traffic studies.

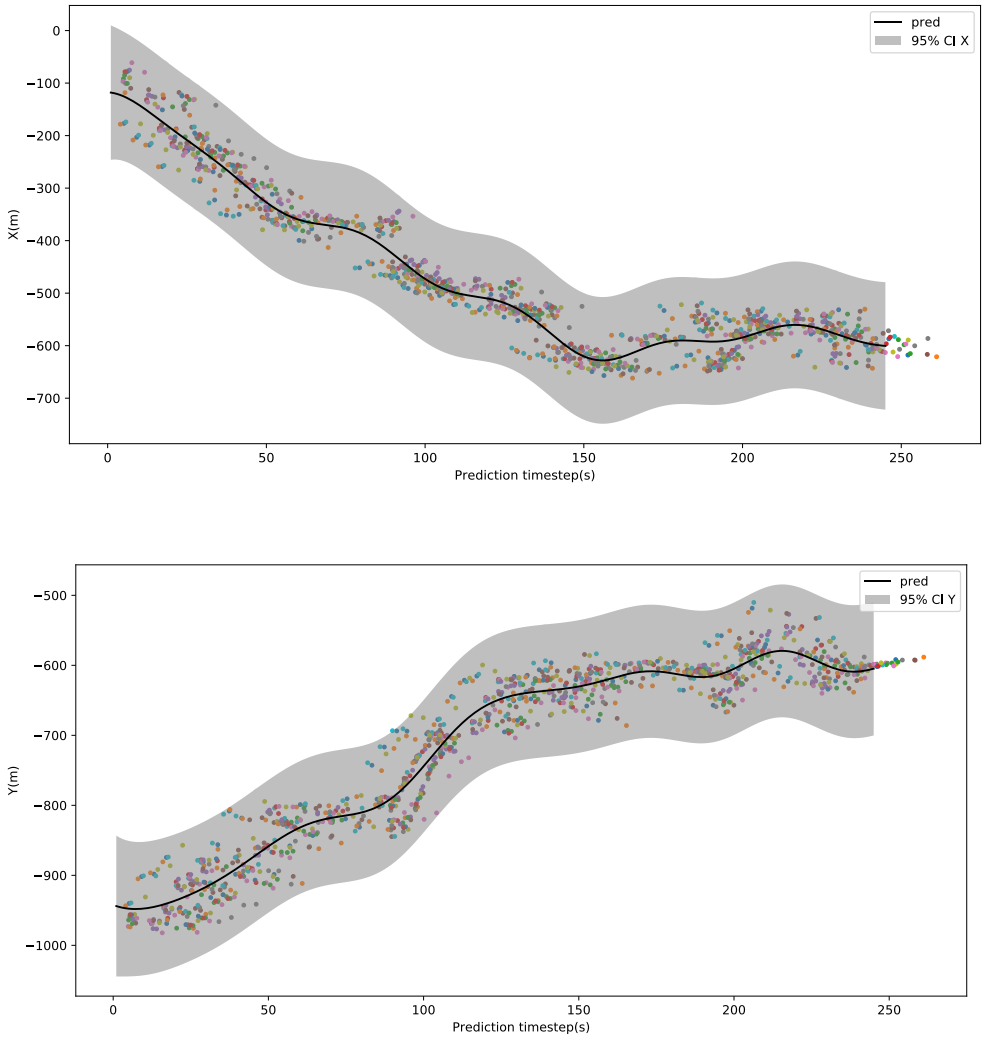

Figure 16. The positional offsets of the predicted trajectory for a decelerating flight (up: X dimension; down: Y dimension; the dots in a different color are extracted from historical trajectories).

Table 1. Experimental results for conflict detection.

Approaches	Results	FA (%)	MPWT (s)
The proposed	53/15	22.06	39.4
The baseline	53/26	32.91	33.7

4. Conclusions

In this work, we proposed a deep Gaussian process-based framework to address the trajectory prediction issue in air traffic studies. In succession, the predicted trajectory was applied to implement a probabilistic conflict detection algorithm. The proposed approach is able to deal with the trajectory prediction for different flight operations, including the cruise phase and maneuverable phase (such as

regulating flight heading, altitude and speed). Gaussian process is proposed to build the temporal dependencies for the sequential data (flight trajectory), in which the predicted mean and standard deviation serve as the nominal flight trajectory and its confidence interval, respectively. The hierarchical architecture in neural network improves the representation ability of nonlinear features, which is very important to cope with the stochastic flight environment. Thanks to the Gaussian property, a probabilistic conflict detection algorithm is naturally achieved based on the proposed trajectory prediction approach. Experimental results on real operating data show that the proposed DGP model obtained higher accuracy compared to other baselines. Moreover, the conflict detection task is also fulfilled with a low false alarm rate based on real data and simulated conflicts.

In the future, we first plan to build more accurate constraints on selecting training data to improve the applicability of the proposed approach. Due to the importance of the training samples, we will also attempt to study a more efficient and effective method to match more proper trajectories to optimize the proposed model. The end-to-end paradigm for regulating flight altitude and speed is also an urgent task in this work. At last, the deep learning-based approach is also deserved to be studied to achieve the conflict detection task in a more straightforward manner.

Author Contributions: Conceptualization, Z.C. and Y.L.; methodology, Z.C. and D.G.; software, D.G.; validation, Z.C., D.G. and Y.L.; investigation, Y.L.; data curation, Y.L.; writing—original draft preparation, Z.C.; writing—review and editing, Y.L.; visualization, D.G.; supervision, Y.L.; project administration, Y.L.; funding acquisition, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was funded by the National Natural Science Foundation of China (grant number 62001315) and Sichuan Science and Technology Program (grant number 2020YFG0327).

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

Abbreviations

A-FTP	altitude-driven flight trajectory prediction
CD	conflict detection
CI	confidence interval
DGP	deep Gaussian process
DNN	deep neural network
FA	false alarm
F-FTP	free flight trajectory prediction
GP	Gaussian process
MEDTW	mean error of dynamic time warping
MPWT	mean prewarning time
NFT	nominal flight trajectory
S-FTP	speed-driven flight trajectory prediction
STCD	short-term conflict detection
TP	trajectory prediction

References

1. Lin, Y.; Guo, D.; Zhang, J.; Chen, Z.; Yang, B. A Unified Framework for Multilingual Speech Recognition in Air Traffic Control Systems. *IEEE Trans. Neural Networks Learn. Syst.* **2020**, 1–13. [CrossRef]
2. Lin, Y.; Zhang, J.; Liu, H. Deep learning based short-term air traffic flow prediction considering temporal–spatial correlation. *Aerosp. Sci. Technol.* **2019**, *93*, 105113. [CrossRef]
3. Lin, Y.; Deng, L.; Chen, Z.; Wu, X.; Zhang, J.; Yang, B. A Real-Time ATC Safety Monitoring Framework Using a Deep Learning Approach. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 4572–4581. [CrossRef]
4. Lin, Y.; Zhang, J.; Liu, H. An algorithm for trajectory prediction of flight plan based on relative motion between positions. *Front. Inf. Technol. Electron. Eng.* **2018**, *19*, 905–916. [CrossRef]
5. De Leege, A.; van Paassen, M.; Mulder, M. A Machine Learning Approach to Trajectory Prediction. In Proceedings of the AIAA Guidance, Navigation, and Control (GNC) Conference, Boston, MA, USA, 19–22 August 2013; American Institute of Aeronautics and Astronautics: Reston, VA, USA, 2013; p. 4782. [CrossRef]

6. Hrastovec, M.; Solina, F. Prediction of aircraft performances based on data collected by air traffic control centers. *Transp. Res. Part C Emerg. Technol.* **2016**, *73*, 167–182. [CrossRef]
7. Shi, Z.; Xu, M.; Pan, Q.; Yan, B.; Zhang, H. LSTM-based Flight Trajectory Prediction. In Proceedings of the 2018 International Joint Conference on Neural Networks, Rio de Janeiro, Brazil, 8–13 July 2018; IEEE: New York, NY, USA, 2018; pp. 1–8. [CrossRef]
8. Tang, X.; Zhou, L.; Shen, Z.; Tang, M. 4D Trajectory Prediction of Aircraft Taxiing Based on Fitting Velocity Profile. In Proceedings of the 2015 Cota International Conference Transportation Professionals, Beijing, China, 24–27 July 2015; American Society of Civil Engineers: Reston, VA, USA, 2015; pp. 1–12. [CrossRef]
9. Chen, Z. *Theory and Method of Airspace Management*, 1st ed.; Science Press: Beijing, China, 2012.
10. Hashemi, S.M.; Botez, R.M.; Grigorie, T.L. New Reliability Studies of Data-Driven Aircraft Trajectory Prediction. *Aerospace* **2020**, *7*, 145. [CrossRef]
11. Prevost, C.G.; Desbiens, A.; Gagnon, E. Extended Kalman Filter for State Estimation and Trajectory Prediction of a Moving Object Detected by an Unmanned Aerial Vehicle. In Proceedings of the 2007 American Control Conference, New York, NY, USA, 9–13 July 2007; IEEE: New York, NY, USA, 2007; pp. 1805–1810. [CrossRef]
12. Yan, H.; Huang, G.; Wang, H.; Shu, R. Application of unscented kalman filter for flying target tracking. In Proceedings of the 2013 International Conference on Information Science and Cloud Computing, Guangzhou, China, 7–8 December 2013; IEEE: New York, NY, USA, 2013; pp. 61–66. [CrossRef]
13. Chan, Y.T.; Hu, A.G.C.; Plant, J.B. A Kalman Filter Based Tracking Scheme with Input Estimation. *IEEE Trans. Aerosp. Electron. Syst.* **1979**, 237–244. [CrossRef]
14. Schultz, C.; Thipphavong, D.; Erzberger, H. Adaptive Trajectory Prediction Algorithm for Climbing Flights. In Proceedings of the AIAA Guidance, Navigation, and Control Conference, Minneapolis, MN, USA, 13–16 August 2012; American Institute of Aeronautics and Astronautics: Reston, VA, USA, 2012; p. 4931. [CrossRef]
15. Thipphavong, D.P.; Schultz, C.A.; Lee, A.G.; Chan, S.H. Adaptive Algorithm to Improve Trajectory Prediction Accuracy of Climbing Aircraft. *J. Guid. Control. Dyn.* **2013**, *36*, 15–24. [CrossRef]
16. Wang, Z.; Delahaye, D. *Short-Term 4D Trajectory Prediction Using Machine Learning Methods*; Seventh SESAR Innovation Days 2017; SESAR Project: Belgrade, Serbia, 2017.
17. Lin, Y.; Li, L.; Jing, H.; Ran, B.; Sun, D. Automated traffic incident detection with a smaller dataset based on generative adversarial networks. *Accid. Anal. Prev.* **2020**, *144*, 105628. [CrossRef]
18. Franco, A.; Rivas, D.; Valenzuela, A. Probabilistic aircraft trajectory prediction in cruise flight considering ensemble wind forecasts. *Aerosp. Sci. Technol.* **2018**, *82*, 350–362. [CrossRef]
19. Jeung, H.; Shen, H.T.; Zhou, X. Mining Trajectory Patterns Using Hidden Markov Models. In Proceedings of the International Conference on Data Warehouse Knowledge Discovery, Regensburg, Germany, 3–7 September 2007; Springer: Berlin/Heidelberg, Germany, 2007; pp. 470–480. [CrossRef]
20. Nascimento, J.C.; Figueiredo, M.A.T.; Marques, J.S. Trajectory classification using switched dynamical hidden markov models. *IEEE Trans. Image Process.* **2010**, *19*, 1338–1348. [CrossRef]
21. Wiest, J.; Hoffken, M.; Kresel, U.; Dietmayer, K. Probabilistic trajectory prediction with Gaussian mixture models. In Proceedings of the 2012 IEEE Intelligent Vehicles Symposium, Alcala de Henares, Spain, 3–7 June 2012; IEEE: New York, NY, USA, 2012; pp. 141–146. [CrossRef]
22. Yan, H.; Yang, B.; Yang, H.; Wang, R. Probabilistic Approach to Conformance Monitoring Using Gaussian Processes. *J. Guid. Control. Dyn.* **2017**, *40*, 1403–1414. [CrossRef]
23. Wu, Z.-J.; Tian, S.; Ma, L. A 4D Trajectory Prediction Model Based on the BP Neural Network. *J. Intell. Syst.* **2019**, *29*, 1545–1557. [CrossRef]
24. Zhang, X.; Mahadevan, S. Bayesian neural networks for flight trajectory prediction and safety assessment. *Decis. Support Syst.* **2020**, *131*, 113246. [CrossRef]
25. Altche, F.; de la Fortelle, A. An LSTM network for highway trajectory prediction. In Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), Yokohama, Japan, 16–19 October 2017; IEEE: New York, NY, USA, 2018; pp. 353–359. [CrossRef]
26. Nivison, S.A.; Khargonekar, P.P. Development of a robust deep recurrent neural network controller for flight applications. In Proceedings of the 2017 American Control Conference (ACC), Seattle, WA, USA, 24–26 May 2017; IEEE: New York, NY, USA, 2017; pp. 5336–5342. [CrossRef]

27. Zhao, Z.; Zeng, W.; Quan, Z.; Chen, M.; Yang, Z. Aircraft trajectory prediction using deep long short-term memory networks. In Proceedings of the CICTP 2019 Transportation China—19th COTA International Conference Transports Professtional, Nanjing, China, 6–8 July 2019; American Society of Civil Engineers: Nanjing, China, 2019; pp. 124–135. [CrossRef]
28. Pang, Y.; Xu, N.; Liu, Y. Aircraft Trajectory Prediction using LSTM Neural Network with Embedded Convolutional Layer. In Proceedings of the Annual Conference of the PHM Society, Scottsdale, AZ, USA, 23–26 September 2019; PHM Society: Scottsdale, AZ, USA, 2019.
29. Liu, Y.; Hansen, M. Predicting Aircraft Trajectories: A Deep Generative Convolutional Recurrent Neural Networks Approach. *arXiv* **2019**, arXiv:1812.11670.
30. Hoel, P.G.; Port, S.C.; Stone, C.J.; Holley, R. Introduction to Stochastic Processes. *IEEE Trans. Syst. Man. Cybern.* **1973**, 533. [CrossRef]
31. Domingues, R.; Michiardi, P.; Zouaoui, J.; Filippone, M. Deep Gaussian Process autoencoders for novelty detection. *Mach. Learn.* **2018**, *107*, 1363–1383. [CrossRef]
32. Paielli, R.A.; Erzberger, H. Conflict Probability for Free Flight. *J. Guid. Control. Dyn.* **1997**, *20*, 588–596. [CrossRef]
33. Damianou, A.C.; Lawrence, N.D. Deep Gaussian Processes. In Proceedings of the 16th International Conference Artificial Intelligence and Statistics, Scottsdale, AZ, USA, 29 April–1 May 2013; PMLR: Scottsdale, AZ, USA, 2013; pp. 207–215.
34. Thang, B.; Daniel, H.-L.; Yingzhen, L.; José, H.-L.; Richard, T. Deep Gaussian Processes for Regression using Approximate Expectation Propagation. In Proceedings of the International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; PMLR: New York, NY, USA, 2016; pp. 1472–1481.
35. Seeger, M. Gaussian Processes for Machine Learning. *Int. J. Neural Syst.* **2004**, *14*, 69–106. [CrossRef]
36. Rasmussen, C.E. Gaussian Processes in Machine Learning. In *Summer School on Machine Learning*; Springer: Cham, Switzerland, 2004; pp. 63–71. [CrossRef]
37. Lawrence, N. Learning for larger datasets with the Gaussian process latent variable model. *J. Mach. Learn. Res. Proc. Track.* **2007**, *2*, 243–250.
38. Zheng, Y. *Computing with Spatial Trajectories*; Springer: New York, NY, USA, 2011. [CrossRef]
39. Lin, Y.; Yang, B.; Zhang, J.; Liu, H. Approach for 4-D Trajectory Management Based on HMM and Trajectory Similarity. *J. Mar. Sci. Technol.* **2019**, *27*, 246–256. [CrossRef]
40. Michalis, T.; Neil, L. Bayesian Gaussian Process Latent Variable Model. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Sardinia, Italy, 13–15 May 2010; PMLR: Sardinia, Italy, 2010; pp. 844–851.
41. Prandini, M.; Hu, J.; Lygeros, J.; Sastry, S. A Probabilistic Approach to Aircraft Conflict Detection. *IEEE Trans. Intell. Transp. Syst.* **2000**, *1*, 199–220. [CrossRef]
42. Kass, R.E.; Gilks, W.R.; Richardson, S.; Spiegelhalter, D.J. Markov Chain Monte Carlo in Practice. *J. Am. Stat. Assoc.* **1997**, *92*, 1645. [CrossRef]
43. Zhang, Y.; Edgar, T.F. A robust Dynamic Time Warping algorithm for batch trajectory synchronization. In Proceedings of the 2008 American Control Conference, Seattle, WA, USA, 11–13 June 2008; IEEE: New York, NY, USA, 2008; pp. 2864–2869. [CrossRef]