https://ntrs.nasa.gov/search.jsp?R=20190027063 2020-03-10T16:39:30+00:00Z

Taxi Time Prediction at Charlotte Airport Using Fast-Time Simulation and Machine Learning Techniques

Hanbong Lee[1] and Waqar Malik[2]
University of California, Santa Cruz, NASA Ames Research Center, Moffett Field, CA 94035, USA

Bo Zhang[3] and Balaji Nagarajan[4]
American Airlines, Fort Worth, TX, 76155, USA

and

Yoon C. Jung[5]
NASA Ames Research Center, Moffett Field, CA 94035, USA

Accurate taxi time prediction is required for enabling efficient runway scheduling that can increase runway throughput and reduce taxi times and fuel consumptions on the airport surface. Currently NASA and American Airlines are jointly developing a decision-support tool called Spot and Runway Departure Advisor (SARDA) that assists airport ramp controllers to make gate pushback decisions and improve the overall efficiency of airport surface traffic. In this paper, we propose to use Linear Optimized Sequencing (LINOS), a discrete-event fast-time simulation tool, to predict taxi times and provide the estimates to the runway scheduler in real-time airport operations. To assess its prediction accuracy, we also introduce a data-driven analytical method using machine learning techniques. These two taxi time prediction methods are evaluated with actual taxi time data obtained from the SARDA human-in-the-loop (HITL) simulation for Charlotte Douglas International Airport (CLT) using various performance measurement metrics. Based on the taxi time prediction results, we also discuss how the prediction accuracy can be affected by the operational complexity at this airport and how we can improve the fast-time simulation model before implementing it with an airport scheduling algorithm in a real-time environment.

Nomenclature

ASDE-X	=	Airport Surface Detection Equipment, Model X
ATC	=	Air Traffic Control
ATCT	=	Airport Traffic Control Tower
CLT	=	airport code for Charlotte Douglas International Airport
DFW	=	airport code for Dallas/Fort Worth International Airport
DST	=	Decision-Support Tool
EDCT	=	Expected Departure Clearance Time
FAA	=	Federal Aviation Administration
FFC	=	FutureFlight Central
HITL	=	Human-In-The-Loop

[1] Assistant Research Scientist, UARC, Mail Stop 210-8, Moffett Field, CA 94035, hanbong.lee@nasa.gov.
[2] Associate Research Scientist, UARC, Mail Stop 210-8, Moffett Field, CA 94035, waqar.malik@nasa.gov.
[3] Senior Analyst, American Airlines Operations Planning and Performance Department, 4333 Amon Carter Blvd, Mail Drop 5423, Fort Worth, TX 76155, bo.zhang@aa.com.
[4] Manager, American Airlines Operations Planning and Performance Department, 4333 Amon Carter Blvd, Mail Drop 5423, Fort Worth, TX 76155, balaji.nagarajan@aa.com.
[5] Aerospace Engineer, NASA Ames Research Center, Mail Stop 210-6, Moffett Field, CA 94035, yoon.c.jung@nasa.gov, AIAA Senior Member.

kNN	=	*k*-Nearest Neighbors
LINOS	=	Linear Optimized Sequencing
LR	=	Linear Regression
MAE	=	Mean-Absolute Error
MIT	=	Miles-In-Trail
NASA	=	National Aeronautics and Space Administration
RF	=	Random Forest
RMSE	=	Root-Mean-Square-Error
SARDA	=	Spot and Runway Departure Advisor
SOSS	=	Surface Operations Simulator and Scheduler
SVM	=	Support Vector Machines
TMI	=	Traffic Management Initiatives

I. Introduction

WITH the growth of air traffic demand, it becomes more important to improve the efficiency in airport operations and reduce the congestion on the airport surface. Increasing the predictability of taxi times and wheels-off times is one of the key factors to achieve more efficient airport operations. Accurate taxi time estimation can be used for determining appropriate holding times for departures at their gates, solving the aircraft taxi scheduling problem, and assigning the best gates to turnaround flights. Furthermore, predicting precise wheels-off times for departures is important for optimizing the departure runway sequence and schedule at the origin airport, as well as for improving the arrival time predictions at the destination airports.

There have been several attempts to predict the taxi-out times for departures at airports. Initially, Idris et al. used a queuing model for taxi-out time estimation at Boston Logan airport.[1] Following the research, a few predictive queuing models were developed to estimate the taxi-out times from gates to the departure runways, and improved by considering the effect of taxiway interactions.[2] These models could be used to mitigate the congestion level on the ground by controlling the number of departures pushing back from the gates. Simple linear equations derived from regression analysis using the Airport Surface Detection Equipment, Model X (ASDE-X) surface surveillance data were also used to predict departure taxi times at several major airports in the United States.[3,4]

Taxi-out time prediction may be considered as a special case of travel time prediction in the field of automobile transportation systems. Research in road travel time prediction is abundant and predates the research in aircraft taxi time prediction. Various road travel time prediction methods using machine learning techniques, including linear regression models, neural networks, regression trees, *k*-nearest neighbors, locally-weighted regression, and support vector regression, were suggested and compared with real time road traffic data on the highway in different cities.[5-9]

The machine learning methods used in predicting the road travel times can be applied to the aircraft taxi time prediction problem. Several statistical approaches have been introduced to predict aircraft taxi times on the airport surface and take-off times of departures. A few linear regression models were used for modeling aircraft taxi times at Dallas/Fort Worth International Airport (DFW) with several independent variables such as taxi distance, airlines, gate group, the number of departures and arrivals.[10,11]A neural network model was also introduced to predict wheels-off times at DFW.[12] In addition, reinforcement learning algorithms were developed and investigated to check the accuracy of taxi-out time prediction at several major airports in the United States.[13-15] Recently, various regression methods, including multiple linear regression, least median squared linear regression, support vector regression, model trees, and fuzzy rule-based systems, were also applied to European airports for taxi time prediction problems.[16,17]

In addition to the data-driven approaches above, air traffic simulation tools can be used for validation and performance evaluation of taxi time prediction functions, provided they represent the surface traffic flow at a target airport realistically. Various simulation tools have been developed for modeling and analyzing airport operations in the past decades.[18] Using these simulation tools, we can test various traffic scenarios at airports and airspace, and forecast the upcoming traffic status quickly before conducting field tests in real operating environments that would require significant investments in human and equipment resources. Microscopic simulation models can reflect individual aircraft movements, consider possible conflicts with other aircraft, and deal with tactical issues in runway and taxiway operations. Most microscopic models are based on a discrete-event simulation approach, where system states change only at the moments when certain events occur. In these models, the airfield and airspace can be represented as a network of nodes and links, on which aircraft in the simulations follow the prescribed paths.

These simulation tools have been enhanced to simulate the movement of individual aircraft with acceptable level of accuracy. These models can also simulate stochastic processes using random variables to reflect the uncertainty in

airport operations. Given flight schedules and gate/runway information, they can calculate the travel times of taxiing aircraft and estimate the take-off times of departures while considering the interactions of aircraft moving on the ground. For example, the Surface Operations Simulator and Scheduler (SOSS) is an airport surface fast-time simulation tool developed by NASA Ames Research Center. SOSS uses the kinematic models of different types of aircraft and node-link networks to simulate surface traffic. This tool was used to analyze the impacts of scheduling algorithms on the surface traffic at DFW and CLT.[19,20] The SOSS model was also applied to estimate the wheels-off times at DFW, but it could not provide better estimation than a data-driven method in which average taxi times from actual data were added to the gate-out times for predictions.[21] Another example is LINear Optimized Sequencing (LINOS), which is a discrete event-based fast-time simulation tool developed by US Airways (now American Airlines).[22] LINOS can predict taxi times and surface congestion to increase awareness of the ramp controller by continuously evaluating the airport state. This simulation tool can compute taxi-out times and expected take-off times for the given airport state for all departing (at gate or already taxiing) flights, as well as taxi-in times and gate-in times for arrival flights.

NASA Ames Research Center developed and tested a decision-support tool (DST) called Spot and Runway Departure Advisor (SARDA) to assist air traffic control tower personnel and airline ramp operators in managing and controlling airport surface operations.[23-26] Multiple high fidelity human-in-the-loop (HITL) simulations were conducted to evaluate the performance of the runway scheduler and the usability of the tool. In recent years, NASA and American Airlines have been working together to implement LINOS in SARDA to provide predictions of the take-off times over the departure runways in a real-time operational environment. We expect that LINOS can provide reasonable predictions of take-off times to the SARDA runway scheduler by estimating the taxi times to runways for individual departure flights. Recently, NASA conducted a HITL simulation with active American Airlines' ramp controllers to test the SARDA runway scheduler's feasibility for the surface operations at Charlotte Douglas International Airport (CLT).[26] In the HITL simulation, the SARDA runway scheduler used a simple Dead Reckoning method (called DR hereinafter) based on unimpeded taxi conditions when calculating the predicted take-off times. By using LINOS, the SARDA scheduler can possibly receive more accurate predictions, and therefore, provide more efficient departure sequencing for controllers and airlines.

This paper evaluates the taxi time prediction performance of LINOS using the HITL simulation data for CLT airport before applying it to real-time airport operations. In order to assess the accuracy of LINOS prediction on the take-off times for departing aircraft, we compare the results with those from machine learning techniques, which are data-driven prediction methods based on historical traffic data. By understanding the performance level of the current LINOS simulation in comparison with those machine learning techniques, we can define a target prediction accuracy of LINOS in a real-time operational environment where taxi time predictions can be updated frequently as airport status changes dynamically. We will also evaluate the prediction performance of LINOS and machine learning methods by comparing it with the results from the existing Dead Reckoning method.

This paper is organized as follows. In Section II, we will present the simulation set-up and traffic data used in this study. We then describe the taxi time prediction methods using LINOS and machine learning techniques in the following section. In Section IV, we will show the prediction results of the taxi-out times for departures. Lastly, the conclusion and future work will be described in Section V.

II. Traffic data from the human-in-the-loop simulation for Charlotte airport

This section describes the airport layout and operational environment of Charlotte airport, and the human-in-the-loop (HITL) simulation for this airport is briefly introduced since the traffic data from the HITL simulation is used for the taxi time prediction study in this paper.

Figure 1 shows the airport layout of Charlotte Douglas International Airport (CLT), which is one of the major hub airports of American Airlines. This airport has four runways and one main terminal building between Runway 18C/36C and 18L/36R. In this paper, we focus on South-flow traffic, which uses the runway configuration (18R, 23 | 18C, 18L). Since most flights at this airport are operated by American Airlines, the ramp area is controlled by this airline. The limited area for aircraft taxiing and the presence of alleyways and single lanes in the ramp area cause frequent interaction of arrivals and departures while moving on the ground. Especially, this airport has a two-way single lane in the ramp next to Taxiway C, where arrivals going to Terminal E (Northeastern concourse) and Westbound departures from the same terminal will pass by each other in the opposite direction. This single lane is a major bottleneck that complicates surface traffic control at this airport.

The traffic data used in this study comes from NASA's HITL simulation for CLT, in which four ramp controllers, three tower controllers, and nine pseudo pilots participated to test a new departure pushback decision-support tool for ramp controllers (called *SARDA-CLT* in Ref. [26]). The simulation study took place at the FutureFlight Central (FFC), the airport tower simulator facility at the NASA Ames Research Center that provides a 360-degree out-the-window view, for three weeks in September through October, 2014. In the HITL simulation, we assumed South-flow configuration only (Runways 18L and 18C dedicated to departures and Runways 23 and 18R dedicated to arrivals) and good weather conditions. Similar to the real operations, we also considered Traffic Management Initiatives (TMIs) in effect for this airport, including 15 nmi separations over the MERIL departure fix and Expect Departure Clearance Time (EDCT) restrictions for selected flights.

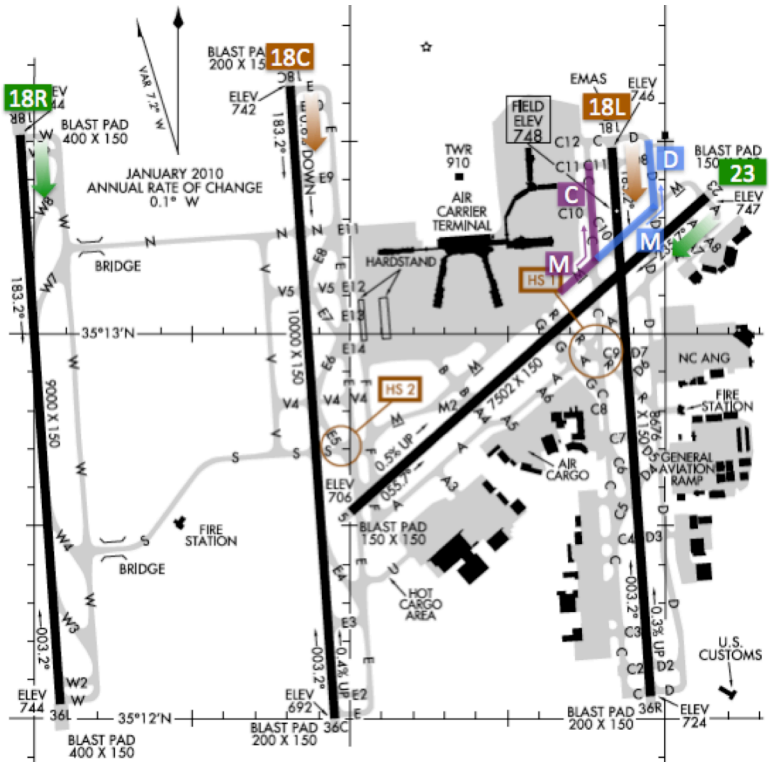

Figure 1. Airport layout at CLT.

Two one-hour scenarios modified from the actually recorded traffic data at CLT on May 16th, 2013 were tested in the HITL simulation. The scenarios consisted of a departure push followed by an arrival push, and were slightly compressed in time to make the traffic more congested with some overlap between departures and arrivals. Scenario 1 had 96 departures and 80 arrivals, whereas Scenario 2 had 84 departures and 72 arrivals. Scenario data files included flight call signs, destination airports, aircraft models, initially assigned gate, spot, runway and fix information, scheduled gate-out times and flight ready times for departures, and estimated landing times for arrivals. The data also contained some entries for aircraft specific variations in pushback time, engine spool-up time, and taxiing speed, which guaranteed repeatability across different runs while allowing uncertainties in surface operations during the HITL simulation.

For each scenario, two types of simulation runs were performed: "Advisory" and "Baseline" runs. In the Baseline runs, ramp controllers were asked to meter the departures at the gates as they would normally do in the current operations at CLT. In these runs, any pushback or holding advisories for individual flights were not provided to the ramp controllers, except for the number of taxiing departures per runway. In the Advisory runs, on the other hand, a gate pushback advisory for each flight was shown on the Ramp Traffic Console display to assist the ramp controllers in metering the departure traffic by specifying whether the flight should be released or held for certain duration. The pushback advisories were provided by the SARDA runway scheduler, a decision-support tool developed by NASA to help controllers mitigate ground congestion and increase runway throughput. To get the estimated take-off times of departing aircraft, the SARDA scheduler used a simple Dead Reckoning algorithm, in which the taxi-out time of a departure was assumed to be the unimpeded taxi time based on its taxi distance and nominal taxi speed.

The HITL simulation was performed for three weeks, repeating the same scenarios with and without pushback advisories. In each week, sixteen runs were conducted, resulting in a total of 48 runs. To reduce learning effects and fatigue bias, the run order in each week was counterbalanced for each of the two scenarios and two advisory modes. For South and East ramp sectors, in addition, a different pair of current CLT ramp controllers participated in the simulation every week, while changing their positions between the ramp sectors during the week. More details of the HITL simulation for CLT are described in Ref. [26].

III. Technical Approaches to Predicting Taxi Times at CLT

This section introduces two different taxi time prediction approaches, the fast-time simulation using LINOS and the data-driven analytical methods using machine learning techniques, and describes how to apply these approaches to the traffic scenarios at CLT.

A. LINOS

LINOS is a discrete-event simulation model that simulates the movement of aircraft between gates and runways along the taxi paths on the ramp area and taxiways. In a real-time environment, it can run continuously with current aircraft status as input, including the latest information on gate departure times for departures, estimated landing times for arrivals, and current positions on the ground for taxiing aircraft, and update its predictions whenever events occur. In LINOS, the taxi paths on the airport surface are built as a node-link network, where aircraft travel from node to node. This node-link network is a dynamic network based on aircraft capability and adapts to airport configuration and operational conditions. LINOS can resolve ground congestion and possible taxi and gate conflicts using its own business rules engine.

LINOS can simulate the current airport state, including flights currently taxiing and flights expected to arrive or depart in the near future, and run the simulation multiple times to account for the stochastic processes in airport operations. During each simulation run, LINOS calculates each flight's simulated taxi time, measures ground congestion (aircraft density based on area), and records taxi and gate conflicts. Each run can have a different outcome due to the randomly generated parameters within the given range to consider the uncertainties of airport operations. Through repeated simulations, this fast-time simulation tool can provide users with the expected taxi time for each aircraft and its variance, take-off sequence, spot and runway queue entry times, runway throughput, congestion status and possible conflicts over time.

In this study, we used LINOS to calculate the predicted taxi times and estimate the take-off times for departures. A node-link network model representing the CLT airport configuration was first created. Then, the operational parameters such as separation time requirements over runways, pushback time distributions depending on aircraft types and terminals, and nominal taxi speeds were set in the model. To make the model similar to the HITL simulation and match the real operations at CLT, several airport specific operational rules were also integrated in LINOS. These included the Arrival Departure Window (ADW) rule for non-intersecting converging-runway operations (CRO), the use of Mike-Charlie (M-C) taxiway bypass routes (the thick solid purple lines in Fig. 1) for selective arrivals to avoid head-on conflicts with departures on the single lane in the ramp, the use of Mike-Delta (M-D) taxi routes (the thick solid blue lines in Fig. 1) for the departures flying over the MERIL departure fix to cross runway 18L on the ground and enter the runway queue on the other side of 18L for increasing runway throughput, and the spacing between aircraft waiting in departure runway queues. The parameters used in this simulation model were calibrated using the Week 1 and 2 simulation data (32 runs) gathered in the first two weeks of the HITL simulation. After the calibration processes, LINOS demonstrated good consistency with the HITL simulation results in taxi-out times for departures, as well as in taxi-in times of arrivals.

To evaluate the taxi time prediction performance of LINOS, we used the remaining Week 3 simulation data collected in the third week of the HITL simulation. For each one-hour long traffic scenario, flight information for every flight in the HITL simulation, including call sign, tail number, aircraft model, gate, spot and runway usage, fix information, and actual pushback or landing time, was provided as input data to LINOS. Considering the various uncertainties of airport operations, the simulations were repeated ten times. The average taxi time from the ten simulation runs was used to calculate the predicted taxi time for each flight. Although LINOS can be run in a real-time operational environment and update the airport state every 10 seconds, we limited its run with static data from the HITL simulation to compare it with the machine learning methods based on deterministic data.

B. Machine Learning Techniques

Since the machine learning algorithms are data-driven approaches, it is important to determine what features should be selected from the considerable traffic data. Among the various characteristics of airport operations, we considered the following features as variables that could have an impact on taxi processes and the taxi time prediction: Gate, Spot, Runway, Aircraft model, Taxi distance from gate to runway, Number of departures on the surface by runway (8 features for Runway 5/23, 18L/36R, 18C/36C, and 18R/36L), and Number of arrivals on the surface by terminal (5 features for Terminal A, B, C, D, and E).

The taxi distance from gate to runway is a dominant factor that determines the unimpeded taxi-out time, but the other information like assigned gate, spot and runway is also significant. The duration that a departing aircraft spends to complete its pushback process can be different at each gate. The wait times to get a taxi clearance to enter

the movement area controlled by FAA Airport Traffic Control Tower (ATCT) and a take-off clearance from the tower may be dependent upon spot and runway, respectively. The aircraft model determines the physical performance and limitations in taxi operations on the ground. The number of departures and arrivals are also included in the variable list to account for the congestion level on the ground and the taxi delay added to the unimpeded taxi time. For more accurate predictions, the numbers of departures moving on the surface are categorized by the assigned runway, while the numbers of arrivals traveling on the taxiways are divided by the terminal in which the assigned gates are located.

In this study, we tried to use four different machine learning methods:

1) Linear Regression (LR),
2) Support Vector Machines (SVM),
3) *k*-Nearest Neighbors (*k*NN), and
4) Random Forest (RF).

Linear Regression (LR) is an approach for modeling the relationship between a scalar dependent variable y and one or more explanatory variables denoted X. Data are modeled using linear predictor functions, and unknown model parameters are estimated from the data. For prediction, it can be used to fit a predictive model to an observed data set of y and X values. After developing such a model, if an additional value of X is given without its accompanying value of y, the fitted model can be used to make a prediction of the value of y.

Support Vector Machines (SVM) are supervised learning models with associated learning algorithms that analyze data and recognize patterns. Given a set of training examples, each marked as belonging to one of two categories, a SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier.

k-Nearest Neighbors (*k*NN) algorithm is a non-parametric method used for classification and regression. The input consists of the *k*-th closest training examples in the feature space. In *k*NN regression, the output is the property value for the object. This value is the average of the values of its *k* nearest neighbors.

Random Forest (RF) is an ensemble learning method for classification and regression that operates by constructing a multitude of decision trees at training time and computing the class that is the mode of the classes output by individual trees.

These four machine learning algorithms were modeled in Python and run using scikit-learn library.[27] While the first 2-week datasets (Week 1 and 2) from the HITL simulation were used to train the machine learning algorithms, the remaining Week 3 datasets were used as test datasets for prediction performance evaluation. In this way, we compared the prediction performance between the machine learning methods and LINOS with respect to the same test datasets. The parameters used in each machine learning algorithm such as the number of neighbors, *k* value, in the *k*NN method and the number of estimators in the RF method were adjusted to obtain the best prediction performance while calibrating the prediction algorithm with the training datasets.

Since machine learning approaches are based on deterministic data, they cannot be used in the dynamic environments like real-time airport operations. For instance, controllers sometimes change the spot or runway assigned to a departure, if they are necessary, while managing the traffic on the surface. It is also difficult for machine learning methods to update the prediction of take-off times for the departures that are out of the gates and actively taxiing. Therefore, we will only use these prediction methods to assess the taxi time prediction performance of the fast-time simulation-based approach using LINOS.

IV. Prediction Results

The following section presents the taxi time prediction results from the aforementioned technical approaches and their performance accuracy. We have applied the proposed taxi time prediction methods to the realistic traffic scenarios used in the HITL simulation for CLT. Each scenario has two datasets: 1) Baseline runs similar to the current operations except for using virtual strips on the touchscreen-based Ramp Traffic Console instead of paper strips,[26] and 2) Advisory runs following pushback advisories from the SARDA runway scheduler. We also compare the prediction accuracy between Baseline and Advisory runs in this section.

A. Taxi-out Time Prediction Results

First, we compared the predicted taxi-out times of departures from four different machine learning approaches and a fast-time simulation using LINOS in terms of several performance metrics. Table 1 shows mean taxi time difference, Root-Mean-Square-Error (RMSE) and Mean-Absolute Error (MAE) values in minutes for each prediction method. The mean taxi time difference is defined by the average value of actual taxi times minus the average value of predicted taxi times. RMSE and MAE values are calculated by the following equations:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(T_i - T_i^*\right)^2} \tag{1}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|T_i - T_i^*\right| \tag{2}$$

where T_i is the actual taxi time value of aircraft *i*, T_i^* is the predicted taxi time value of aircraft *i*, and *n* is the total number of departing aircraft.

Table 1 Comparison of performance metrics in minutes for Baseline and Advisory runs

Test dataset	Performance metrics	LR	SVM	*k*NN	RF	LINOS	DR
Baseline runs	Mean Taxi Time Difference	-0.31	0.22	-0.28	-0.23	-0.36	6.56
	Root Mean-Squared Error	2.83	2.29	1.22	1.15	2.67	7.14
	Mean-Absolute Error	2.07	1.58	0.90	0.85	1.91	6.56
Advisory runs	Mean Taxi Time Difference	-0.15	0.16	-0.16	-0.11	-0.04	5.71
	Root Mean-Squared Error	2.62	2.16	1.12	1.21	2.06	6.25
	Mean-Absolute Error	1.87	1.37	0.87	0.92	1.24	5.71

For all the proposed prediction approaches, Baseline and Advisory runs showed the mean taxi-out time differences between actual and predicted taxi times were below 0.36 minutes (22 seconds) and 0.16 minutes (10 seconds), respectively. This result means that the calibration of the parameters used in LINOS was good enough to have a similar accuracy to the machine learning algorithms trained with the training datasets. Among the four different machine learning methods, the *k*-Nearest Neighbors (*k*NN) and Random Forest (RF) methods showed the best performance on both RMSE and MAE metrics. The LINOS simulation demonstrated better performance than the Linear Regression (LR) algorithm and the similar level of prediction accuracy to the Support Vector Machines (SVM) method's results.

In the last column in Table 1, we also included the performance metrics of the Dead Reckoning (DR) method as a reference. The DR method computes the unimpeded taxi-out times based on the taxi distance from gate to runway along the taxi route with the default taxi speed of 8.75 knots. This approach has been used in the current SARDA scheduler as a default method to estimate the arrival times of departures at runway. Because the DR method does not account for the congestion on the surface and the wait times in departure runway queues, the predicted mean taxi time from this method is much shorter than the actual mean taxi time, as can be seen in Table 1. Similarly, the RMSE and MAE metrics in the DR method also have greater values, compared with the proposed taxi time prediction methods, in both Baseline and Advisory runs.

Boxplots in Fig. 2 illustrate the taxi-out time difference distributions between actual taxi times and predicted taxi times from Baseline and Advisory runs for each traffic scenario, for each prediction method. According to these boxplots, the *k*-Nearest Neighbors (*k*NN) and Random Forest (RF) methods show the least deviations from the actual taxi times, without any outliers beyond a 5-minute gap, among the four different machine learning approaches used in this study for both Baseline and Advisory runs. LINOS also shows good taxi time prediction performance. In the LINOS cases, it is noted that we can obtain a better predictability when following gate pushback advisories (in the Advisory runs), compared to the Baseline runs that have no advisories.

The cumulative curves in Fig. 3 show the departures along with the absolute taxi time prediction error, depending on the prediction methods. In the Baseline runs, about 90% of the departures are below the absolute 4-minute prediction error for all the prediction methods, except for the Linear Regression (LR) method (about 5 minutes). The Advisory runs showed better prediction accuracy, with 97% departures within the same prediction error of +/– 4 minutes.

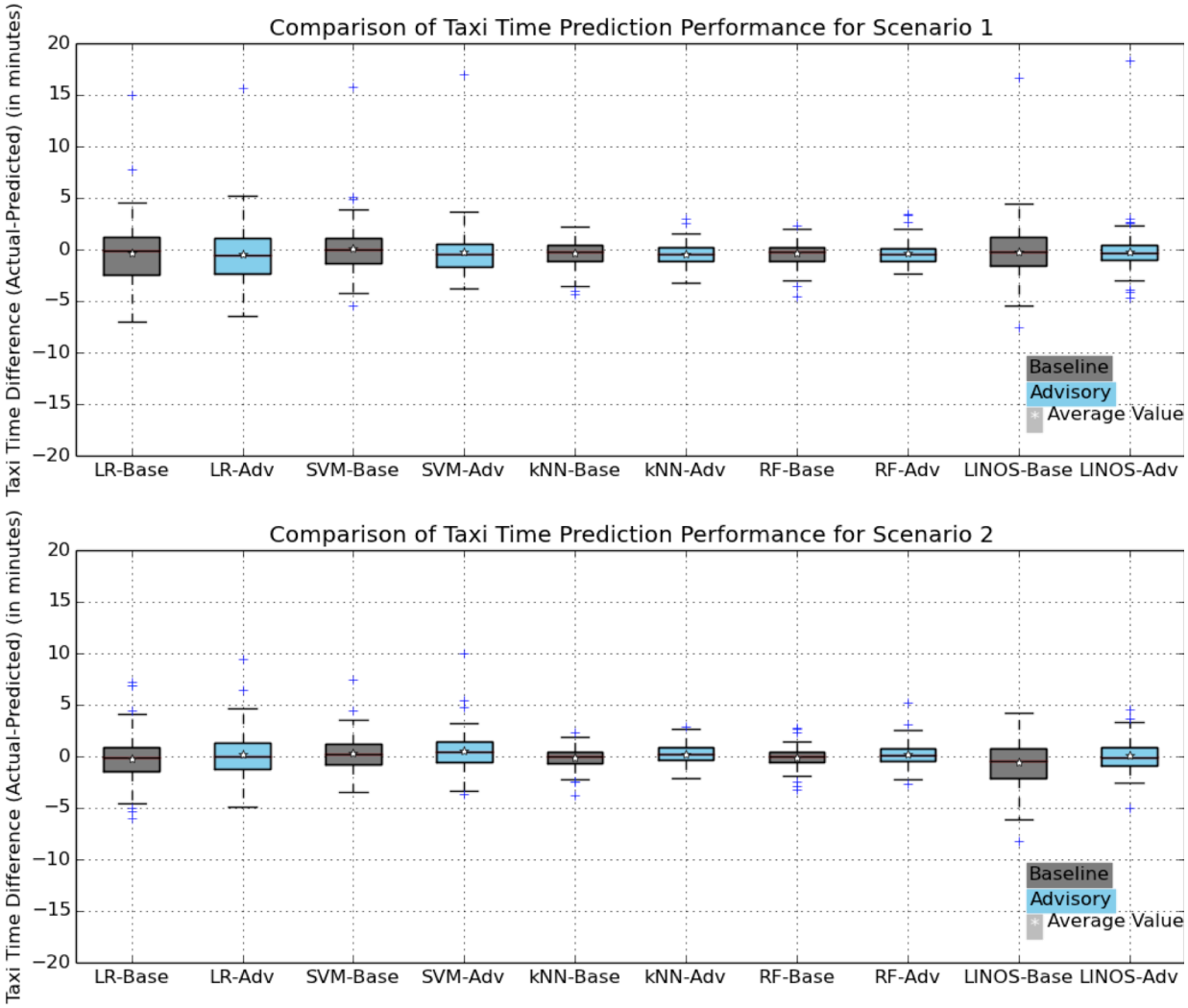

Figure 2. Comparison of Distributions on the Differences between Actual and Predicted Taxi Times for Scenario 1 (upper) and Scenario 2 (lower).

B. Discussion

There are several reasons why the actual taxi-out times from the HITL simulation are different from the predicted taxi times with some variations. Controllers in real operations and in the HITL simulation can change the taxi routes, if available, to mitigate the ground congestion and avoid potential conflicts on taxiways. At CLT, the ramp controllers sometimes direct arrivals to follow the M-C taxiway route outside the ramp area instead of the single lane inside the ramp area so as to prevent flights from having head-on conflicts with departures on the dynamic single direction taxiway. The use of M-C taxiway route can reduce the congestion in the ramp area, leading to shorter taxi times for both departures and arrivals. Ramp controllers also hold some flights in the hardstands or the designated holding areas in the ramp until the traffic congestion is resolved. In real operations, some arrivals entering the ramp area through Spot 4 or 5 are frequently sent to the west hardstand and held for a while during peak periods. Furthermore, ramp controllers need to consider the TMI constraints such as Expect Departure Clearance Time (EDCT) for selected flights and miles-in-trail (MIT) restrictions over a departure fix. These constraints can make it difficult to predict accurate taxi times of the flights affected by the restrictions because those flights are controlled differently from normal flights in order to meet the requirements, including long gate-holding, intersection take-offs, and detoured taxi routes. In fact, the prediction methods proposed in this paper showed large deviations in the taxi time prediction for a couple of flights having EDCT restrictions.

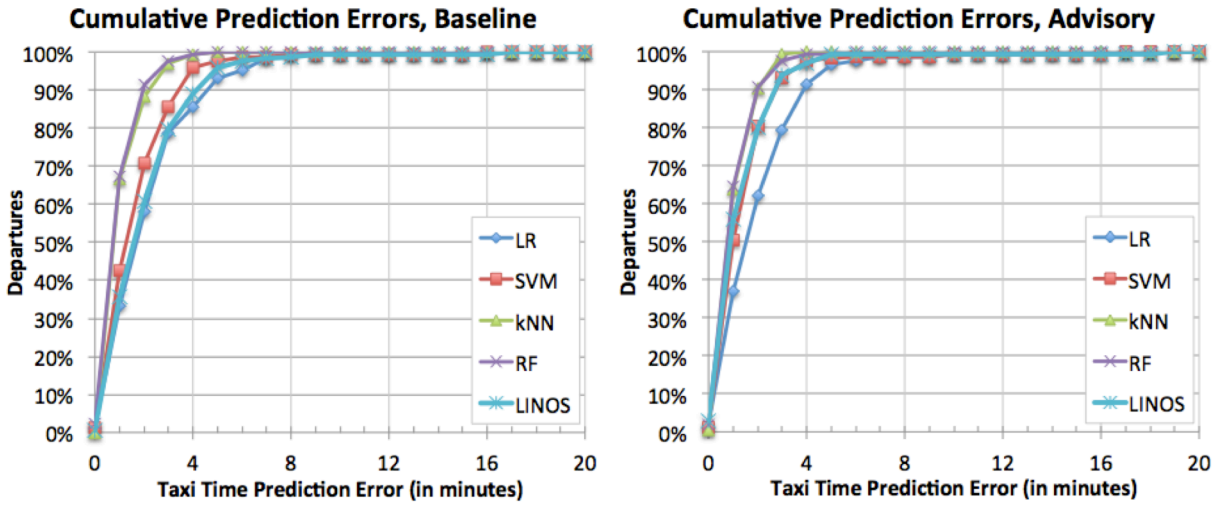

Figure 3. Cumulative absolute value of the taxi time prediction errors for Baseline (left) and Advisory (right) runs.

Tower controllers sometimes allow departures to take the intersection take-offs for increasing runway throughput, when possible. The current version of LINOS, however, does not account for these bypass take-offs. Tower controllers also send some outbound aircraft taxiing toward runway 18L to the M-D taxiway route by having them cross the runway and form another departure queue on the other side of 18L in order to increase runway throughput. These tactical operations by ramp and tower controllers make it difficult to predict the accurate taxi times.

According to the taxi time prediction results, some of machine learning methods showed better prediction performance than the fast-time simulation using LINOS. These results stem from the fact that both training and test datasets were obtained from the repetitive simulation runs using the same traffic scenarios under stable test conditions. To use the machine learning approaches in real-world operational environments, a significant amount of historical traffic data is required with acceptable level of reliability. In addition to the existing features mentioned in this paper, we may also need to consider weather/visibility conditions, time of day, day of week, and runway configuration as additional variables. Furthermore, machine learning methods can predict the take-off times only for the departures staying at their gates in a real-time environment. Once the flight leaves its gate, they cannot predict the taxi time accurately due to the limitation of data availability. LINOS, on the contrary, can be applied to real operations without any significant modifications or additional data input. Also, it can update the taxi time predictions continuously as the airport status changes dynamically. Besides predicting take-off times, LINOS can also provide ad-hoc estimations such as taxi times to spot and departure queue.

The prediction results also showed that the Advisory runs show better predictability, compared to the Baseline runs. In the Advisory runs, the departures are held at their gates, instead of waiting in the runway queues. This mitigates the traffic congestion in the ramp and assists tower controllers in planning the take-off sequence by reducing the uncertainties in the runway sequencing and scheduling operations, which enables more accurate taxi time prediction.

V. Conclusion

NASA and American Airlines have jointly worked to implement LINOS, a fast-time air traffic simulation tool, in the real-time operational environment and tested it with the human-in-the-loop (HITL) simulation data for the airport operations at Charlotte airport (CLT). It is expected that LINOS can provide more accurate prediction of take-off times for departures than a Dead Reckoning method used by the current SARDA runway scheduler. In order to assess the prediction accuracy of LINOS for taxi-out times, we also developed several machine learning algorithms based on historical traffic data and compared the predicted taxi times from both simulation-based and data-driven approaches with the actual ones from the HITL simulation.

The prediction results showed that LINOS could predict the taxi-out times as accurate as the Support Vector Machines method and better than the Linear Regression method and the Dead Reckoning method based on unimpeded taxi times. From the results, we discussed various factors of surface operations showing the operational

complexity at this airport, including alternative taxi routes, hardstands, bypass take-offs, and TMI constraints. These factors could affect the taxi time prediction performance.

Although we already integrated many aspects of real-world operational environments at CLT and calibrated the parameters used in LINOS, there are more opportunities to improve the taxi time prediction accuracy of LINOS. For example, we will add capability to handle EDCT flights and enable intersection take-offs. We believe that LINOS will be able to show better prediction performance in real-time operations when performing field tests at the airport because, as soon as it receives the surveillance data, it can precisely reflect the aircraft status on the surface and update the predictions accordingly with the latest information. In the future, we will expand the same taxi time prediction approaches to other runway configurations at CLT like North-flow traffic, and to the other airports having different airfield layouts and operational conditions.

References

[1]Idris, H. R., Clarke, J. P., Bhuva, R., and Kang, L., "Queuing Model for Taxi-Out Time Estimation," *Air Traffic Control Quarterly*, Vol. 10, No. 1, 2002, pp. 1-22.

[2]Simaiakis, I., and Balakrishnan, H., "Queuing Models of Airport Departure Processes for Emissions Reduction," *AIAA Guidance, Navigation, and Control (GNC) Conference*, Chicago, IL, August 2009.

[3]Legge, J., and Levy, B., "Departure Taxi Time Predictions Using ASDE-X Surveillance Data," *the 26th Congress of International Council of the Aeronautical Sciences (ICAS)*, Anchorage, AK, September 14-19, 2008.

[4]Srivastava, A., "Improving Departure Taxi Time Predictions Using ASDE-X Surveillance Data," *in Proceedings of the IEEE/AIAA 30th Digital Avionics Systems Conference (DASC)*, Seattle, WA, October 16-20, 2011.

[5]Handley, S., Langley, P., and Rauscher, F. A., "Learning to Predict the Duration of an Automobile Trip," *in Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining*, New York, 1998.

[6]Dharia, A., and Adeli, H., "Neural Network Model for Rapid Forecasting of Freeway Link Travel Time," *Engineering Applications of Artificial Intelligence*, Vol. 16, Issues 7-8, 2003, pp. 607-613.

[7]Wu, C., Ho, J., and Lee, D. T., "Travel-Time Prediction With Support Vector Regression," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 5, No. 4, December 2004.

[8]Nikovski, D., Nishiuma, N., Goto, Y., and Kumazawa, H., "Univariate Short-Term Prediction of Road Travel Times," *in Proceedings of the 8th International IEEE Conference on Intelligent Transportation Systems*, Vienna, Austria, September 13-16, 2005.

[9]Vanajakshi, L., and Rilett, L. R., "Support Vector Machine Technique for the Short Term Prediction of Travel Time," *in Proceedings of the 2007 IEEE Intelligent Vehicles Symposium*, Istanbul, Turkey, June 13-15, 2007.

[10]Jordan, R., Ishutkina, M., and Reynolds, T., "A Statistical Learning Approach to the Modeling of Aircraft Taxi Time," *in Proceedings of the IEEE/AIAA 29th Digital Avionics Systems Conference (DASC)*, Salt Lake City, UT, October 3-7, 2010.

[11]Kistler, M. S., and Gupta, G., "Relationship between Airport Efficiency and Surface Traffic," *the 9th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Hilton Head, SC, 2009.

[12]Chatterji, G. B., and Zheng, Y., "Wheels-Off Time Prediction Using Surface Traffic Metrics," *the 12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Indianapolis, IN, September 17-19, 2012.

[13]Balakrishna P., Ganesan, R., Sherry, L., and Levy, B., "Estimating Taxi-Out Times with a Reinforcement Learning Algorithm," *in Proceedings of the 27th Digital Avionics Systems Conference (DASC)*, St. Paul, MN, October 26-30, 2008.

[14]Balakrishna, P., Ganesan, R., and Sherry, L., "Application of Reinforcement Learning Algorithms for Predicting Taxi-Out Times," *the 8th USA/Europe Air Traffic Management (ATM) Research and Development (R&D) Seminars*, Napa, CA, 2009.

[15]Balakrishna, P., Ganesan, R., and Sherry, L., "Accuracy of Reinforcement Learning Algorithms for Predicting Aircraft Taxi-Out Times: a Case-Study of Tampa Bay Departures," *Transportation Research Part C: Emerging Technologies*, Vol. 18 No. 6, 2010, pp. 950-962.

[16]Ravizza, S., Atkin, J. A. D., Maathuis, M. H., and Burke, E. K., "A Combined Statistical Approach and Ground Movement Model for Improving Taxi Time Estimations at Airports," *Journal of the Operational Research Society*, Vol. 64, 2013, pp. 1347-1360.

[17]Ravizza, S., Chen, J., Atkin, J. A. D., Stewart, P., and Burke, E. K., "Aircraft Taxi Time Prediction: Comparisons and Insights," *Applied Soft Computing*, Vol. 14, 2014, pp. 397-406.

[18]Odoni, A. R., Bowman, J., Delahaye, D., Deyst, J. J., Feron, E., Hansman, R. J., Khan, K., Kuchar, J. K., Pujet, N., and Simpson, R. W., "Existing and Required Modeling Capabilities for Evaluating ATM System and Concepts," Technical report, International Center for Air Transportation, Massachusetts Institute of Technology, Cambridge, MA, March 1997.

[19]Windhorst, R. D., "Towards a Fast-Time Simulation Analysis of Benefits of the Spot and Runway Departure Advisor," *AIAA Guidance, Navigation, and Control (GNC) Conference*, Minneapolis, MN, August 13-16, 2012.

[20]Montoya, J., Windhorst, R., Zhu, Z., Gridnev, S., Griffin, K. J., Saraf, A., and Stroiney, S., "Analysis of Airport Surface Schedulers Using Fast-Time Simulation," *the 13th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Los Angeles, CA, August 12-14, 2013.

[21]Chatterji, G. B., and Zheng, Y., "Wheels-Off Time Estimation at Non-ASDE-X Equipped Airports," *the 13th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Los Angeles, CA, August 12-14, 2013.

[22]Ergan, H., and Zhang, B., "US Airways' Road to Precise Departure Sequencing," *AGIFORS Airline Operations 2012*, Atlanta, GA, August 2012.

[23]Malik, W. A., Gupta, G., and Jung, Y., "Managing Departure Aircraft Release for Efficient Airport Surface Operations," *AIAA Guidance, Navigation, and Control (GNC) Conference*, Toronto, Canada, August 2010.

[24]Malik, W., Gupta, G., and Jung, Y., "Spot Release Planner: Efficient Solution for Detailed Airport Surface Traffic Optimization," *the 12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, Indianapolis, IN, September 2012.

[25]Gupta, G., Malik, W., Tobias, L., Jung, Y., Hoang, T., and Hayashi, M., "Performance Evaluation of Individual Aircraft Based Advisory Concept for Surface Management," *the 10th USA/Europe Air Traffic Management (ATM) Research and Development (R&D) Seminars*, Chicago, IL, June 2013.

[26]Hayashi, M., Hoang, T., Jung, Y. C., Malik, W., Lee, H., and Dulchinos, V. L., "Evaluation of Pushback Decision-Support Tool Concept for Charlotte Douglas International Airport Ramp Operations," *the 11th USA/Europe Air Traffic Management (ATM) Research and Development (R&D) Seminars*, Lisbon, Portugal, June 2015.

[27]Scikit-learn for Machine Learning in Python, Version 0.15.2, 2014. http://scikit-learn.org/stable/tutorial/index.html