Predicting Estimated Time of Arrival for Commercial Flights

Samet Ayhan*
University of Maryland
College Park, Maryland
sayhan@cs.umd.edu

Pablo Costas
Boeing Research & Technology Europe
Madrid, Spain
pablo.costas@boeing.com

Hanan Samet
University of Maryland
College Park, Maryland
hjs@cs.umd.edu

ABSTRACT

Unprecedented growth is expected globally in commercial air traffic over the next ten years. To accommodate this increase in volume, a new concept of operations has been implemented in the context of the Next Generation Air Transportation System (NextGen) in the USA and the Single European Sky ATM Research (SESAR) in Europe. However, both of the systems approach airspace capacity and efficiency deterministically, failing to account for external operational circumstances which can directly affect the aircraft's actual flight profile. A major factor in increased airspace efficiency and capacity is accurate prediction of Estimated Time of Arrival (ETA) for commercial flights, which can be a challenging task due to a non-deterministic nature of environmental factors, and air traffic. Inaccurate prediction of ETA can cause potential safety risks and loss of resources for Air Navigation Service Providers (ANSP), airlines and passengers. In this paper, we present a novel ETA Prediction System for commercial flights. The system learns from historical trajectories and uses their pertinent 3D grid points to collect key features such as weather parameters, air traffic, and airport data along the potential flight path. The features are fed into various regression models and a Recurrent Neural Network (RNN) and the best performing models with the most accurate ETA predictions are compared with the ETAs currently operational by the European ANSP, EUROCONTROL. Evaluations on an extensive set of real trajectory, weather, and airport data in Europe verify that our prediction system generates more accurate ETAs with a far smaller standard deviation than those of EUROCONTROL. This translates to smaller prediction windows of flight arrival times, thereby enabling airlines to make more cost-effective ground resource allocation and ANSPs to make more efficient flight schedules.

CCS CONCEPTS

• **Mathematics of computing** → **Multivariate statistics**; • **Information systems** → **Data analytics**; • **Applied computing** → **Aerospace**;

*Senior Engineer at Boeing Research & Technology.

KDD '18, August 19–23, 2018, London, United Kingdom

ACM ISBN 978-1-4503-5552-0/18/08…$15.00
https://doi.org/10.1145/3219819.3219874

Figure 1: Air traffic in Spanish airspace on July 12, 2016.

KEYWORDS

Predictive Analytics, Air Traffic Management, Estimated Time of Arrival, Regression, Recurrent Neural Network

ACM Reference Format:
Samet Ayhan, Pablo Costas, and Hanan Samet. 2018. Predicting Estimated Time of Arrival for Commercial Flights. In *KDD '18: The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, August 19–23, 2018, London, United Kingdom.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3219819.3219874

1 INTRODUCTION

Accurate prediction of Estimated Time of Arrival (ETA) for commercial flights is a key component of collaborative decision making (CDM), a process that attempts to keep the costs under control along with improvement in four key Air Traffic Management (ATM) areas; safety, capacity, efficiency and environmental impact. However, just a seemingly insignificant event such as a delay in obtaining a wheel chair can result in delays as slots are missed and reassigned which can be costly for airlines resulting in increased fuel-burn, emissions and expanded flight hours. This can also cause passenger dissatisfaction and loss of market share due to a hampered corporate image. The impact can grow exponentially due to ripple effect caused by delays in subsequent flights, missed connections, and even disruptions. Hence, due to the nature of unknowns, and complexity of airspace system, it is a challenging task to make an accurate ETA prediction. To illustrate the airspace complexity and volume of air traffic in a single day in Spain, we present Figure 1.

Traditional methods approach the ETA prediction problem deterministically, taking into account aircraft performance models, along with either parametric or physics-based trajectory models. They

usually begin by estimating the flight trajectory, which includes the lateral flight path together with altitude and speed profiles, and then proceed to calculating the time required to fulfill the predicted trajectory. These models by themselves fail to account for external operational circumstances such as weather phenomena, airspace sector densities, and airport congestion, which can directly affect the aircraft's actual flight profile. Note that decreasing the average delay per flight by one minute could save millions of dollars in annual crew costs and fuel savings for a mid-sized airline. Hence, unlike traditional methods, we propose a machine learning based systematic approach to address the ETA prediction problem. We make a considerable effort in feature engineering to use a richer set of features including general information about flights as well as weather, air traffic, and airport data for more accurate modeling and prediction. Our prediction system is built upon the concepts presented in the Aircraft Trajectory Prediction System [5] in which airspace is considered as a set of data cubes around grid points as part of the 3D reference grid network.

The paper contains three main contributions:

- We propose an ETA Prediction System for commercial flights. Unlike other systems that collect and use features only for the arrival airport, we use a richer set of features along the potential route, such as weather parameters and air traffic data in addition to those that are particular to the arrival airport. Our feature construction process generates an extensive set of multidimensional time series data which goes through Time Series Clustering with Dynamic Time Warping (DTW) to generate a single set of representative features at each time instance.
- We present algorithms for two major features that highly contribute to the accurate ETA prediction: airspace sector congestion rate, and airport congestion rate.
- We perform a comparative analysis using a number of regression models and an RNN and present their performances. Using the best performing model, we compare our results with those generated by the EUROCONTROL. Our experiments on real trajectory, weather, air traffic and airport data verify that our system outperforms EUROCONTROL's ETA prediction by offering not only a higher accuracy but also a far smaller standard deviation, resulting in smaller prediction windows of flight arrival times.

The proposed system can be used by the ANSPs and airlines for more accurate scheduling and resource management resulting in improvement in the four ATM key performance areas; safety, capacity, efficiency, and environmental impact. The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 describes the data used in this study followed by Section 4 where we present a proper set of features selected for ETA prediction via feature engineering. Section 5 states the problem and reviews two top-ranking boosting methods, while Section 6 presents our experimental evaluation. Section 7 draws concluding remarks.

2 RELATED WORK

There has been much work and an abundant literature on predicting estimated rate of arrival for humans [23] and ETA for various transportation vehicles on land (road and rail) [9, 25] and water [29], in addition to air. Traditional methods to ETA prediction for air transportation, particularly in the case of commercial flights use a deterministic approach by heavily relying on aircraft performance models along with either parametric or physics-based trajectory models. They usually begin by computing the flight trajectory, which includes the lateral flight path together with altitude and speed profiles, and then proceed to calculate the time required to fulfill the predicted trajectory [7, 21, 22, 24, 42]. However, the time of arrival at a fixed point on the ground is dependent not only on the airspeed the aircraft will fly, but also on the winds, temperatures, air traffic along the route and congestion at the arrival airport. Therefore, if the atmospheric, airspace and airport traffic data is uncertain, the reference trajectory may not have been laid out correctly with respect to the real world situation, yielding inaccurate prediction of estimated time-of-arrival for the specified airport. Probabilistic methods usually develop stochastic linear models of aircraft motion to estimate future aircraft positions. They model aircraft trajectories as Discrete-Time Stochastic Hybrid Linear Systems in which aircraft path decisions form hybrid modes which then inform mode transitions between turning and straight flight modes [17, 31, 43].

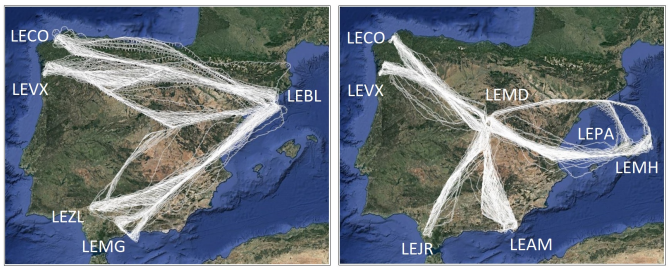
Unlike traditional methods, we propose a machine learning based systematic approach in which general information about the flight as well as weather and air traffic data are all taken into consideration. Our objective is to learn from historical data, possibly stored in an aviation data warehouse [3], discover patterns and generate a set of features to build a model for accurate ETA prediction. There are only a few papers following the machine learning approach to the ETA prediction problem for air transportation in the literature [15, 20, 40].

In their study, Kern et al. [20] aim to enhance ETA predictions generated by the Federal Aviation Administration (FAA)'s Enhanced Traffic Management System (ETMS). They select a set of baseline features and gradually increase complexity by adding new ones to check if they interfere with each other. The features are selected in 4 phases; 1) flight data, 2) flight and weather data, 3) flight and air traffic data, and 4) flight, weather, and air traffic data. Next, they apply Random Forest (RF) to generate the model. According to their results, their model predicts ETA 78.8% more accurately than the FAA's ETMS. Although their model's performance is applaudable, they use weather and air traffic data only for arrival airports during their feature collection, missing all the data along the routes between departure and arrival airports.

In his paper, Takacs [40] presents a solution to the General Electric's Flight Quest contest to make flights more efficient by improving the accuracy of arrival estimates. The contestant presents his prediction approach to runway and gate arrival times of en route U.S. domestic flights based on flight history, weather, air traffic control, and other data available at a given time. His approach follows 6 consecutive stages of ridge regression and gradient boosting machines, trained on a total of 56 features. The model structure is a result of a heuristic, hand-run optimization process. The workflow iteratively defines features and tries to incorporate the modeling to decrease the RMSE as much as possible. However, due to time constraints imposed by the contest, the author decided to keep the model simple by using only 56 features.

Table 1: A set of major air routes in Spain.

DepartureAirport	*ArrivalAirport*
Barcelona–El Prat Airport (LEBL)	A Coruña Airport (LECO) Málaga Airport (LEMG) Vigo–Peinador Airport (LEVX) Seville Airport (LEZL)
Adolfo Suárez Madrid-Barajas Airport (LEMD)	Almeria Airport (LEAM) A Coruña Airport (LECO) Jerez Airport (LEJR) Menorca Airport (LEMH) Palma de Mallorca Airport (LEPA) Vigo–Peinador Airport (LEVX)



a) Flights departing from LEBL. b) Flights departing from LEMD.

Figure 2: A set of trajectories for flights departing from Barcelona and Madrid.

Glina et al. [15] use Quantile Regression Forests (QRF), which is an extension of RF as part of ensemble of Classification and Regression Tress (CART), to generate point predictions and pertinent conditional probability distributions for the ETA of individual flights. They validate their model on data from the Dallas/Fort Worth International Airport, obtaining mean absolute errors (MAE) of less than 60 seconds for their estimates of time-to-wheels-on for flights with a distance to the airport less than or equal to 20 nautical miles. Unfortunately, their data pertains to a single airport and it spans the period of only 5 days.

To summarize, our approach is distinguished from past efforts with the following respects: 1) Our system predicts ETA for runway arrival times. 2) Our system predicts ETA before departure, when the flight trajectory is still unknown. In fact, our prediction system doesn't make use of costly flight plans, offering a more cost-effective solution. Hence, our approach addresses the ETA prediction problem strategically over a time horizon of several hours. 3) For our model, we collect and use a richer set of features not only for the arrival airport but also for the airspace along the potential route. 4) We validate our model using an extensive set of real trajectory, weather, and airport data over the period of 11 months.

3 DATA DESCRIPTION

This section introduces the datasets used for ETA prediction.

3.1 Trajectory Data

Trajectory data plays a large role in our study. Trajectories have been the subject of much work in the spatial domain with an emphasis on cars along roads [37]. The focus has been on their generation (e.g., [38]), queries (e.g., [28, 30, 33, 35, 36]), and matching (e.g., [18, 27, 34]). This data is collected continuously and is quite voluminous. Instead, our focus here is on the flight domain.

Due to the fact that there exists no system that continuously records and stores exact positions of an aircraft's original trajectory [4], 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. The raw trajectory data is provided by Spanish ANSP, ENAIRE, using radar surveillance feed with a 5 seconds update rate. The raw data is wrangled as part of the Data-driven AiRcraft Trajectory prediction research (DART) project under the SESAR Joint Undertaking Work Programme [39].

The data contains all commercial domestic flights for Spain, a total of 119,563 raw trajectories and 80,784,192 raw trajectory points for the period of January through November 2016. The fields of the raw trajectory data are as follows: *Flight No, Departure Airport, Arrival Airport, Date, Time, Aircraft Speed* in *X, Y, Z* directions, and position information *(Latitude, Longitude, Altitude)*. Note that, as a preprocessing step, we downsample raw trajectory data from the original resolution of 5 seconds to 60 seconds and align them to our 3D reference grid [1]. This is to build a set of features for each trajectory point, i.e. weather observations along the aligned trajectories. Table 1 lists a set of major flight routes departing from two major cities in Spain and Figure 2 provides their visual representation on a map.

3.2 Meteorology Data

The meteorology data is obtained from the National Oceanic and Atmospheric Administration's (NOAA) Global Forecasting System (GFS) [26]. The original data has 28-km spatial and 6-hour temporal resolution and it contains over 40 weather parameters including *Atmospheric, Cloud* and *Ground* attributes for each grid point as part of its 3D weather model. Hence, for this study's geographic volume of interest for the time period of January through November 2016, over 80TB of weather data is collected.

3.3 Airport Data

The airport traffic data is provided by EUROCONTROL as part of the Data-driven AiRcraft Trajectory prediction research (DART) project under the SESAR Joint Undertaking Work Programme [39]. The data contains airport traffic data for Spain, a total of 1,252,571 sets of records, where each record is composed of *Date, Flight No, Departure Airport, Arrival Airport, Aircraft type, Actual Departure Time, Actual Arrival Time, Scheduled Departure Time* and *Scheduled Arrival Time*.

3.4 Airspace Data

The airspace sector data is computed using raw trajectory data along with airspace sector volumes. The data contains aircraft counts within 15-minute bins for each sector in Spain for a period of 11 months. The records include *Sector Name, Date, Time of Sector Entry* and *Time of Sector Exit* with counts of aircraft.

Table 2: The description of features.

Feature Type	Feature	Description
Flight	Airline	Name of airline
	Flight no	Flight number
	Aircraft type	Type of aircraft
Spatial	Latitude	Latitude of an occurrence
	Longitude	Longitude of an occurrence
	Altitude	Altitude of an occurrence
	Sector	Boundaries of a 3D airspace volume
Temporal	Date	Date of an occurrence
	Time	Time of an occurrence
	Time bin	15-minute time bin, an occurrence falls into
Meteorological	Atmospheric temperature	Atmospheric temperature recording along the flight route
	Atmospheric wind speed	Atmospheric wind speed recording along the flight route
	Atmospheric wind direction	Atmospheric wind direction recording along the flight route
	Atmospheric humidity	Atmospheric humidity recording along the flight route
	Atmospheric pressure	Atmospheric pressure recording along the flight route
	Ground temperature	Ground temperature recording at the arrival airport
	Ground wind speed	Ground wind speed recording at the arrival airport
	Ground wind direction	Ground wind direction recording at the arrival airport
	Ground humidity	Ground humidity recording at the arrival airport
	Ground pressure	Ground pressure recording at the arrival airport
	Ground wind gust	Ground wind gust recording at the arrival airport
Airport	Arrival airport	Airport the flight arrives at
	Departure airport	Airport the flight departs from
	Airport arrival count	Number of arrivals at an airport
	Airport departure count	Number of departures from an airport
	Airport congestion rate	Current vs. historical arrival+departure counts
Airspace	Sector aircraft count	Number of aircraft in a sector along the flight route
	Sector congestion rate	Current vs. historical aircraft count in a sector
Combinational	Flight-spatial	Traversed locations of particular flights
	Flight-spatial-temporal-airspace	Traversed locations of particular flights at particular time instances
	Meteorological-temporal	Observed weather parameters at particular time instances
	Airport-temporal	Arrival and departure counts of aircraft at a particular airport
	Spatial-temporal-airspace	Traversed locations by an aircraft at particular time instances
	Spatial-temporal-meteorological	Observed weather parameters at particular positions and time instances

4 FEATURE ENGINEERING

In this section, we select and construct a proper set of features for ETA prediction via feature engineering. Feature engineering refers to the process of using domain knowledge of data to create features that best represent the underlying problem of the predictive models, resulting in improved model accuracy on unseen data. We adopt high-dimensional features to build models so they possess the ability to predict ETA more accurately. Features extracted from each domain are listed in Table 2.

4.1 Basic Features

Basic features are extracted from each individual domain. We exploit *Airline, Flight No* and *Aircraft Type* as the flight features. The first three letters of *Flight No* identify the *Airline*. Intuitively, each of these basic flight features or combination of them exhibit a distinct pattern, potentially yielding a different ETA. For instance, as *Aircraft Type*, Bombardier Canadair Regional Jet (CRJ) has different performance parameters than that of Boeing 737-800's (B738), likely to generate a different ETA.

We use *Latitude, Longitude, Altitude*, and *Sector* as the spatial features. Aircraft trajectories are nothing more than a set of joint 3D coordinates, where each coordinate is associated with a time stamp. Trajectories are usually of different length, formed by a different set of 3D coordinates. Hence, different spatial features may cause various patterns and biases in ETA prediction.

We adopt *Date, Time*, and *Time Bin* as the temporal features. Note that the temporal features listed in Table 2 are kept at a high level for the sake of saving space. Time bins are used to generate histograms. Note that, these features can refer to a flight departure or arrival or sector crossing as well as observation of a weather parameter. Essentially, ETA is an accumulated set of time intervals along the trajectory. Hence, a trajectory of 60 segments, where each segment is one-minute long, translates to an ETA of departure time plus 60 minutes.

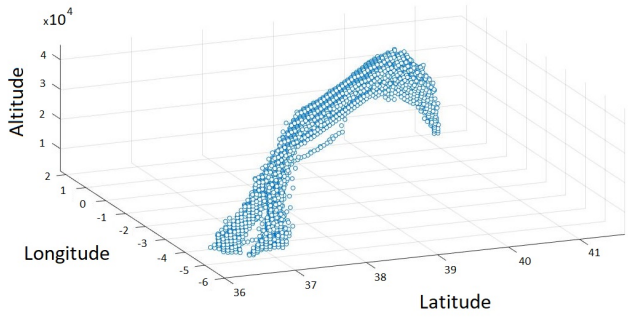


Figure 3: Historically traversed 3D grid points between departure and arrival airports.

Meteorological data is one of the major factors impacting the spatial-temporal patterns of trajectories. Among over 40 weather parameters, we exploit 5 atmospheric features for airspace *Temperature, Wind Speed, Wind Direction, Humidity*, and *Pressure*, and 6 ground features for airports *Temperature, Wind Speed, Wind Direction, Humidity, Pressure*, and *Wind Gust*. Convective weather shaping up along the aircraft's planned route may cause a trajectory deviation yielding a different ETA.

The most recent work that aimed at addressing ETA prediction with a machine learning approach utilized particular features such as weather observations and traffic data for the arrival airport only, disregarding features along the potential trajectory [20]. This is partly due to fact that the exact trajectory is unknown before departure. However, excluding these key features along the potential trajectory greatly degrades the accuracy. Now, a critical question arises: *How can we obtain a set of features along the potential trajectory when the trajectory is still unknown?* Our approach to this problem is as follows: As a preprocessing step, we align raw historical trajectories to the 3D reference grid [5]. The process generates a time series of aligned historical trajectories. Figure 3 illustrates a set of 3D grid points historically traversed by a sample flight between a departure and arrival airport. Using spatially and temporally closest set of data points from the weather model, we perform interpolation and extract the pertinent parameters for each aligned trajectory point. Once weather observations have been extracted for each grid point at a particular time instance, we omit spatial information. Next, we perform Time Series Clustering with Dynamic Time Warping (DTW) [6, 19] on weather observations of variable length. The process aggregates the weather observations and generates a single set of representative features for each time instance. The result is a set of features in the form of time series. For instance, we use 300 meteorological features for a flight that is 60-minutes long. This is due to fact that each trajectory point is recorded once a minute and at each trajectory point 5 meteorological features are adopted along the potential route. The same applies to the sector congestion rates. For the same flight of 60 minutes, we use 4 time bins, resulting in 4 sector density features along the flight route.

Airport features consist of *Arrival Airport, Departure Airport, Airport Arrival Count* and *Arrival Departure Count*, in addition to *Airport Congestion Rate*, which is computed using the actual and scheduled arrival and departure counts. Each of these features are likely to have an impact on flight's ETA, as the higher the airport congestion rate, the higher the likelihood of delay on arrival time.

We exploit *Sector Aircraft Count* and *Sector Congestion Rate* as airspace features. Similar to the impact by the airport congestion rate, the higher the airspace congestion rate along the planned route, the higher the likelihood of trajectory diversion causing a delay on arrival time.

4.2 Combinational Features

We combine various features from two or more domains to build combinational features. Combining flight and spatial features produces *Flight-spatial* combinational feature. Intrinsically, each airline's flight exhibits a distinct trajectory pattern, yielding a different ETA. Figure 4a. shows Ryan Air (RYR) versus Vueling Airline's (VLG) flights between the LEBL and LEZL airports. Red trajectories representing the RYR flight dominates the northern part, while white trajectories representing the VLG flight dominates the southern part of the routes.

Combining meteorological and temporal features produces *Meteorological-temporal* combinational feature. Since weather observations vary over time and ETA is likely to be impacted by the weather patterns, it is intuitive to combine these features. Figure 4b. shows distribution of normalized temperature over the period of January through November 2016 for the flight VLG22XV between the LEBL and LEZL airports. Due to fact that average flight time for the flight VLG22XV is 75 minutes and that each trajectory point is recorded once per minute, each time instance on Figure 4b. has 75 temperature recordings. As shown in the figure, normalized temperature increases during the summer months.

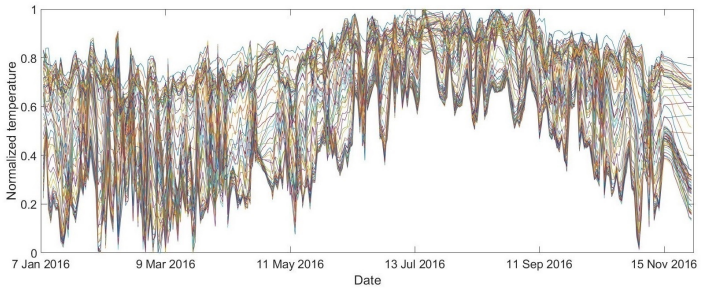
We combine airport and temporal features to generate the *Airport-temporal* combinational feature. This enables us to gain insight on the distribution of aircraft counts on various airports at various time bins during the day, potentially impacting the ETA prediction. Figure 4c. captures the distribution of normalized arrival and departure counts within 15-minute time bins for the LEZL airport. The airport appears to have more traffic at time bins around 8:00.

To examine combination of various features and derive insights on trends, find anomalies and perform strategic planning, a novel interactive visualization tool, NormSTAD [2] can be used.

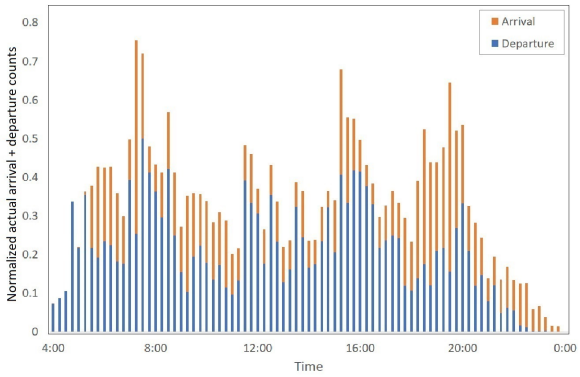
4.3 Airport Congestion Rate

Given the actual (historical) arrivals and departures along with scheduled arrivals and departures for a particular airport, we want to compute the airport congestion rate for the 15-minute time bin, the flight of interest is scheduled to arrive. First, we create a set of 15-minute time bins throughout the day. Next, we compute the average counts for the actual arrival and departures for each of these 15-minute time bins. Given the scheduled arrival time of the flight of interest, we determine which time bin it falls into. For that particular time bin, we compute the airport congestion rate as presented in Algorithm 1. Figure 4d. illustrates congestion rates in each 15-minute bins at the LEZL airport for the flight VLG22XV, indicating a potential delay in late March due to a high congestion rate at the LEZL airport.

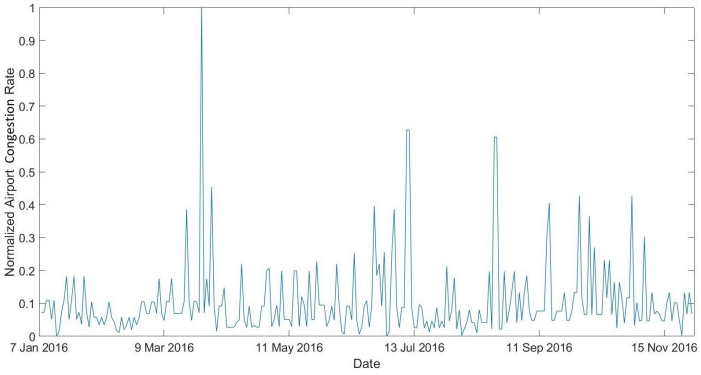
a) Two distinct trajectory patterns by two airlines.



b) Distribution of normalized temperature over time.



c) Distribution of normalized arrival and departure counts over 15-min. bins.



d) Congestion rates over 15-min. bins at LEZL airport for fight VLG22XV.

Figure 4: Distribution of various features.

Algorithm 1: Airport Congestion Rate

Result: Airport congestion rate

Input : Counts of scheduled arrivals $ctSA$, counts of scheduled departures $ctSD$, average count of actual arrivals $avgCtAA$, and average count of actual departures $avgCtAD$ of a flight with a flight number fno and timestamp ts for a particular airport

Output: Congestion rate cr of a particular airport, given the time stamp ts and a flight number fno of an arriving flight fl

$TB \leftarrow [tb_1, tb_2, ..., tb_k]$
$3dp \leftarrow lat, lon, alt$
$fl \leftarrow fno, 3dp, ts$
foreach $tb \in TB$ **do**
  if $fl.ts \subset tb$ **then**
    return $acr[tb] \leftarrow (ctSA[tb] + ctSD[tb])/(avgCtAA[tb] + avgCtAD[tb])$
  end
end

4.4 Sector Congestion Rate

Given the actual (historical) flights' sector crossings, along with scheduled flights' sector crossings for a particular sector, we want to compute the sector congestion rate for each sector, for each 15-minute time bin, the flight of interest is scheduled to cross. Due to fact that airspace sectors are nothing more than a set of extruded polygons, we use a Point-In-Polygon algorithm [1] to compute the counts of aircraft in each sector. To achieve this, we first create a set of 15-minute time bins throughout the day. Next, we compute the average counts for the actual flights' sector crossings for each of these 15-minute time bins. Given the scheduled crossing time of the flight of interest, we determine which time bin it falls into, and which sector it likely crosses. Due to fact that the flight's trajectory is unknown at the time of ETA prediction, we consider all sectors the flight of interest is likely to cross based on actual sector crossings. Next, for that particular time bin, we compute the airport acceptance rate as presented in Algorithm 2.

5 PROBLEM STATEMENT AND MODEL REVIEWS

Given a set of historical flights with their attributes such as flight number, trajectory, departure and arrival airport, actual departure and arrival date and time, airline name, aircraft type, and associated weather and air traffic parameters, we aim at learning a model that predicts the current flight's ETA before it departs. Note that ETA can refer to both runway and gate arrival times. Our system predicts runway arrival times i.e. times between aircraft wheels-off and wheels-on, as we exclude airport surface data when we build our model. To address this problem, we build a number of models, and rank them based upon their performance.

Algorithm 2: Sector Congestion Rate

Result: Sector congestion rate
Input : Counts of scheduled flights' sector crossings $ctSF$, average counts of actual flights' sector crossings $avgCtAF$, for each time bin tb for each sector
Output: Congestion rate of a particular sector scr, given the time stamp ts and a flight number fno of a crossing flight fl

$TB \leftarrow [tb_1, tb_2, ..., tb_k]$
$3dp \leftarrow lat, lon, alt$
$fl \leftarrow fno, 3dp, ts$
foreach $tb \in TB$ **do**
  if $fl.ts \subset tb$ **then**
    return $scr[tb] \leftarrow (ctSF[tb]/avgCtAF[tb])$
  end
end

5.1 AdaBoost

Adaptive Boosting is a meta-estimator to fit a sequence of weak learners on iteratively modified versions of the data. All the predictions are then combined through a weighted sum to produce the final prediction. The data modifications at each iteration consist of applying weights to each of the training samples. The first step simply trains a weak learner on the original data. For each successive iteration, the sample weights are individually modified and the learning algorithm is reapplied to the reweighted data. At a given step, those training examples that were incorrectly predicted by the boosted model induced at the previous step have their weights increased, whereas the weights are decreased for those that were predicted correctly. As iterations proceed, examples that are difficult to predict receive ever-increasing influence. Each subsequent weak learner is thereby forced to concentrate on the examples that are missed by the previous ones in the sequence [16].

Formally, we assign an initial weight $w_i = 1 \quad i = 1, ..., N_1$ to each training pattern. We repeat the following procedure while the average loss $\bar{L}$ is less than 0.5.
(1) The probability that training sample i is in the training set is $p_i = w_i / \sum w_i$, where the summation is over all members of the training set. Select N_1 samples with replacement to form the training set. This may be implemented by marking a line of length $\sum w_i$ and subsection of length w_i. A uniform number picked from the range $[0, \sum w_i]$ and landing in section i corresponds to picking pattern i.
(2) Build a regression machine t from the training set. Each machine makes a hypothesis: $h_i : x \rightarrow y$.
(3) Pass every member of the training set through this machine to obtain a prediction $y_i^{(p)}(x_i) \quad i = 1, ..., N_1$.
(4) Compute a loss for each training sample $L_i = L\left[|y_i^{(p)}(x_i) - y_i|\right]$. The loss L may be of any functional form as long as $L \in [0, 1]$. If we let $D = sup|y_i^{(p)}(x_i) - y_i| \quad i = 1, ..., N_1$ then we have three candidate loss functions:

$$L_i = \frac{|y_i^{(p)}(x_i) - y_i|}{D} \quad (linear)$$

$$L_i = \frac{|y_i^{(p)}(x_i) - y_i|^2}{D^2} \quad (squarelaw)$$

$$L_i = 1 - \exp\left[\frac{-|y_i^{(p)}(x_i) - y_i|}{D}\right] \quad (exponential)$$

(5) Calculate an average loss: $\bar{L} = \sum_{i=1}^{N_1} L_i p_i$
(6) Form $\beta = \frac{\bar{L}}{1 - \bar{L}}$. β is a measure of confidence in the prediction.
(7) Update the weights: $w_i \rightarrow w_i \beta^{[1-L_i]}$. The smaller the loss, the more weight is reduced making the probability smaller that this pattern will be selected as a member of the training set for the next machine in ensemble.
(8) For a particular input x_i, each of the T machines makes a prediction. Obtain the cumulative prediction h_f using the T predictors:

$$h_f = \inf\left\{y \in Y : \sum_{t:h_t \leq y} \log(1/\beta_t) \geq \frac{1}{2} \sum_t \log(1/\beta_t)\right\}$$

This is the weighted median. Equivalently, each machine h_t has a prediction $y_i^{(t)}$ on the ith pattern and associated β_t. For pattern i the predictions are relabeled such that for pattern i we have: $y_i^{(1)} < y_i^{(2)} <, ..., < y_i^{(T)}$ Next, we sum the $\log(1/\beta_t)$ until we reach the smallest t so that the inequality is satisfied. The prediction from that machine t we take as the ensemble prediction. If the β_t were all equal, this would be the median.

Intuitively, the effect of varying the weight w_i to give more emphasis to difficult examples means that each subsequent machine has a disproportionately harder set of examples to train on. Thus, the average loss tends to increase as we iterate through the algorithm and ultimately the bound on L is not satisfied and the algorithm terminates [10, 12]

5.2 Gradient Boosting

Gradient Boosting [13, 14] produces a prediction model in the form of an ensemble of weak prediction models. It builds additive regression models by sequentially fitting a simple parameterized function to current pseudo residuals by least squares at each iteration. The pseudo residuals are the gradient of the loss functional being minimized, with respect to the model values at each training data point, evaluated at the current step.

Formally, in the function estimation problem one has a system consisting of a random output variable y and a set of random input variables $x = \{x_1, ..., x_n\}$. Given a training sample $\{y_i, x_i\}_1^N$ of known (y, x) values, the goal is to find a function $F^*(x)$ that maps x to y, such that over the joint distribution of all (y, x) values, the expected value of some specified loss function $L(y, F(x))$ is minimized:

$$F^*(x) = \underset{F(x)}{\operatorname{argmin}} E_{y,x} L(y, F(x))$$

Boosting approximates $F^*(x)$ by an additive expansion:

$$F(x) = \sum_{m=0}^{M} \beta_m h(x; a_m)$$

Table 3: A list of prediction models used in this study.

Method	*Algorithm*
Linear	Linear Regression (LR)
	Lasso Regression (LASSO)
	Elastic Net Regression (EN)
Non-linear	Classification and Regression Trees
	Support Vector Regression (SVR)
	k-Nearest Neighbors (KNN)
Ensemble	Adaptive Boosting (AdaBoost)
	Gradient Boosting (GBM)
	Random Forest Regression (RF)
	Extra Trees Regression (ET)
Recurrent Neural Network	Long Short-Term Memory (LSTM)

where the functions $h(x; a)$ which are base learners are usually chosen to be simple functions of x with parameters $a = \{a_1, a_2, ...a_k\}$. The expansion coefficients ${\beta_m}_0^M$ and the parameters ${a_m}_0^M$ are jointly fit to the training data in a forward stage-wise manner. One starts with an initial guess $F_0(x)$, and then for $m = 1, 2, ..., M$:

$$(\beta_m, a_m) = \underset{\beta, a}{\operatorname{argmin}} \sum_{i=1}^{N} L(y_i, F_{m-1}(x_i) + \beta h(x_i; a))$$

and

$$F_m(x) = F_{m-1}(x) + \beta_m h(x; a_m)$$

Gradient boosting approximately solves (β_m, a_m) for arbitrary loss functions $L(y, F(x))$ with a two step procedure. First, the function $h(x; a)$ is fit by least-squares:

$$a_m = \underset{a, \rho}{\operatorname{argmin}} \sum_{i=1}^{N} [\tilde{y}_{im} - \rho h(x_i; a)]^2$$

to the current pseudo residuals:

$$\tilde{y}_{im} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)}$$

Then, given $h(x; a_m)$, the optimal value of the coefficient β_m is determined:

$$\beta_m = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{N} L(y_i, F_{m-1}(x_i) + \beta h(x_i; a_m))$$

This strategy replaces a potentially difficult function optimization problem by one based on a least-squares, followed by a single parameter optimization based on a general loss criterion L.

6 EXPERIMENTAL STUDY

This section presents the evaluations of our ETA prediction system.

6.1 Setup

We evaluate the performance of our prediction system on 10 major flight routes in Spain, presented in Table 1. We chronologically order each dataset for each route, normalize and standardize, and use the first 80% for modeling and the remaining 20% for validation. Table 3 lists the prediction models used in this study. We use 10-fold cross-validation and evaluate the algorithms using Root Mean Squared Error (RMSE) metric. We create a baseline of performance on this problem and spot-check a number of algorithms to address the regression problem. The first round of algorithms we use are linear and non-linear regression methods, in addition to historical average (HA), which simply averages all historical flight times for the same routes for the same time period. The algorithms all use default tuning parameters. The second round of algorithms we use are ensemble methods comprised of two boosting (AB and GBM), and two bagging (RF and ET) methods. These algorithms also use default tuning parameters. The final algorithm we use is the Long Short-Term Memory (LSTM), which is a type of RNN. Given the flight times for the previous 10 most recent subsequent days, we compute the time for the next day's flight, provided that all flights share the same defining features such as flight number, departure and arrival airports, aircraft type, etc. In order to perform LSTM on our dataset, we first transform our multivariate time series data into a supervised learning problem. Next, we define the LSTM with 50 neurons in the first hidden layer and 1 neuron in the output layer for predicting the flight time. The input shape is one time step with over 300 features (the number of features depends on the average flight time). The model is fit for 50 training epochs with a batch size of 72.

Our experiments are conducted on a computer with Intel Core i7-6820HQ CPU @ 2.70GHz and 16GB memory, running on Linux Ubuntu 16.04.1 64-bit Operating System. All the algorithms in our system are implemented in Python 3.6.4.

6.2 Results

Table 4 shows the RMSE value for each algorithm. The best score for each route is highlighted. In a quick glance, it seems clear that HA and linear models (LR, LASSO, EN) perform poorly on all routes. KNN and SVR seem to achieve the best scores among the linear and non-linear methods. In fact, KNN generates 8, SVR generates 2 best scores out of 10 routes among the linear and non-linear methods. The default value for the number of neighbors in KNN is set to 7. We obtain slightly better results when we use a grid search to try different numbers of neighbors. Unlike the previous work [15, 20] which emphasize the accuracy of bagging methods (RF and its extensions), our results indicate that boosting methods (AB and GBM) dominate the bagging methods (RF and ET) on all routes. LSTM yields relatively unstable scores, generating only one best score on all 10 routes. This may not be a surprise as LSTM is better suited for sequence prediction than autoregression.

Overall, the boosting methods dominate the entire list, generating the best scores among all including the linear, non-linear, ensemble and RNN methods. In fact, the scores obtained by boosting methods can be further improved with tuning. The default number of boosting stages to perform (n estimators) is originally set to 100. We define a parameter grid n estimators values from 50 to 400 in increments of 50. Each setting is evaluated using 10-fold cross-validation. By parameter tuning, we reach even better scores.

We complete our comparative study among the models listed in Table 3 and select the best score for each route. Next, we compare our final results with the ETA values, EUROCONTROL uses [39].

Table 4: The RMSE values of all algorithms.

Model	*LEBL LECO*	*LEBL LEMG*	*LEBL LEVX*	*LEBL LEZL*	*LEMD LEAM*	*LEMD LECO*	*LEMD LEJR*	*LEMD LEMH*	*LEMD LEPA*	*LEMD LEVX*
HA	5.590140	5.498889	5.555261	4.911651	3.265500	4.000898	3.311410	3.953404	3.787092	4.666325
LR	6.565834	5.583422	6.669648	5.240185	4.848514	4.612558	4.167661	4.433904	4.943242	5.183345
LASSO	5.423176	5.328147	4.566031	4.620595	2.979159	3.939358	3.061945	3.618408	3.830357	4.676572
EN	5.129900	5.328147	4.510600	4.598869	2.979159	3.972221	3.078638	3.618059	3.759207	4.562907
KNN	4.315834	4.041417	4.080471	3.768015	3.768015	3.472992	2.837109	3.109983	3.633360	3.409278
CART	6.208306	5.228796	5.098875	4.717167	4.527573	4.439111	3.729169	3.790538	4.691645	4.175965
SVR	4.850719	5.036133	4.276852	4.320527	2.914718	3.575904	3.001464	3.199346	3.536542	4.151692
AB	**3.991038**	3.790171	4.244822	**3.620588**	**2.840969**	**3.128822**	2.602778	2.987388	3.347820	**3.092941**
GBM	3.993338	**3.430164**	4.068700	3.630516	3.045774	3.276929	**2.589360**	**2.908439**	**3.344821**	3.174045
RF	4.308547	3.841439	4.103479	3.706125	3.016808	3.345249	2.740151	2.910654	3.5721842	3.437595
ET	4.171390	3.538316	4.266809	3.731383	3.081153	3.396700	2.630542	3.028200	3.6377407	3.436975
LSTM	7.586650	5.674340	**3.708545**	4.673223	2.932544	3.714577	2.604398	3.988213	3.789673	4.311236

Figure 5 illustrates RMSE values in minutes for each route between our prediction versus EUROCONTROL's prediction. Note that Figure 5 presents both results at two different scales. Figure 5a. is a closer look at the box plots, where the median values are visible. However, the full extent of the boxplots are missing in Figure 5a. due to outliers. Hence, we provide Figure 5b., where the full extent of the boxplots including the outliers are visible. From the results, we make the following observations: *i*) Our prediction yields better median scores on eight routes, while the EUROCONTROL's ETA shows better median scores on two routes (LEBL-LEVX and LEBL-LEZL). *ii*) The standard deviation values in EUROCONTROL's ETAs are much larger, resulting in larger windows of predictability at arrival times. This may have a significant adverse impact on airlines' ground resource management as well as connecting flights' scheduling. *iii*) Boxplots representing EUROCONTROL's ETAs in Figure 5b. show extreme outliers. In summary, our prediction system offers more accurate ETA prediction with far smaller standard deviation than those of EUROCONTROL.

To evaluate the effectiveness of our features, we rank them based on their relative importance. The top 10 features are presented in Table 5. Evidently, the arrival airport is of critical importance on accurate ETA prediction. Aside from that, meteorological features in the form of a time series along the potential trajectory make a significant impact during regression. This insight validates our approach that meteorological and airspace sector features along the potential trajectories should be taken into account to achieve accurate ETA prediction.

7 CONCLUSION

We have presented a novel ETA prediction system for commercial flights. Our experiments verify that our system can predict any commercial flight's ETA in Spain within 4 minutes of RMSE on average regardless of the flight length. Our system outperforms EUROCONTROL's ETA prediction by offering not only a higher accuracy but also a far smaller standard deviation, resulting in increased predictability of flight arrival times. This enables airlines to better coordinate the action of ground handling personnel and equipment, thereby reducing the waste of time and energy. Likewise, air traffic flow management, airport runway and gate assignment, and ground support equipment usage optimization are all processes that can benefit from a more accurate ETA prediction for commercial flights.

Some future work could involve adding a spatial browsing capability [8, 11, 32] for the trajectories as well as incorporating our methods in distributed spatial environment [41].

Table 5: Top 10 features ranked by their relative importance.

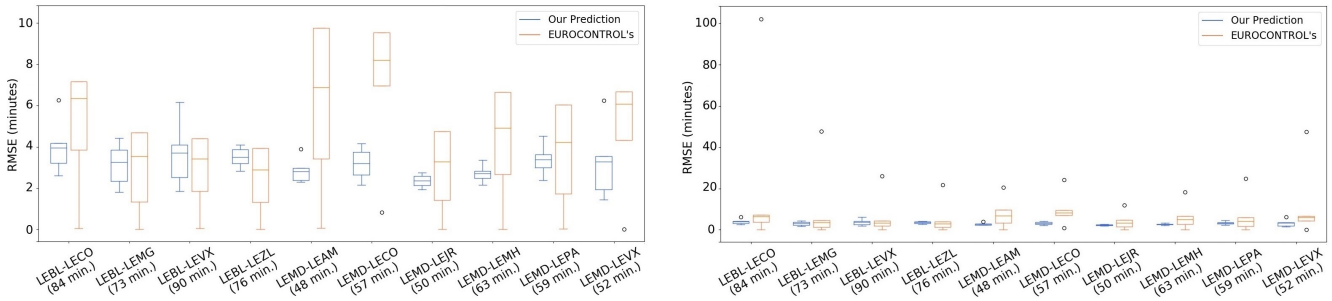
Rank	*Feature*	*Score*
1	Arrival airport	1.0
2	Atmospheric pressure	0.67854
3	Atmospheric wind speed	0.66231
4	Atmospheric wind direction	0.65224
5	Atmospheric humidity	0.63331
6	Atmospheric temperature	0.61314
7	Airport congestion rate	0.53212
8	Sector congestion rate	0.31153
9	Flight no	0.29192
10	Aircraft type	0.13221

8 ACKNOWLEDGEMENTS

This work was supported in part by the NSF under Grant and IIS-13-20791. We are grateful to the Boeing Research & Technology Europe, a member of the project DART [20] team for providing the data used in this study. The project DART has received funding from the SESAR Joint Undertaking under Grant Agreement No. 699299 under the European Union's Horizon 2020 Research and Innovation Programme.

REFERENCES

[1] Samet Ayhan, Paul Comitz, and Ron LaMarche. 2008. Implementing Geospatially Enabled Aviation Web Services. In *2008 Integrated Communications, Navigation and Surveillance Conference (ICNS)*. Herndon, VA, 1–8.
[2] Samet Ayhan, Brendan Fruin, Fan Yang, and Michael O. Ball. 2014. NormSTAD Flight Analysis: Visualizing Air Traffic Patterns over the United States. In *Proceedings of the 7th ACM SIGSPATIAL IWCTS*. Dallas/Fort Worth, TX, 1–10.
[3] Samet Ayhan, Johnathan Pesce, Paul Comitz, Gary Gerberick, and Steve Bliesner. 2012. Predictive Analytics with Surveillance Big Data. In *Proceedings of the 1st*



a) RMSE by our prediction vs. EUROCONTROL's prediction at a larger scale.

b) RMSE by our prediction vs. EUROCONTROL's prediction at a smaller scale.

Figure 5: RMSE by our prediction vs EUROCONTROL's on all flight routes.

ACM SIGSPATIAL Int'l Workshop on Analytics for Big Geospatial Data. Redondo Beach, CA, 81–90.

[4] Samet Ayhan and Hanan Samet. 2015. DICLERGE: Divide-Cluster-Merge Framework for Clustering Aircraft Trajectories. In *Proceedings of the 8th ACM SIGSPATIAL IWCTS*. Seattle, WA, 7–14.

[5] Samet Ayhan and Hanan Samet. 2016. Aircraft Trajectory Prediction Made Easy with Predictive Analytics. In *Proceedings of the 22nd ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining*. San Francisco, CA, 21–30.

[6] Samet Ayhan and Hanan Samet. 2016. Time Series Clustering of Weather Observations in Predicting Climb Phase of Aircraft Trajectories. In *Proceedings of the 9th ACM SIGSPATIAL IWCTS*. San Francisco, CA, 25–30.

[7] Xiaoli Bai, Lesley A. Weitz, and Stephanie Priess. 2016. Evaluating the Impact of Estimated Time of Arrival Accuracy on Interval Management Performance. In *AIAA GNC Conference and Exhibit*. San Diego, CA.

[8] Frantisek Brabec and Hanan Samet. 2007. Client-based spatial browsing on the world wide web. *IEEE Internet Computing* 11, 1 (January/February 2007), 52–59.

[9] Dingxiong Deng, Cyrus Shahabi, Ugur Demiryurek, Linhong Zhu, Rose Yu, and Yan Liu. 2016. Latent Space Model for Road Networks to Predict Time-Varying Traffic. In *Proceedings of the 22nd ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining*. San Francisco, CA.

[10] Harris Drucker. 1997. Improving Regressors Using Boosting Techniques. In *Proceedings of the 14th Int'l Conference on Machine Learning*. San Francisco, CA.

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

[12] Yoav Freund and Robert E. Schapire. 1997. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* 55, 1 (August 1997), 119–139.

[13] Jerome H. Friedman. 2001. Greedy Function Approximation: A Gradient Boosting Machine. *Annals of Statistics* 29, 5 (October 2001), 1189–1232.

[14] Jerome H. Friedman. 2002. Stochastic Gradient Boosting. *Comput. Stat. Data Anal.* 38, 4 (February 2002), 367–378.

[15] Yan Glina, Richard Jordan, and Mariya Ishutkina. 2012. A Tree-based Ensemble Method for Prediction and Uncertainty Quantification of Aircraft Landing Times. In *Proceedings of the 10th Conference on Artificial and Computational Intelligence and its Applications to the Environmental Sciences*. New Orleans, LA.

[16] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. 2001. *The Elements of Statistical Learning*. Springer New York Inc., New York.

[17] Haomiao Huang, Kaushik Roy, and Claire Tomlin. 2007. Probabilistic Estimation of State-Dependent Hybrid Mode Transitions for Aircraft Arrival Time Prediction. In *AIAA GNC Conference and Exhibit*. Hilton Head, SC.

[18] Edwin Jacox and Hanan Samet. 2008. Metric space similarity joins. *ACM Transactions on Database Systems* 33, 2 (June 2008), 7.

[19] Eamonn Keogh and Chotirat Ann Ratanamahatana. 2005. Exact Indexing of Dynamic Time Warping. *Knowl. Inf. Syst.* 7, 3 (March 2005), 358–386.

[20] Christian S. Kern, Ivo P. de Medeiros, and Takashi Yoneyama. 2015. Data-Driven Aircraft Estimated Time of Arrival Prediction. In *Proceedings of the 9th IEEE Int'l Systems Conference (SysCon)*. Vancouver, BC.

[21] Joel Klooster, Keith Wichman, and Okko Bleeker. 2008. Strategic Aircraft Trajectory Prediction Uncertainty and Statistical Sector Traffic Load Modeling. In *AIAA GNC Conference and Exhibit*. Honolulu, HI.

[22] Jimmy Krozel, Changkil Lee, and Joseph Mitchell. 1999. Estimating Time of Arrival in Heavy Weather Conditions. In *AIAA GNC Conference and Exhibit*. Portland, OR.

[23] Benjamin Letham, Lydia M. Letham, and Cynthia Rudin. 2016. Bayesian Inference of Arrival Rate and Substitution Behavior from Sales Transaction Data with Stockouts. In *Proceedings of the 22nd ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining*. San Francisco, CA.

[24] Benjamin Levy and David Rappaport. 2007. Arrival Time Estimation (ETA) from On-Final to Gate. In *AIAA ATIO Conference*. Belfast, Northern Ireland.

[25] Wei-Hua Lin and Jian Zeng. 1999. Experimental Study of Real-Time Bus Arrival Time Prediction with GPS Data. *Journal of the Transportation Research Board* 1666 (April 1999), 101–109.

[26] NOAA. 2016. *NCEP Global Forecast System*. https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/global-forcast-system-gfs

[27] Sarana Nutanong, Edwin Jacox, and Hanan Samet. 2011. An incremental Hausdorff distance calculation algorithm. *PVLDB* 4, 8 (August 2011), 506–517.

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

[29] Ioannis Parolas, Ron Van Duin, Ioanna Kourounioti, and Lorant A. Tavasszy. 2017. *Prediction of Vessels' ETA Using Machine Learning A Port of Rotterdam Case Study*. Technical Report. Transportation Research Board, Washington, DC.

[30] Shangfu Peng, Jagan Sankaranarayanan, and Hanan Samet. 2016. SPDO: High-Throughput Road Distance Computations on Spark Using Distance Oracles. In *Proceedings of the 32nd IEEE Int'l Conference on Data Engineering*. Helsinki, Finland, 1239–1250.

[31] Kaushik Roy, Benjamin Levy, and Claire Tomlin. 2006. Target Tracking and Estimated Time of Arrival (ETA) Prediction for Arrival Aircraft. In *AIAA GNC Conference and Exhibit*. Keystone, CO.

[32] Hanan Samet, Houman Alborzi, Frantisek Brabec, Claudio Esperança, Gisli R. Hjaltason, Frank Morgan, and Egemen Tanin. 2003. Use of the SAND spatial browser for digital government applications. *Commun. ACM* 46, 1 (January 2003), 63–66.

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

[34] Jagan Sankaranarayanan, Houman Alborzi, and Hanan Samet. 2006. Distance Join Queries on Spatial Networks. In *Proceedings of the 14th ACM Int'l Symposium on Advances in Geographic Information Systems*. Arlington, VA, 211–218.

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

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

[37] Jagan Sankaranarayanan and Hanan Samet. 2010. Roads belong in databases. *IEEE Data Engineering Bulletin* 33, 2 (June 2010), 4–11.

[38] Jagan Sankaranarayanan, Hanan Samet, and Houman Alborzi. 2009. Path oracles for spatial networks. *PVLDB* 2, 1 (August 2009), 1210–1221.

[39] SESAR. 2017. *Data-Driven Aircraft Trajectory Prediction Research*. http://dart-research.eu/the-project/

[40] Gabor Takacs. 2014. Predicting Flight Arrival Times with a Multistage Model. In *IEEE Int'l Conference on Big Data*. Keystone, CO.

[41] Egemen Tanin, Aaron Harwood, and Hanan Samet. 2005. A distributed quadtree index for peer-to-peer settings. In *Proc. of the 21st IEEE Int'l Conf. on Data Engineering*. Tokyo, Japan, 254–255.

[42] Paul T. Wang, Craig R. Wanke, and Frederick P. Wieland. 2004. Modeling Time and Space Metering of Flights in the National Airspace System. In *Proceedings of the Winter Simulation Conference*. Washington, DC.

[43] Jian Wei, Jooyoung Lee, and Inseok Hwang. 2015. Estimated Time of Arrival Prediction based on State-Dependent Transition Hybrid Estimation Algorithm. In *AIAA GNC Conference and Exhibit*. Kissimmee, FL.