In *Proceedings of the 22nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 21–30,
San Francisco, August 2016.

Aircraft Trajectory Prediction *Made Easy* with Predictive Analytics

Samet Ayhan [*] Hanan Samet
Department of Computer Science
University of Maryland
College Park, MD 20742
{sayhan} {hjs} @cs.umd.edu

ABSTRACT

At the heart of Air Traffic Management (ATM) lies the Decision Support Systems (DST) that rely upon accurate trajectory prediction to determine how the airspace will look like in the future to make better decisions and advisories. Dealing with airspace that is prone to congestion due to environmental factors still remains the challenge especially when a deterministic approach is used in the trajectory prediction process. In this paper, we describe a novel stochastic trajectory prediction approach for ATM that can be used for more efficient and realistic flight planning and to assist airspace flow management, potentially resulting in higher safety, capacity, and efficiency commensurate with fuel savings thereby reducing emissions for a better environment.

Our approach considers airspace as a 3D grid network, where each grid point is a location of a weather observation. We hypothetically build cubes around these grid points, so the entire airspace can be considered as a set of cubes. Each cube is defined by its centroid, the original grid point, and associated weather parameters that remain homogeneous within the cube during a period of time. Then, we align raw trajectories to a set of cube centroids which are basically fixed 3D positions independent of trajectory data. This creates a new form of trajectories which are 4D joint cubes, where each cube is a segment that is associated with not only spatio-temporal attributes but also with weather parameters. Next, we exploit machine learning techniques to train inference models from historical data and apply a stochastic model, a Hidden Markov Model (HMM), to predict trajectories taking environmental uncertainties into account. During the process, we apply time series clustering to generate input observations from an excessive set of weather parameters to feed into the Viterbi algorithm. Our experiments use a real trajectory dataset with pertaining weather observations and demonstrate the effectiveness of our approach to the trajectory prediction process for ATM.

Keywords

Aircraft Trajectory Prediction; Predictive Analytics; Air Traffic Management; Hidden Markov Model; Time Series

1. INTRODUCTION

The goals of Air Traffic Control (ATC) are the safe and efficient management of aircraft. The Air Navigation Service Provider (ANSP) in the USA, the FAA, is concerned with providing optimal airspace capacity and efficiency within the National Airspace System (NAS). These goals require the separation and sequencing of airborne aircraft by controllers who often use DSTs to monitor the progress of each aircraft and resolve conflicts when necessary. Other controllers monitor the future aggregate flows of aircraft traffic and using a DST implement flow management decisions to ensure that the traffic density always remains within safe limits. At the core of these automation tools resides the Trajectory Prediction (TP) tool, that given the aircraft initial state, its flight plan and the wind and temperatures, computes where the aircraft will be in the future in four dimensions. Inaccurate trajectory prediction can have a substantial impact on the performance of the DST and the ATM system, resulting in

- larger separation standards that limit the number of aircraft that can be allowed in controllers' sectors and/or
- increased numbers of conflicts increasing the workload of the controllers and reducing their sector capacity and/or
- inefficient fuel consumption due to continual deviations to avoid other aircraft and/or
- adverse impact on the environment due to more fuel burn leading to more emissions.

As a consequence, the performance of the TP tool which supports the DSTs is critical to the success of the DST functions.

Although prior trajectory prediction research and development activities have been able to address the challenge to some extent, dealing with increasingly congested airspace and new environmental concerns still remains the challenge when deterministic approach is used in trajectory prediction process. Hence due to the nature of uncertainties that contribute to trajectory prediction errors, in our research, we have taken a stochastic approach to address the trajectory prediction issues.

We define airspace as a set of spatio-temporal data cubes where each cube is considered as an atomic unit. Weather parameters such as temperature, wind speed, and wind di-

[*] Senior Engineer at Boeing Research & Technology

KDD '16, August 13-17, 2016, San Francisco, CA, USA
© 2016 ACM. ISBN 978-1-4503-4232-2/16/08…$15.00
DOI: http://dx.doi.org/10.1145/2939672.2939694

rection remain homogeneous within the cube during a period of time. Other parameters that describe the cube include its center coordinates (latitude, longitude, altitude) along with a time stamp. These spatio-temporal data cubes form the overall airspace. Then, we adapt raw trajectories to the cubes, by aligning each trajectory vertex to the nearest cube centroid, inspired by the *Realm* method [19]. The process yields 4D joint cubes that can be considered as piecewise trajectory segments. Using adapted trajectories in the form of 4D joint cubes, we train our model with the historical data and choose a state sequence that best explains the current observations, the pertinent weather parameters. This process corresponds to the Problem No.2 described in [33] where it attempts to uncover the hidden part of the model, which is the optimal state sequence, given the observations. Note that in the process, each segment is chosen from one of the many hidden segments and the occurrence of a hidden segment that underlines an observation is dependent on the hidden state of the previous segment. Since our objective is to find the most likely trajectory, we use HMM that is nothing more than a probabilistic function of a Markov process. Due to the nature of interconnected cube centroids forming a trellis, we use the Viterbi algorithm [48] to efficiently generate the optimal trajectory by joining the multiple segments together, where one segment is only dependent on the previous segment.

Now, a critical question arises: *What constitutes the input observations?* or more importantly, *among so many weather observations in the airspace volume of interest during the time period of flight, which ones should be passed to the Viterbi process?* We address this problem by clustering the current weather observations for the cube centroids that were historically traversed. Before the process, we split the current weather observations into buckets and then perform time series clustering [21]. We finally feed the output cluster into the Viterbi algorithm to probabilistically generate the best state sequence.

In summary, the contributions of this paper are as follows:

- We propose a novel way of representing aircraft trajectories, a set of 4D joint cubes generated upon an alignment and fusion process.
- We build a stochastic model, HMM that learns from the combination of historical trajectories and aircraft specifications, and their correlation with the pertinent weather parameters. Then, we perform time series clustering on the trajectory segments that were historically traversed. Finally, we feed the series of cluster centroids as observations into the Viterbi algorithm in order to predict trajectories that can be used as more realistic pre-departure flight plans.
- We conduct experiments based on real host track and aircraft specification data and demonstrate that our system effectively predicts aircraft trajectories.

Although the proposed solution can be adapted for both tactical and strategic trajectory prediction, our experimental study focuses more on a ground-based tactical trajectory prediction system that can be used by both AOC dispatchers to file more realistic flight plans and/or flow managers at the Air Traffic Control System Command Center (ATCSCC) to approve the proposed flight plans based on their conformance to the relatively more realistic predicted trajectories, right before the aircraft departs.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 introduces the preliminary concepts, presents the problem, and overviews the proposed system. Section 4 discusses in the detail the aircraft trajectory prediction system, that we propose. Section 5 presents the results of an experimental evaluation. Section 6 concludes the paper and outlines some future work.

2. RELATED WORK

There has been much work on trajectories in the spatial domain for motor vehicles along roads [40] with an emphasis on their generation (e.g., [41]), queries (e.g., [30, 32, 36, 38, 39]), and matching (e.g., [20, 29, 37]). This is not our subject here. On the other hand, there has been a vast amount of research and abundant literature with regards to predicting aircraft trajectories. Methodologies to attain this goal can be divided into deterministic and probabilistic approaches. Deterministic approaches are made up of nominal and worst-case techniques, and probabilistic techniques include Sequential Monte Carlo (SMC), Hidden Markov Models (HMM) and others.

The nominal technique gives the aircraft position by propagating estimated states into the future along a single trajectory without considering uncertainties of the state estimate and the prediction [13, 22]. Worst-case techniques assume that an aircraft will perform any of a set of maneuvers and the worst case, defined by the application, is considered for aircraft trajectory prediction. Algorithms based on this idea are too conservative since civilian aircraft rarely perform extreme maneuvers [47]. Overall, deterministic techniques [8, 10, 11, 34] suffer from degraded accuracy due to fact that they don't account for uncertainties and address only specific phase(s) of the flight when predicting aircraft trajectories.

Past efforts addressing aircraft trajectory prediction using probabilistic methods are similar to our approach in a way that they aim at modeling uncertainties to describe potential changes in the future trajectory of an aircraft. However, many of them [23, 24, 26, 42, 50, 51] either lack empirical validation or use a simulated set of aircraft trajectories instead of real host track data in their evaluations.

Aside from accounting for uncertainties in the computation, there is considerable research in identifying potential sources of uncertainties in trajectory prediction [12, 27, 45]. In addition to purely deterministic and probabilistic methods, a hybrid method combining the SMC and the worst-case method was also proposed [15]. Although the majority of the research on trajectory prediction is used toward detecting and resolving conflicts with other aircraft, there is also a number of research papers on the same topic from the standpoint of convective weather and Special Use Airspace (SUA) avoidance [25, 28, 43, 44, 46].

Our work is closer in spirit to research [14, 16, 49] that studied the aircraft trajectory prediction problem with machine learning approach, in which they train the model using historical surveillance data and make predictions using various observations. Choi et al [14] present an approach to predict future motion of a moving object based on its past movement. The approach exploits the similarities of short-term movement behaviors by modeling a trajectory as a concatenation of short segments. Although their approach shares some common ground with ours, they have never applied it to aircraft trajectory prediction. Leege et al. [16] specifically propose a machine learning approach to aircraft trajectory prediction. However, unlike our study, they use a stepwise regression approach to systematically determine which input variables to include in the models based on explanatory power. In his PhD. dissertation, Winder et al. [49] present a framework for designing a hazard avoidance alerting system that is based on a Markov decision process model.

The machine learning approach also inspires our study as with our stochastic technique, we leverage the historical Aircraft Situation Display to Industry (ASDI) [3] track data and aircraft specifications to select among the possible intermediate reference points and predict the aircraft trajectory based on meteorological observations. During the process, we assume that the weather observations are realizations of hidden trajectory segments and the transitions between the underlying hidden segments follow a Markov model.

To summarize, our approach is distinguished from the past efforts in at least one of the following four respects: *i)* We use probabilistic approach by taking the uncertainties into account, which yields higher accuracy *ii)* We consider trajectories as a set of 4D joint cubes, which helps us to unify trajectory segments and associate them with pertinent weather parameters, *iii)* We perform time series clustering on the excessive set of current observations to generate time series to feed into the Viterbi algorithm, which is an efficient way to find the optimal state sequence, and *iv)* We use real host track and aircraft specifications data along with weather observations to validate the effectiveness of our approach.

3. PRELIMINARY AND OVERVIEW

In this section we introduce some preliminary concepts, present the problem, and give an overview of the proposed aircraft trajectory prediction system.

3.1 Concepts

The EUROCONTROL/FAA Action Plan 16 (AP16) defines a(n) (aircraft) trajectory as *"the path a moving aircraft follows through the airspace and can be mathematically described by a time-order set of trajectory (state) vectors or the geometry of the flight path"* in the TP Requirements Engineering Methodology paper [1]. The International Civil Aviation Organization (ICAO)'s definition of trajectory is slightly different; *"the description of the movement of an aircraft, both in the air and on the ground, including position, time, and at least via calculation, speed and acceleration"* [4]. Our definition of trajectory, which we call "original trajectory", inspired by Ayhan et al. [9] is slightly different:

Definition 1. An *original trajectory* of an aircraft is a continuous representation of its motion with 4D spatio-temporal parameters *(latitude, longitude, altitude, and time)*, indicating the exact path, traveled by the aircraft.

Due to the fact that there exists no system that continuously records and stores exact positions of an aircraft's original trajectory, only a discrete set of sample data are recorded and stored which presumably represent a close approximation of the original trajectory. We call this a raw trajectory, which we formally define below.

Definition 2. A *raw trajectory* T of an aircraft is a finite sequence of positions with timestamps sampled from the original trajectory. $T = [p_1, p_2, ..., p_n]$, where each point p is defined by its 4D spatio-temporal parameters *(latitude, longitude, altitude, and time)*.

Depending on sampling strategy, which is the way how the position recording is triggered, discrete set of spatio-temporal parameters are captured as the aircraft moves. Among others, the most widely used sampling strategies are time-driven (e.g once every minute), distance-driven (once every mile), and geometry-driven (e.g. once the aircraft deviates from its heading more than 15 degrees). Our study uses ASDI track data which is a form of time-driven raw trajectory, described in detail in Section 4.

Definition 3. A *reference point* r is a fixed spatial location in the 3D space, that is independent of the trajectory data source. A set of evenly distributed reference points form a *reference system* R.

Our study uses the Global Forecast System (GFS) Rapid Refresh (RAP) weather model's 3D grid network [7] as a reference system, described in detail in Section 4. Figure 1.a shows a 2D partial view of the RAP weather model's grid network, which is the reference model overlaid on top of the western Florida area in Google Earth [5].

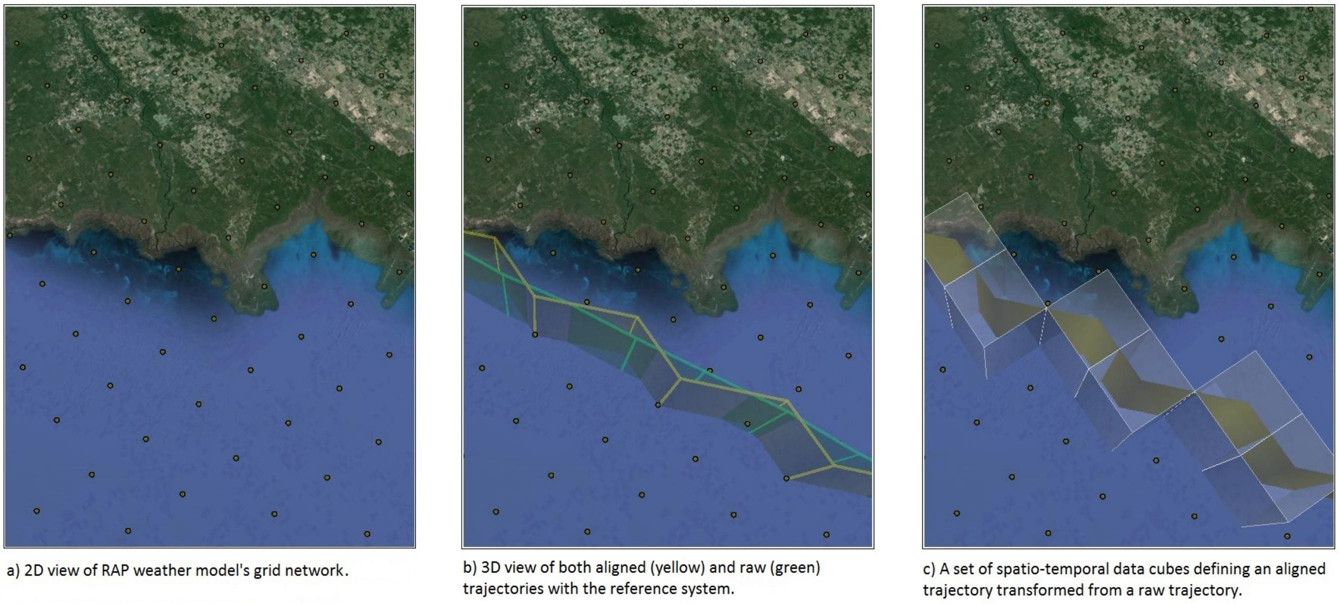
Definition 4. An *aligned trajectory* $\bar{T}$ is a set of reference points that the raw trajectory points are transformed into. More formally, given a *reference system* R, the aligned trajectory $\bar{T}$ for the *raw trajectory* $T = [p_1, p_2, ..., p_n]$ is $\bar{T} = [r_1, r_2, ..., r_n]$, where $r_i \in R$.

Aligned trajectories must preserve the original trajectories as much as possible. Note that erroneous adjustments are introduced each time original trajectories are turned into raw trajectories and raw trajectories are turned into aligned trajectories. Hence, preserving original trajectories in aligned trajectories is obviously a challenging task. This study uses Euclidean distance as a distance function and searches for the nearest neighboring grid point in the reference system to transform raw trajectory points into reference points. Figure 1.b illustrates 3D partial view of both aligned and raw trajectories with the reference system on top of the western Florida area in Google Earth.

Definition 5. A *spatio-temporal data cube* is an atomic trajectory unit in space, defined by its reference point coordinates *(latitude, longitude, altitude)* where weather parameters (temperature, wind speed, and wind direction) are considered to remain constant within a period of time.

Aligning raw trajectories to reference points is a critical preparation step where all necessary parameters are aggregated per point. However, due to uncertainties, we hypothetically create cubes around reference points, building an entire airspace composed of 3D cubes, where each data cube has homogeneous weather parameters within during a certain period of time. This way, trajectories are defined by a set of cubical segments. In our study, the lateral resolution of each cube is $13km$ and temporal resolution is $1hr$. Figure 1.c is an illustration of a set of spatio-temporal data cubes defining an aligned trajectory transformed from a partial raw trajectory.

Definition 6. An *airspace volume of interest* is a 4D airspace volume in which a number of spatio-temporal data cubes are stacked up horizontally and/or vertically.



a) 2D view of RAP weather model's grid network.

b) 3D view of both aligned (yellow) and raw (green) trajectories with the reference system.

c) A set of spatio-temporal data cubes defining an aligned trajectory transformed from a raw trajectory.

Figure 1: Partial illustration of reference system, raw and aligned trajectories, and spatio-temporal data cubes on top of the western Florida area in Google Earth.

3.2 Problem Statement

Given a set of historical raw trajectories for specific aircraft types along with pertinent historical weather observations, we aim at learning a model that reveals the correlation between weather conditions and aircraft positions and predicts trajectories in the form of a time series. In this problem space, we assume that the weather observations are realizations of hidden aircraft positions i.e. trajectory segments and the transitions between the underlying hidden segments follow a Markov model. This assumption considers a finite set of states, each of which is associated with a probability distribution over all possible trajectory segments. Transitions among the states are managed by a set of probabilities. The states are not visible, but the pertinent observations are. Given a sequence of observations, we want to learn an HMM, a statistical Markov model, and derive a sequence of hidden states that correspond to the sequence of observations.

The Viterbi algorithm [48] is an efficient method to compute this. However, in our problem space, the observation sequence for which we want to predict the state sequence is unknown. In other words, given a set of weather observations for the full set of spatio-temporal cubes covering the airspace volume of interest and approximate time period of flight, we need to identify the ones that we can input into the Viterbi process.

3.3 System Overview

Figure 2 shows the overview of the proposed aircraft trajectory prediction system, which generates the optimal state sequence in three steps:

- In the first step, we perform training data processing by generating the HMM parameters based on a set of historical trajectories and weather parameters.
- In the second step, we perform test data processing by executing time series clustering on weather observations for the entire airspace volume of interest which yields the observation sequence.
- In the final step, we use the output of the first two steps in the Viterbi algorithm to generate the optimal state sequence based on the observation sequence.

In Figure 2, data storage is colored yellow, training data processing is colored red, test data processing is colored blue, and the Viterbi process with the final output is colored green.

4. AIRCRAFT TRAJECTORY PREDICTION

This section presents our approach to aircraft trajectory prediction and elaborates in detail each step of the process.

4.1 Training Data Processing

This step enables us to accurately fuse weather parameters per sample point of a raw trajectory. To attain this goal, we need a weather model with the highest spatio-temporal resolution available and that offers both current and historical weather data. The National Oceanic and Atmospheric Administration (NOAA) GFS RAP product operational at the National Center for Experimental Prediction (NCEP) meets these requirements. Hence, we use GFS RAP weather model's 3D grid network as our reference system.

Training data processing, colored red in Figure 2, is performed as follows: Based on the reference system, grid-based alignment is performed on the historical raw trajectories. The process is based on the simple idea of finding the nearest 3D reference point for each sample point of a raw trajectory and then mapping the original sample point to a nearest 3D reference point, inspired by the *Realm* method [19]. More precisely, each sample point in a raw trajectory is aligned to a nearest 3D reference point. This process generates aligned trajectories for historical trajectory data. Although, this seems to suffer from accuracy, it allows to form a unified set resulting in increased similarity between aligned trajectory

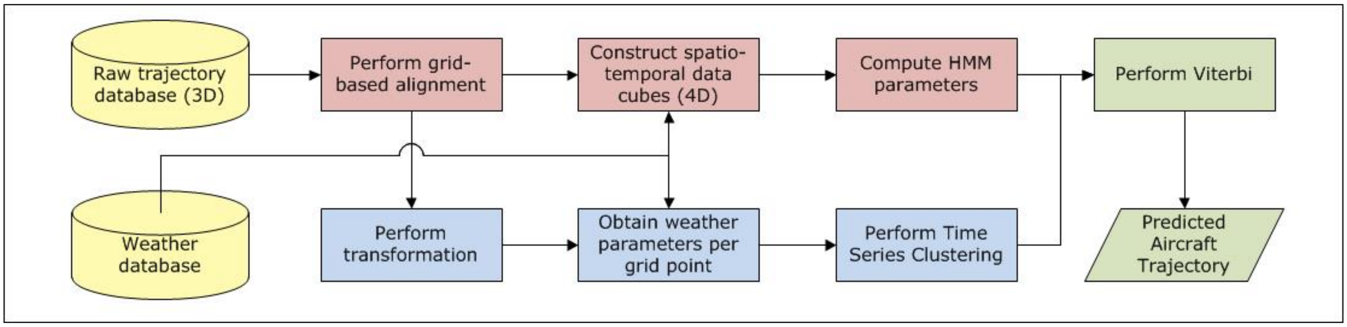

Raw trajectory database (3D)
Perform grid-based alignment
Construct spatio-temporal data cubes (4D)
Compute HMM parameters
Perform Viterbi
Weather database
Perform transformation
Obtain weather parameters per grid point
Perform Time Series Clustering
Predicted Aircraft Trajectory

Figure 2: System overview. Data storage is colored yellow, training data processing is colored red, test data processing is colored blue, and the Viterbi process with the final output is colored green

points. Spatio-temporal data cubes are formed in the next step. During this step, weather parameters for the pertinent time window are retrieved from the weather database and resampled to generate N buckets with distinct ranges. Then, the weather parameters in distinct buckets are fused with spatial data for each grid point along the aligned trajectory. The process yields training data where each historical raw trajectory becomes a set of 4D joint data cubes.

4.2 Test Data Processing

In order for us to compute the maximum probability of HMM generating the optimal state sequence of $s_1, s_2, ..., s_m$, the observation sequence $o_1, o_2, ..., o_m$ is needed. Although the reference points covering the entire airspace volume of interest are known, we don't know which one of these should be fed into the Viterbi process. To answer this question, we perform time series clustering with Dynamic Time Warping (DTW) on excessive set of observations and generate the input time series.

Test data processing, colored blue in Figure 2, enables us to address this question. In this step, among all reference points between the departure and arrival airports, we consider only the ones that were traversed in the historical trajectories. The first step in the process is to perform transformation on these aligned historical trajectories by replacing the original date per reference point with the current date, presuming the flight is about to depart. In other words, during this step, all aligned historical trajectories are treated as if they are current. In the second step, based on new timestamps, weather parameters are retrieved from the weather database and resampled to generate N buckets with distinct ranges. Next, the weather parameters in distinct buckets are fused with spatial data for each grid point along the aligned trajectory. This step generates hypothetical cubes, in the same way as in the training data processing. The output of this step is an excessive set of spatio-temporal data cubes forming distinct trajectories in the form of a time series.

In the final step of the test data processing, spatial data is omitted and *k-Nearest Neighbors (k-NN)* clustering is performed using DTW on the weather parameters along the time series. The algorithm enforces a *locality* constraint and uses a *LB Keogh* lower bound.

Note that in time series analysis, DTW is an algorithm for measuring similarity between multiple temporal sequences which may vary in time or speed. For simplicity, suppose there are two temporal sequences, Q and C, of length n and m respectively, such that $Q = q_1, q_2, ..., q_i, ..., q_n$ and $C = c_1, c_2, ..., c_i, ..., c_n$. To align two sequences using DTW, n-by-m matrix is constructed where the (i^{th}, j^{th}) element of the matrix contains the distance $d(q_i, c_j)$ between the two points q_i and c_j. Each matrix element (i, j) corresponds to the alignment between the points q_i and c_j. A *warping path* W is a contiguous set of matrix elements that defines a mapping between Q and C. During the process, the warping path is subject to a *locality* constraint. There are exponentially many warping paths that satisfy this constraint. We are only interested in the path that minimizes the warping cost. This path can be found using dynamic programming to evaluate the following recurrence, which defines the cumulative distance $\gamma(i, j)$ as the distance $d(i, j)$ found in the current cell and the minimum of the cumulative distances of the adjacent elements:

$$\gamma(i,j) = d(q_i, c_j) + min\{\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)\}$$

By enforcing *locality* constraint through a threshold window, the algorithm ensures that it is unlikely for d_i and t_j to be matched if i and j are too far apart. In this algorithm, the number of clusters is set apriori to 1 and similar time series are clustered together. The process yields an observation sequence in the form of a set of weather parameters which are fed into the Viterbi process, in the next step.

4.3 HMM Processing and Viterbi

We approach the HMM problem by identifying the parameter set computed based on training data as follows:

- States $S = \{S_1, S_2, ..., S_K\}$ are represented by reference points' coordinates *(latitude, longitude, altitude)* that form aligned trajectories.
- Transition probabilities $A = \{a_{ij}\}$, $1 \leq i, j \leq K$, i.e. a_{ij} is the probability of an aircraft discretely transitioning from one state S_i to another S_j along its aligned trajectory, $\bar{T}$.
- Emission probabilities $B = \{b_i(o)\}$, $1 \leq i \leq K$ is the probability of discrete weather parameters having been observed at a specific state, S_i.
- Initial probabilities $\pi = \{\pi_i\}$, $1 \leq i \leq K$ is the probability of an aligned trajectory beginning at a specific state, S_i.

These parameters form an HMM, denoted by $\lambda = \{S, A, B, \pi\}$. The next step in the process is to choose a corresponding state sequence $s = \{s_1, s_2, ..., s_m\}$, that best explains the observation sequence, $O = \{o_1, o_2, ..., o_m\}$. To answer this question, we use the Viterbi algorithm [48], which has a recursive approach that works in parallel for all states in a

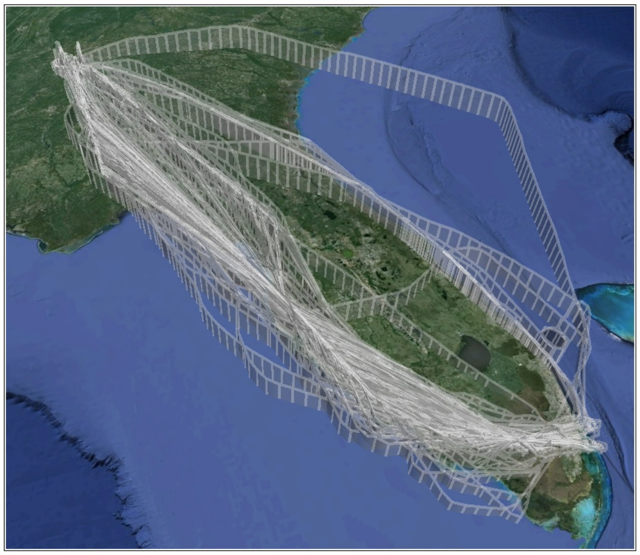


Figure 3: 3D raw trajectories of flight DAL2173 between Atlanta and Miami for the period of May 2010 through December 2011 in Google Earth

strictly time synchronous manner. The key component in the algorithm is the *optimal probability*, denoted as $\delta_m(i)$, which is the maximal probability of HMM generating the observation segment $o_1, o_2, ..., o_m$, along the optimal state sequence $s_1, s_2, ..., s_m$, in which $s_m = i$. Hence, we compute:

$$\delta_m(i) = \max_{s_1,...,s_{m-1}} \pi_{s_1} b_{s_1}(o_1) \prod_{j=2}^{m} (a_{s_{j-1},s_j} b_{s_j}(o_j))$$

The process colored green in Figure 2 yields the predicted aircraft trajectory.

5. EXPERIMENTAL EVALUATION

In this section, we conduct experiments to validate the effectiveness of our aircraft trajectory prediction system. Section 5.1 introduces the experimental dataset. Section 5.2 presents the implementation details and our environment. Section 5.3 discusses the results for our prediction system, and Section 5.4 outlines the discoveries gained by our trajectory prediction system.

5.1 Experimental Dataset

Our empirical evaluation used real host track data along with real weather data: The Delta Airlines' flights DAL2173, departing from Hartsfield-Jackson Atlanta International Airport (ATL) and arriving at Miami International Airport (MIA) for the period of May 2010 through December 2011 were studied. The dataset has a total of 594 trajectories and 56752 points. ATL to MIA is one of the major routes in the NAS due to fact that the departure airport is the busiest airport in the U.S. and the flights are prone to frequent convective weather in the airspace controlled by three Air Route Traffic Control Centers (ARTCC), Atlanta (ZTL), Jacksonville (ZJX), and Miami (ZMA). Figure 3 shows 3D raw trajectories of flight DAL2173 between Atlanta and Miami for the period of May 2010 through December 2011.

Two main data sources to our Aircraft Trajectory Prediction System are the FAA's ASDI and NOAA's RAP data which are colored yellow in Figure 2 and presented next.

5.1.1 ASDI

The source of the raw trajectory data is ASDI, which is a continuous stream of messages delivered over a TCP/IP network socket. Note that a total of over 30 million ASDI messages are processed in any single day. ASDI messages can be flight plan related data, oceanic reports, or host track reports. The host track data is recorded once in every 60 seconds, and provided in near real-time by the FAA. Upon processing, the data is stored in a relational database and made available for use by our trajectory prediction system. Note that surface data is not included in the ASDI feed, although departure and arrival airports remain the same, the positions of the first and last track records may differ over time. The raw trajectory data is generated by joining various ASDI message types including: Track Information and Flight Plan Information. The process generates *source center, date, time, aircraft ID, speed, latitude, longitude, altitude* values from the Track Information and *special aircraft qualifier* from the Flight Plan Information message type.

5.1.2 RAP

The source of the weather data is NOAA National Centers for Environmental Prediction (NCEP) Rapid Refresh (RAP) [7] which is the continental-scale NOAA hourly-updated assimilation/modeling system. RAP covers North America and is comprised primarily of a numerical forecast model and an analysis/assimilation system to initialize that model. It has 13km horizontal resolution with 50 vertical levels, ptop=10hPa, sigma vertical coordinate. Although there is an experimental 3km hourly updated nest inside of the 13km Rapid Refresh, it is not widely available yet. The RAP weather data is stored as a set of *grib2* files [6] each hour and made available for use by our prediction system.

5.2 Implementation

The ASDI host track data is composed of radar recordings at an approximate rate of once every 60 seconds by each Air Route Traffic Control Center (ARTCC) for any aircraft operating under Instrument Flight Rules (IFR) within the confines of an ARTCC's airspace. Our airspace volume of interest is controlled by three separate ARTCCs, due to fact that an aircraft traveling between Atlanta and Miami airports uses Atlanta ARTCC (ZTL), Jacksonville ARTCC (ZJX), and Miami ARTCC (ZMA). The ASDI host track data contains multiple recordings when the aircraft crosses a boundary between multiple ARTCCs. Hence, our initial implementation step filtered these multiple recordings by taking the timestamp and source center (ARTCC) information into account. The filtering process reduced the number of historical recordings from 56752 to 48694. As in rare cases, ARTCCs recorded multiple track data within a relatively short period of time, such as 20 seconds, we also filtered these recordings. The process brought the number of historical recordings down to 44824.

The remaining historical raw trajectory points went through alignment process as elaborated in Section 4.1. During the process, we used GFS RAP weather model's 3D grid network as our reference model, and mapped the raw trajectory points to the nearest reference points. This pro-

Table 1: Buckets for weather parameters

Temperature ($temp$)		
No	Bucket	Value ($kelvin$)
1	$temp \leq 220$	220
2	$220 < temp \leq 240$	240
3	$240 < temp \leq 260$	260
4	$260 < temp \leq 280$	280
5	$280 < temp \leq 300$	300
6	$300 < temp \leq 350$	350

Wind speed (ws)		
No	Bucket	Value ($knots$)
1	$ws \leq 30$	30
2	$30 < ws \leq 60$	60
3	$60 < ws \leq 90$	90
4	$90 < ws \leq 120$	120
5	$120 < ws \leq 150$	150

Wind direction (wd)		
No	Bucket	Value ($degrees$)
1	$wd \leq 45$	45
2	$45 < wd \leq 90$	90
3	$90 < wd \leq 135$	135
4	$135 < wd \leq 180$	180
5	$180 < wd \leq 225$	225
6	$225 < wd \leq 270$	270
7	$270 < wd \leq 315$	315
8	$315 < wd \leq 360$	360

cess yielded aligned historical trajectories. In the next step, spatio-temporal data cubes were formed and reference points along aligned trajectories were fused with weather parameters. However, due to lack of historical weather data for the flight time window of interest of some aligned trajectories, we were not able to expand their dimensions. Subsequently these aligned historical trajectories were eliminated bringing the total number of reference points down to 37849. Fusing positional data with weather parameters yielded the following attributes per spatio-temporal data cube: *source center, date, time, aircraft id, speed, latitude, longitude, altitude, aircraft type, temperature, wind speed, wind direction.* As the final step of the training data processing, we split the weather parameters into buckets as shown in Table 1.

Next, we computed the following HMM parameters:

- 3698 distinct states, S were generated.
- A sparse transition matrix, A of size 3698 x 3698 = 13675204 was generated.
- An emission matrix, B of size 117 x 3698 = 432666 was generated.
- An initial matrix, π of size 28 x 1 was generated.

Aside from these HMM parameters, we also needed to feed the observation sequence into the Viterbi algorithm.

To evaluate our prediction system, we performed bootstrapping by drawing many trajectory samples with replacement from the historical trajectories. In the meantime, we removed the pertinent track records from the training dataset. The process allowed us to plot 95% confidence interval (CI) for the mean error and standard deviation of our trajectory samples. Note that one of the trajectory samples we draw happened to be May 14, 2011. Hence, we present the process for this particular trajectory sample. The test data processing repeated itself for other trajectory samples.

Test data processing for this particular trajectory sample is as follows: We used aligned trajectories for the same historical data. However, this time we treated each trajectory as if it was flown on May 14, 2011. This required a simple transformation of replacing date value per aligned trajectory point. The next step in the process required that we obtain weather parameters for the positional data. However, in order for us to do that, we needed to identify the time period of the flight. Given the median duration of the flight DAL2173 being 78 minutes, we decided to use two *RAP grib2* weather files, one targeting weather observations recorded at 15:00 UTC and the other targeting weather observations recorded at 16:00 UTC on May 14, 2011. By using these files, we retrieved the pertinent weather parameters per aligned trajectory point. Next, we split the weather parameters into buckets as shown in Table 1. In the final step of test data processing, we performed time series clustering, using DTW. The input to the process was 474 time series of 78 weather observations, where each observation contained *temperature, wind speed, wind direction* parameters. The *k-nearest neighbors (k-NN)* clustering with k=1 used DTW Euclidean distance for similarity measure. To speed up the process, we used *LB Keogh* lower bound. The resulting set of cluster centroids identified by weather parameters defined the observation sequence Y_s.

As the final step of the process, we fed the observation sequence Y_s and HMM parameters, $\lambda = \{S, A, B, \pi\}$ into the Viterbi algorithm. The algorithm worked in parallel for all states in a strictly time synchronous manner and returned the optimal state sequence with the maximum probability.

Our experiments were conducted on a computer with Intel Core i7-3840QM CPU @ 2.80GHz and 16GB memory, running on Linux Ubuntu 14.04.2 64-bit LTS Operating System. All the algorithms were implemented in Python v2.7.

5.3 Results

We evaluated the effectiveness of our prediction system based on bootstrapping by drawing 23 trajectory samples with replacement from the historical trajectories. This way, we compared our prediction with the ground truth, flight DAL2173's aligned trajectory. Next, we computed the mean error and standard deviation per trajectory sample. Then, we ranked order the means to estimate the 2.5 and 97.5 percentile values for 95% CI.

Figure 4 shows actual flown trajectory in red overlaid on top of spatio-temporal data cubes in white predicted by our system for one of the trajectory samples drawn, May 14, 2011 as part of the bootstrapping. Major prediction errors are introduced around top of climb and descent phases.

Our quantitative evaluation is based on trajectory prediction accuracy metrics, including horizontal, along-track, cross-track, and vertical errors, as outlined in [18, 31]. The errors illustrated in Figure 5 are based on the coordinates of the aircraft, denoted as AC, and a trajectory segment containing the points TJ_1 and TJ_2.

Table 2 captures mean error μ and standard deviation σ for horizontal, vertical, along-track, and cross-track errors for the entire set of trajectory samples with 95% CI. The errors were computed based on our prediction versus flight DAL2173's raw trajectory per trajectory sample.

The horizontal error is unsigned whereas the along-track and cross-track errors are signed errors. Note that the mean value for the cross-track error $\mu(e_{cross})$ along the entire tra-

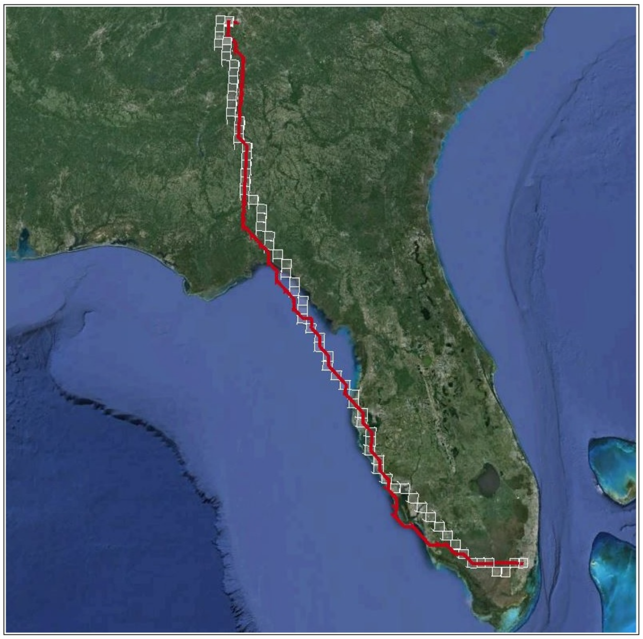


Figure 4: Actual flown trajectory overlaid on top of spatio-temporal data cubes generated for flight DAL2173 on May 14, 2011 by our prediction system

jectory of flight DAL2173 is 6.804nm (=12.601km), when the sign is omitted. Due to fact that spatial resolution of our trajectory prediction system is 13km, which is the horizontal length of each spatio-temporal cube, we can conclude that the mean value $\mu(e_{cross})$ for the cross-track error of 12.601km is within the boundaries of our spatial resolution.

Figure 6 illustrates the pertinent histograms for horizontal, vertical, along-track and cross-track errors for the entire set of trajectory samples with 95% CI. Due to fact that the along-track and cross-track errors are signed errors, they generate a relatively symmetric distribution about the zero, as shown in Figure 6. The area of the histograms provides an indication of overall performance.

To evaluate behavior of trajectory prediction errors over time, we also computed horizontal, along-track, and cross-track errors for the entire set of trajectory samples with 95% CI based on look-ahead time, as shown in Figure 7. Generally, the longer the look-ahead time, the greater the uncertainty, yielding larger errors. However, this is not the case for our system, as shown in Figure 7.

As part of the time-based evaluation, we compared estimated time of arrival initially entered by AOC, predicted time of arrival computed by our system and the actual time of arrival for flight DAL2173.

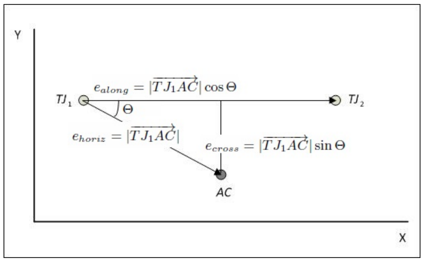

Y
$e_{along} = |\overrightarrow{TJ_1AC}| \cos\Theta$
TJ_1
Θ
TJ_2
$e_{horiz} = |\overrightarrow{TJ_1AC}|$
$e_{cross} = |\overrightarrow{TJ_1AC}| \sin\Theta$
AC
X

Figure 5: Trajectory prediction accuracy metrics

Table 2: Mean and standard deviation values for horizontal, vertical, along-track, and cross-track errors for flight DAL2173 on trajectory samples

Mean error (μ)			
$\mu(e_{horiz})$	$\mu(e_{along})$	$\mu(e_{cross})$	$\mu(e_{vert})$
12.965nm	1.454nm	-1.859nm	687.497ft
Standard deviation (σ)			
$\sigma(e_{horiz})$	$\sigma(e_{along})$	$\sigma(e_{cross})$	$\sigma(e_{vert})$
12.093nm	13.833nm	10.836nm	4910.691ft

Although our prediction system considers time of last ASDI track record as arrival time, there is an additional 4 minutes in average for the aircraft to actually land. By taking this offset into account, our predicted arrival time virtually matches the actual arrival time. Accurate arrival time prediction is a critical capability in maintaining safe separation and sequencing of aircraft.

5.4 Discoveries

In this section, we present the value our system offers in an operational view of flight planning.

The majority of other ground-based tactical TP tools communicate with the aircraft periodically throughout the flight, obtain and update the following parameters, and predict trajectories: *i*) initial conditions (aircraft's 4D position), *ii*) flight plan, *iii*) aircraft performance model, *iv*) weather observations, and *v*) aircraft intent.

As with proprietary implementations, their prediction accuracy depends on correct initial state values and look-ahead time. Hence, they require frequent parameter updates from the aircraft. Each update requires datalink communication throughout the flight, resulting in high communication costs. To give a notional idea on communication cost, it is safe to assume that a TP tool communicates with an aircraft once every 10 seconds for position and once every 2 minutes for aircraft intent data where each message costs $0.50, although airlines pay yearly flat rates to data link service providers such as ARINC and SITA that are kept confidential. Based on the assumption, for a single flight of 80-minute duration, the data link communication cost can add up to $260. Depending on the fleet size and frequency and distance of flights, data link communication cost can easily become millions of dollars per year.

Unlike other TP tools, our system inputs only weather observations and predicts a trajectory, based on an HMM that was learned from the historical trajectories for particular aircraft types. Our prediction system is also ground-based and runs offline requiring no datalink communication with the aircraft. As presented in the analysis of mean error versus look-ahead time, the accuracy of a trajectory predicted by our system is independent of look-ahead time. Our system saves the airlines data communications costs, if deployed.

Initial pre-departure flight plans are filed by the AOC and approved by the FAA, when validated. As presented in the analysis of mean errors and standard deviation values, our system predicts aircraft trajectory within the boundaries of our horizontal spatial resolution of 13km. Hence, when deployed by the airlines, AOC can generate an accurate flight plan and file for approval. In return, the FAA can run our system and validate the filed flight plans, when deployed.

As presented in the analysis of arrival times, our system predicted virtually the same arrival time of flight DAL2173

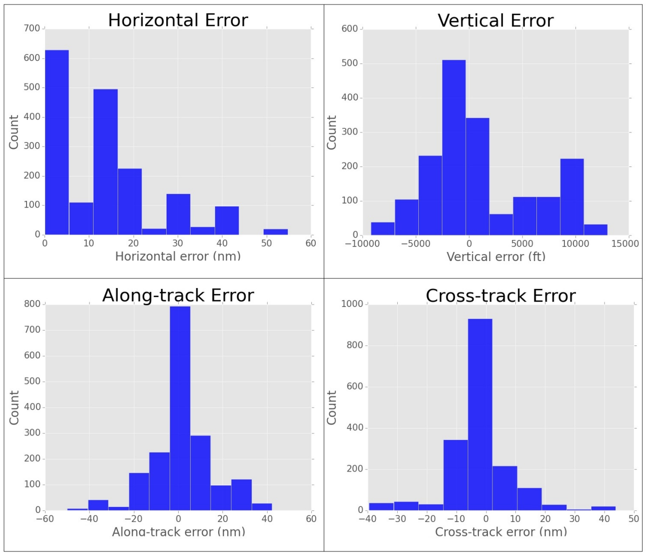


Figure 6: Histograms for horizontal, vertical, along-track, and cross-track errors

as the actual arrival time. The accuracy would help in maintaining separation and sequencing of aircraft in the NAS. Better maintained separation and sequencing means: i) higher safety and capacity, ii) accurate arrival metering which translates to efficiency and so fuel savings and reduced emissions. An erroneously predicted arrival time would lead the Air Traffic Controller to allot more time or insufficient time and airspace for the flight, and generate the incorrect landing sequence potentially reducing runway acceptance rate and causing other aircraft to be perturbed in their operations resulting in reduced efficiency, waste of resources, and possibly reducing safety. Note that the aircraft type Delta Airlines flown for DAL2173 on May 14, 2011 is Boeing 757-200 and operational cost of this particular aircraft per hour is $8,383.15 [2], or $139.72 each minute of extra flight time due to erroneous arrival prediction plus the cost of the impact on other perturbed dependent flights.

6. CONCLUSIONS

The role and performance of trajectory prediction system is critical to the success of the DST functions which have substantial impact on ATM and airspace flow management. In this paper, we have proposed a novel approach to aircraft trajectory prediction that can be used for more efficient and realistic flight planning by aircraft operators and to assist airspace flow management, potentially achieving higher safety, capacity, and efficiency and commensurate fuel savings and so emission reductions for a better environment.

Our evaluation on a real trajectory dataset verified that our prediction system achieved horizontal accuracy of $12.601km$ which is defined as a mean cross-track error $\mu(e_{cross})$ that is within the boundaries of highest spatial resolution, our trajectory prediction system has to offer. In order for us to achieve higher accuracy, we need weather observations of higher spatial resolution than $13km$, which is the current highest spatial resolution available. We plan to investigate the feasibility of resampling and intelligent interpolation of weather data to generate higher spatial resolution. Another option would be to use the recent High-Resolution Rapid Refresh (HRRR) weather product, which offers $3km$ of spatial resolution for a smaller geographic region, yet still experimental by the NOAA. The introduction of a browsing capability similar to what is available in the spatial domain (e.g., [17, 35]) is also of interest.

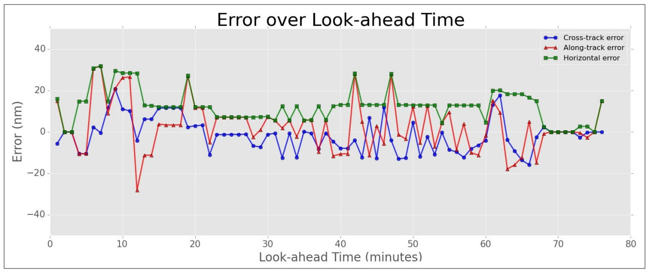


Figure 7: Horizontal, along-track, and cross-track mean errors over look-ahead time

7. ACKNOWLEDGMENTS

The authors would like to thank Milad Gholami and Alexander Wainwright for their assistance with this research. This work was supported in part by the NSF under Grants IIS-12-19023 and IIS-13-20791.

8. REFERENCES

[1] Action plan 16: Common trajectory prediction capability. https://acy.tc.faa.gov/cpat/tjm/TP_Requirements_Engineering_Methodology.pdf.

[2] Aircraft operating expenses. http://www.opshots.net/2015/04/aircraft-operating-series-aircraft-operating-expenses/.

[3] Aircraft situation display to industry. http://www.fly.faa.gov/ASDI/asdi.html.

[4] Definition of an aircraft trajectory by icao. http://ap16.atmrt.org/.

[5] Google earth. https://www.google.com/earth//.

[6] Grib2 documentation. http://www.nco.ncep.noaa.gov/pmb/docs/grib2/grib2_doc.shtml.

[7] Ncep wmo grib2 documentation. http://www.nco.ncep.noaa.gov/pmb/docs/grib2/grib2_doc.shtml.

[8] G. Avanzini. Frenet-based algorithm for trajectory prediction. *Journal of Guidance, Control, and Dynamics*, 27(1):127–135, January 2004.

[9] S. Ayhan and H. Samet. Diclerge: Divide-cluster-merge framework for clustering aircraft trajectories. In *Proceedings of the 8th ACM SIGSPATIAL IWCTS*, Seattle, WA, November 2015.

[10] J. Benavides, J. Kaneshige, S. Sharma, R. Panda, and M. Steglinski. Implementation of trajectory prediction function for trajectory based operations. In *AIAA Atmospheric Flight Mechanics Conference*, Atlanta, GA, 2014.

[11] J. Bronsvoort, G. McDonald, J. Lopez-Leones, and H. Visser. Improved trajectory prediction for air traffic management by simulation of guidance logic and inferred aircraft intent using existing data-link technology. In *AIAA GNC Conference and Exhibit*, Minneapolis, MN, August 2012.

[12] E. Casado, C. Goodchild, and M. Vilaplana. Identification and initial characterization of sources of uncertainty affecting the performance of future trajectory management automation systems. In *Proceedings of the 2nd Int'l Conference on ATACCS*, pages 170–175, Toulouse, France, May 2012.

[13] G. Chatterji. Short-term trajectory prediction methods. In *AIAA GNC Conference and Exhibit*, Portland, OR, August 1999.

[14] P. Y. Choi and M. Hebert. Learning and predicting moving object trajectory: a piecewise trajectory segment approach. Technical report, Carnegie Mellon University School of Computer Science, Pittsburgh, PA, August 2006.

[15] E. Crisostomi, A. Lecchini-visintini, and J. Maciejowski. Combining monte carlo and worst-case methods for trajectory prediction in air traffic control: A case study. *Journal of Guidance, Control and Dynamics*, 43(4), 2008.

[16] A. de Leege, M. van Paassen, and M. Mulder. A machine learning approach to trajectory prediction. In *AIAA GNC Conference and Exhibit*, Boston, MA, August 2013.

[17] C. Esperança and H. Samet. Experience with SAND/Tcl: a scripting tool for spatial databases. *Journal of Visual Languages and Computing*, 13(2):229–255, April 2002.

[18] C. Gong and D. McNally. A methodology for automated trajectory prediction analysis. In *AIAA GNC Conference and Exhibit*, Providence, RI, August 2004.

[19] R. H. Güting and M. Schneider. Realm-based spatial data types: The rose algebra. *The VLDB Journal*, 4(2):243–286, April 1995.

[20] E. Jacox and H. Samet. Metric space similarity joins. *ACM Transactions on Database Systems*, 33(2):7, June 2008.

[21] E. Keogh and C. A. Ratanamahatana. Exact indexing of dynamic time warping. *Knowl. Inf. Syst.*, 7(3):358–386, March 2005.

[22] J. Krozel and D. Andrisani. Intent inference and strategic path prediction. In *AIAA GNC Conference and Exhibit*, San Francisco, CA, August 2005.

[23] Y. Liu and X. R. Li. Intent based trajectory prediction by multiple model prediction and smoothing. In *AIAA GNC Conference and Exhibit*, New Orleans, LA, January 2015.

[24] I. Lymperopoulos, J. Lygeros, and A. Lecchini. Model based aircraft trajectory prediction during takeoff. In *AIAA GNC Conference and Exhibit*, Keystone, CO, August 2006.

[25] S. Mondoloni. A genetic algorithm for determining optimal flight trajectories. In *AIAA GNC Conference and Exhibit*, Boston, MA, August 1998.

[26] S. Mondoloni. A multiple-scale model of wind-prediction uncertainty and application to trajectory prediction. In *6th AIAA Aviation Technology, Integration and Operations Conference*, Wichita, KS, September 2006.

[27] K. T. Mueller, J. A. Sorensen, and G. J. Couluris. Strategic aircraft trajectory prediction uncertainty and statistical sector traffic load modeling. In *AIAA GNC Conference and Exhibit*, Monterey, CA, August 2012.

[28] H. K. Ng, S. Grabbe, and A. Mukherjee. Design and evaluation of a dynamic programming flight routing algorithm using the convective weather avoidance model. In *AIAA GNC Conference and Exhibit*, Chicago, IL, August 2009.

[29] S. Nutanong, E. H. Jacox, and H. Samet. An incremental Hausdorff distance calculation algorithm. *PVLDB*, 4(8):506–517, August 2011.

[30] S. Nutanong and H. Samet. Memory-efficient algorithms for spatial network queries. In *Proceedings of the 29th IEEE Int'l Conference on Data Engineering*, pages 649–660, Brisbane, Australia, April 2013.

[31] M. Paglione and R. Oaks. Implementation and metrics for a trajectory prediction validation methodology. In *AIAA GNC Conference and Exhibit*, Hilton Head, SC, August 2007.

[32] S. Peng, J. Sankaranarayanan, and H. Samet. SPDO: High-throughput road distance computations on spark using distance oracles. In *Proceedings of the 32nd IEEE Int'l Conference on Data Engineering*, pages 1239–1250, Helsinki, Finland, May 2016.

[33] L. R. Rabiner. Readings in speech recognition. chapter A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, pages 267–296. Morgan Kaufmann, San Francisco, CA, 1990.

[34] H. Ryan and M. Paglione. State vector based near term trajectory prediction. In *AIAA GNC Conference and Exhibit*, Honolulu, HI, 2008.

[35] H. Samet, H. Alborzi, F. Brabec, C. Esperança, G. R. Hjaltason, F. Morgan, and E. Tanin. Use of the SAND spatial browser for digital government applications. *Communications of the ACM*, 46(1):63–66, January 2003.

[36] J. Sankaranarayanan, H. Alborzi, and H. Samet. Efficient query processing on spatial networks. In *Proceedings of the 13th ACM Int'l Symposium on Advances in Geographic Information Systems*, pages 200–209, Bremen, Germany, November 2005.

[37] J. Sankaranarayanan, H. Alborzi, and H. Samet. Distance join queries on spatial networks. In *Proceedings of the 14th ACM Int'l Symposium on Advances in Geographic Information Systems*, pages 211–218, Arlington, VA, November 2006.

[38] J. Sankaranarayanan and H. Samet. Distance oracles for spatial networks. In *Proceedings of the 25th IEEE Int'l Conference on Data Engineering*, pages 652–663, Shanghai, China, March 2009.

[39] J. Sankaranarayanan and H. Samet. Query processing using distance oracles for spatial networks. *IEEE Transactions on Knowledge and Data Engineering*, 22(8):1158–1175, August 2010.

[40] J. Sankaranarayanan and H. Samet. Roads belong in databases. *IEEE Data Engineering Bulletin*, 33(2):4–11, June 2010.

[41] J. Sankaranarayanan, H. Samet, and H. Alborzi. Path oracles for spatial networks. *PVLDB*, 2(1):1210–1221, August 2009.

[42] C. Schultz, D. Thipphavong, and H. Erzberger. Adaptive trajectory prediction algorithm for climbing flights. In *AIAA GNC Conference and Exhibit*, Minneapolis, MN, August 2012.

[43] T. Stewart, L. Askey, and M. Hokit. A concept for tactical reroute generation, evaluation and coordination. In *12th AIAA ATIO Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Indianapolis, IN, September 2012.

[44] T. Stewart, J. DeArmon, , and D. Chaloux. Using flight information to improve weather avoidance predictions. In *Aviation Technology, Integration, and Operations Conference*, Los Angeles, CA, August 2013.

[45] S. Swierstra. Common trajectory prediction capability for decision support tools. In *Proceedings of ATM 2003, 5th USA/Europa R&D Seminar*, Budapest, Hungary, June 2003.

[46] C. P. Taylor and C. Wanke. Improved dynamic generation of operationally acceptable reroutes using network optimization. *Journal of Guidance, Control, and Dynamics*, 34(4):961–975, August 2011.

[47] C. Tomlin, G. J. Pappas, and S. Sastry. Conflict resolution for air traffic management: A study in multiagent hybrid systems. *IEEE Transactions on Intelligent Transportation Systems*, 43(4):509–521, April 1998.

[48] A. J. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, April 1967.

[49] L. F. Winder and J. K. Kuchar. *Hazard Avoidance Alerting with Markov Decision Processes*. PhD thesis, Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA, August 2004.

[50] J. L. Yepes, I. Hwang, and M. Rotea. An intent-based trajectory prediction algorithm for air traffic control. In *AIAA GNC Conference and Exhibit*, San Francisco, CA, August 2005.

[51] J. L. Yepes, I. Hwang, and M. Rotea. New algorithms for aircraft intnet inference and trajectory prediction. *Journal of Guidance, Control, and Dynamics*, 30(2):370–382, March 2007.